III Всеукраїнська науково-практична конференція

«Системні науки та інформатика»

Збірник доповідей

25–29 листопада 2024 року

Київ, Україна

Системні науки та інформатика: збірник доповідей III науково-практичної конференції «Системні науки та інформатика», 25–29 листопада 2024 року, Київ. – К., НН ІПСА КПІ ім. Ігоря Сікорського, 2024. – 462 с.

Верстка збірника: Савченко І.О.

ЗМІСТ

Секція 01
Системний аналіз і управління

Секція 02
Системний аналіз фінансового ринку

Секція 03

Інтелектуальні сервіс-орієнтовані розподілені обчислювання

Секція 04

Системи і методи штучного інтелекту

АНАЛІЗ ЗОБРАЖЕНЬ МЕТОДАМИ ШТУЧНОГО ІНТЕЛЕКТУ

Грищенко К.О.[1], Данилов В.Я.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] grishanchik2002@gmail.com

Аналіз зображень методами штучного інтелекту є особливо актуальним напрямком розвитку нейронних мереж. Можливість створення об'ємних карт середовищ на основі зображень та ідентифікація об'єктів на зображеннях є ключовими задачами котрі можуть бути реалізовані методами штучного інтелекту. В роботі було створено моделі штучного інтелекту для виконання цих задач на основі уже існуючих принципів та алгоритмів для реалізації подібних задач, навчання моделей відбувалось на загальнодоступних наборах зображень.

Ключові слова: аналіз зображень, штучний інтелект, карти глибини, ідентифікація об'єктів.

1. ВСТУП

Сучасні технології штучного інтелекту активно використовуються для аналізу зображень, що відкриває нові горизонти в різних галузях, таких як медицина, автомобільна індустрія, робототехніка та багато інших. Одним із ключових напрямів є обробка візуальних даних з використанням карт глибини, ідентифікації об'єктів та точного вимірювання відстаней.

Аналіз зображень з використанням карт глибини дозволяє визначити тривимірну структуру сцени, що зображена на двовимірній площині. Це досягається за допомогою алгоритмів обчислення різниці між позиціями об'єктів відносно камери, використання стереозображень або технологій LiDAR. Карти глибини широко застосовуються у сферах автономного водіння та управління дронами, допомагаючи уникати перешкод і прогнозувати рух.

Ідентифікація об'єктів на зображеннях є ще одним важливим компонентом. Алгоритми ШІ, такі як глибокі нейронні мережі, навчені на великих наборах даних, можуть точно визначати й класифікувати об'єкти у реальному часі. Це відкриває можливості для автоматизації процесів, зокрема у відеоспостереженні, медичній діагностиці, логістиці та виробництві.

Застосування цих методів значно оптимізує роботу спеціалістів, дозволяючи автоматизувати рутинні задачі, підвищувати точність результатів та скорочувати час виконання аналізу. Надійність і доступність таких технологій зростає з кожним роком завдяки новим досягненням у розробці апаратного забезпечення та алгоритмів. Вони формують основу для створення систем підтримки прийняття рішень у візуально-орієнтованих задачах, надаючи користувачам зручний інструмент для оцінки та дій на основі аналізу зображень.

2. ПОСТАНОВКА ЗАДАЧІ

Метою дослідження є реалізація моделей штучного інтелекту, котрі будуть створювати карти глибини для кращого розуміння сцени відтвореної на двох-вимірному зображені та ідентифікувати об’єкти на фотографіях.

3. ПІДХОДИ ДО РЕАЛІЗАЦІЇ ТА ГОТОВІ МОДЕЛІ

Сучасний аналіз зображень, що базується на методах штучного інтелекту (ШІ), пропонує значний вибір готових моделей для вирішення таких задач, як побудова карт глибини, ідентифікація об'єктів та вимірювання відстаней. Ці моделі дозволяють скоротити час розробки систем і значно підвищити їх ефективність завдяки використанню вже навчених нейронних мереж і спеціалізованих алгоритмів.

Для задач побудови карт глибини широко використовуються моделі MiDaS і DPT. MiDaS спеціалізується на оцінці глибини сцени з одного зображення, що особливо корисно в задачах автономного водіння, робототехніки та візуалізації. Вона базується на глибоких нейронних мережах і демонструє високу точність навіть у складних сценах. Модель DPT, яка використовує технологію трансформерів, дозволяє оцінювати тривимірну структуру сцени з вражаючою деталізацією та швидкістю. Для стереозображень популярними є алгоритми StereoBM і StereoSGBM, доступні в бібліотеці OpenCV. Ці моделі дозволяють створювати карти глибини шляхом порівняння пар зображень і мають перевагу у швидкості обробки, що важливо для реальних застосувань.

У сфері ідентифікації об'єктів важливе місце займають моделі YOLO та Faster R-CNN. YOLO є надзвичайно швидкою та ефективною для використання у реальному часі, наприклад, у системах відеоспостереження або потокової аналітики. Faster R-CNN забезпечує високу точність у задачах, де необхідна деталізація, наприклад, в медицині або промисловій автоматизації. Для задач сегментації широко використовується Mask R-CNN, яка дозволяє виділяти об'єкти на зображеннях з точністю до пікселя. Інноваційні Vision Transformers (ViT) пропонують новий рівень продуктивності для складних задач, які вимагають аналізу контексту та великих обсягів даних.

Готові бібліотеки, такі як OpenCV, Detectron2 і TensorFlow Object Detection API, надають інструменти для інтеграції моделей у проекти. OpenCV забезпечує базові алгоритми для роботи з глибинними картами та ідентифікацією об'єктів. Detectron2, створена Facebook AI, пропонує набір сучасних моделей для ідентифікації, сегментації та аналізу. TensorFlow Object Detection API включає попередньо навчені моделі, які легко адаптуються до специфічних задач.

Готові моделі для аналізу зображень стали ключовим елементом у створенні інноваційних систем. Вони дозволяють не тільки швидко розпочати розробку, але й інтегрувати передові технології ШІ для вирішення складних задач.

4. ПРИКЛАДИ РОБОТИ ПРОГРАМНОГО ПРОДУКТУ

У ході досліджень були реалізовані моделі з побудови карт глибини та індетифікації об’єктів мовою програмування Python в середовищі Google Colab, що і є результатами роботи.

Реалізована модель побудови карт глибини виводить результат у вигляді теплової карти, де відстань до об’єкту демонструється у зміні кольору. Так типові результати виконання програми мають виглядають таким чином (рис. 1).

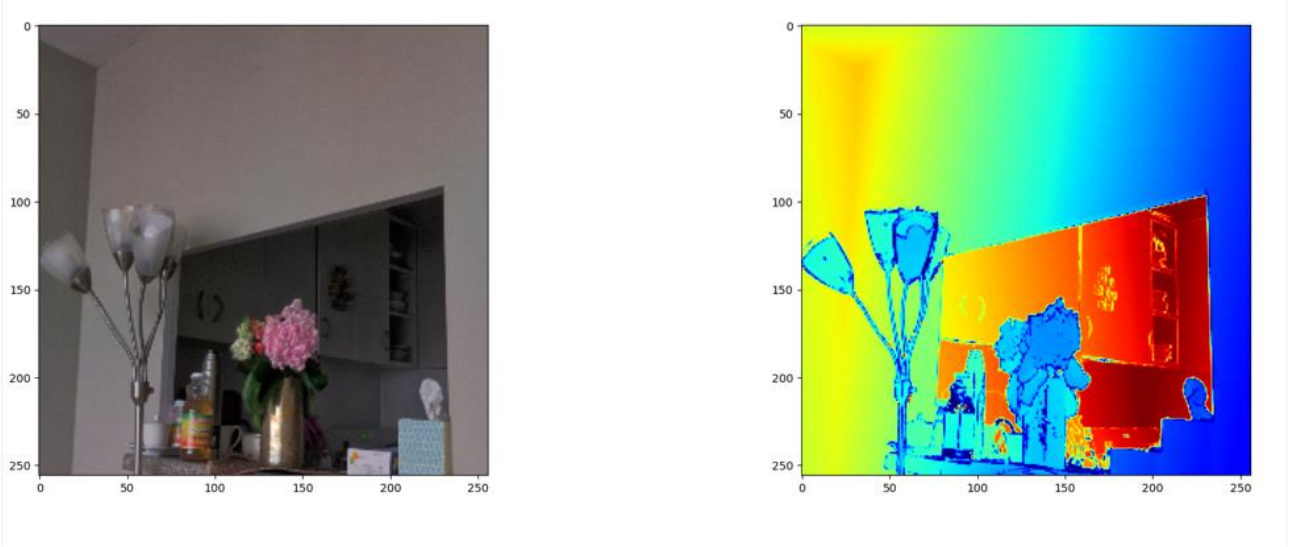


Рисунок 1. Теоретичний результат роботи моделі з побудови карти глибини

Побудова карт глибини важлива для розуміння сцени, що зображена на фотографії, але зважаючи на слабкі сторони монокулярного розпізнавання сцени, а саме відсутність стереоскопічного зображення сцени, що б дало краще розуміння орієнтацію предметів у просторі відносно один одного, розуміння об’єму предметів та інше, потрібно розуміти, що при монокулярному розпізнаванні будуть появлятись помилки, котрі б навряд появились, при стереоскопічному розпізнаванні сцени.

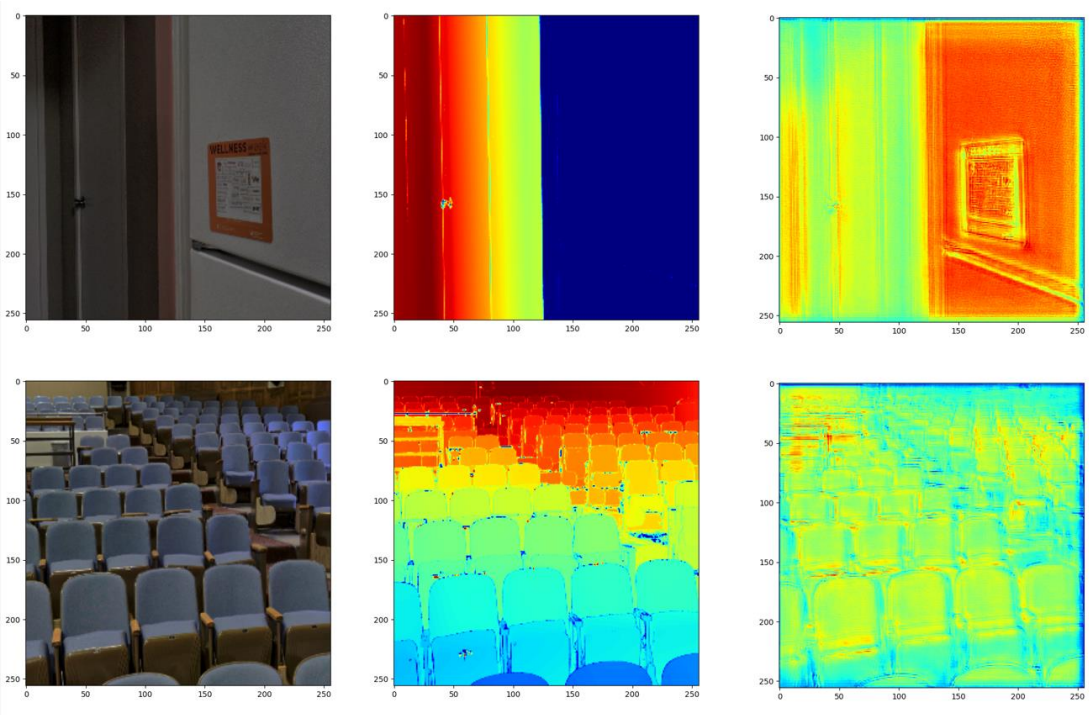


Рисунок 2. Типові помилки монокулярної побудови карт глибини

Так на рис. 2 видно, що монокулярне розпізнавання сцени досить часто, допускає помилку при розпізнаванні фотографій на яких зустрічаються дві площини, а сама помилка полягає в складності розпізнавання яка площина є ближчою до камери. Також типовою

помилкою є недостатнє або надто сильне надання глибини зображенням. Що проілюстровано результатом виконання роботи на сцені зі стільцями, де модель розпізнає стільці як окремі об’ємні предмети, але не розуміє, що стільці що знаходяться на фото вище, знаходяться на більшій відстані від об’єктиву. Але загалом монокулярне розпізнавання є досить перспективним напрямком, адже воно потребує менших затрат на зйомку сцени і загалом видає непогані результати.

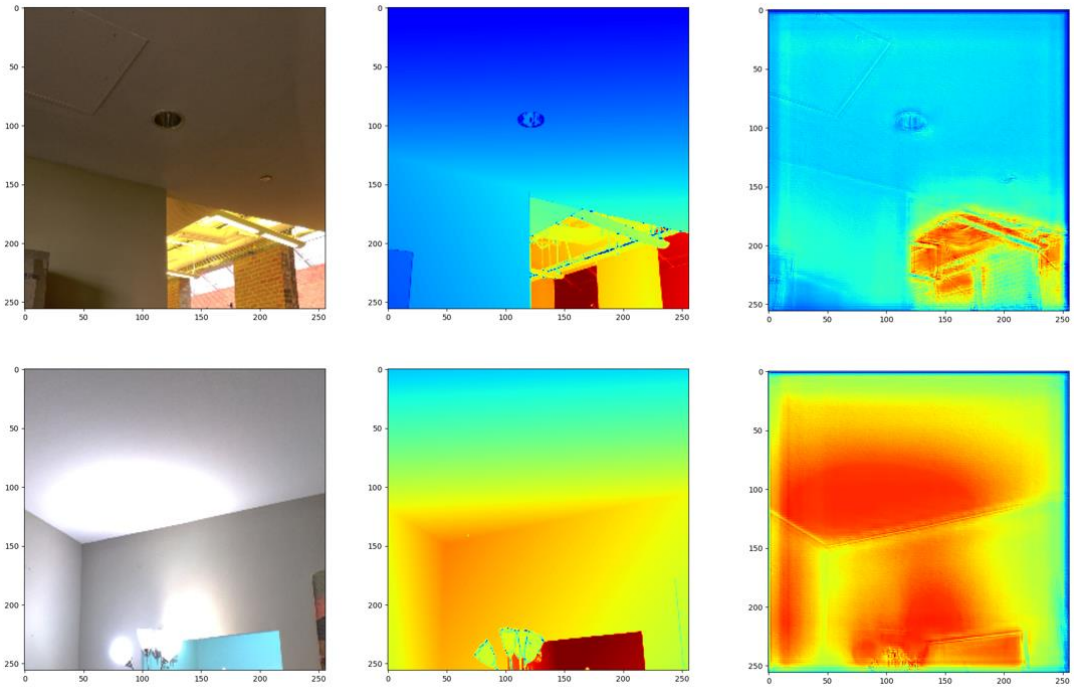


Рисунок 3. Вдалі приклади монокулярного монокулярної побудови карт глибини

Як бачимо з рис. 3, в цілому моделі монокулярного розпізнавання глибини працюють непогано. На прикладі нижче звісно є артефакти пов’язані з занадто яскравими зонами на стіні, але ця помилка не є притаманною лише монокулярним моделям побудови карт глибини, а є недоліком всіх подібних моделей.

Ідентифікація об’єктів.

Реалізація моделей штучного інтелекту з розпізнавання об’єктів зустрічається з проблемою величезної кількості об’єктів які існують в світі багато з яких є схожими один на одного та загалом реалізувати нейронну мережу яка розпізнає об’єкти кожного типу виглядає як задача яку майже неможливо реалізувати.

Реалізована мною модель розпізнавання об’єктів добре справляється з тривіальними об’єктами на яких моделі навчають (рис. 4).

Але заради об’єктивності потрібно розуміти що модель може не розпізнавати нетривіальні об’єкти, які рідко є присутніми в навчальних вибірках. Так на рисунку 5 видно що півсфери які відмежовують тротуари від доріг були розпізнані як спортивні м’ячі.

Рисунок 4. Результат роботи моделі з розпізнавання об’єктів

Рисунок 5. Розпізнавання об’єктів на перенасиченому об’єктами фото

5. ВИСНОВКИ

В даній роботі було реалізовано моделі штучного інтелекту для аналізу зображень. В процесі виконання роботи було розглянуто багато напрямів роботи над поставленою задачею

і відібрано основний, що у був реалізований. Результатом роботи стали моделі штучного інтелекту котрі створюють карти глибини та розпізнають об’єкти. На основі отриманих результатів були розглянуті сильні та слабкі сторони обраних методів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
2. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). "ImageNet Classification with Deep Convolutional Neural Networks". *Advances in Neural Information Processing Systems*, 1097–1105.
3. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). "You Only Look Once: Unified, Real-Time Object Detection". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 779–788.
4. Russakovsky, O., Deng, J., Su, H., et al. (2015). "ImageNet Large Scale Visual Recognition Challenge". *International Journal of Computer Vision*, 115(3), 211–252.
5. "How to Estimate Depth from a Single Image" Jacob Marks, Ph.D. Jan 25, 2024 https://towardsdatascience.com/how-to-estimate-depth-from-a-single-image-7f421d86b22d

СТРАТЕГІЯ РОЗРОБКИ КОМП'ЮТЕРНОЇ МОДЕЛІ ЦИФРОВИХ ДВІЙНИКІВ ДЛЯ СУПРОВОДУ ФУНКЦІОНУВАННЯ МОСТІВ ЯК КІБЕРФІЗИЧНИХ СИСТЕМ

Грішин К.Д.[1], Панкратова Н.Д.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] constantine1223h@gmail.com [0009-0006-5950-3739],
[2] natalidmp@gmail.com [0000-0002-6372-5813]

Метою роботи є реалізація запропонованої стратегії розробки комп'ютерної моделі цифрових двійників для супроводу функціонування мостів як кіберфізичних систем, яка використовує сучасні математичні методи системного аналізу та технології штучного інтелекту. Результатом є продукт на мові програмування Python із залученням середовища ANSYS, який являє собою комп'ютерну модель цифрового двійника мосту. Новизна полягає в удосконаленні процесу автоматизованої модифікації дизайну моста інноваційними сурогатними моделями та налаштуванням їх гіперпараметрів, подальшому розвитку архітектури нейронних мереж системи динамічного зважування. Для модулю динамічного зважування згенерована синтетична вибірка розміром 6028 елементів, на якій навчені моделі ШІ спроможні оцінювати навантаження від транспортних засобів. Запропонована архітектура мала високу точність. Система може використовуватись в мостобудівельній галузі.

Ключові слова: цифровий двійник, метод скінченних елементів, алгоритм рою частинок, сурогатна модель, система динамічного зважування, рекурентні нейронні мережі.

1. ВСТУП

Українська економіка та інфраструктура дедалі більше зазнають руйнувань із початку повномасштабного вторгнення. Щоб її майбутнє відновлення відповідало сучасним стандартам, повинні залучатися досягнення Industry 4.0, зокрема, цифрові двійники (ЦД). Необхідність опанування технології ЦД продиктована не лише обсягом робіт, але й загальним зростанням їх складності. Класична пірамідальна методологія проєктування систем управління підприємством витісняється підходами прямої взаємодії між її компонентами на основі Інтернету речей [1]. *Однак в мостобудівельній галузі ця технологія перебуває на ранній стадії розвитку, поки не адаптувавши усіх передових методів вирішення задач, які в ній виникають, фокусуючись, в основному, на моніторингу стану готових конструкцій*.

Базова концепція ЦД – наявність фізичного об'єкта, комп'ютерної моделі та механізму забезпечення обміну інформацією між ними. *ЦД є унікальним засобом досягнення кіберфізичної інтеграції, що дозволяє здійснювати постійний моніторинг, діагностику, прогнозування та оптимізацію спостережуваного процесу/об'єкта/системи, забезпечуючи живучість його функціонування на всьому життєвому циклі* [2].

У рамках дослідження створено комп'ютерну модель цифрового двійника мосту, що складається з трьох компонент.

- *Моделі мосту на основі методу скінченних елементів* (Finite Element Method, FEM), розробленої в програмному забезпеченні ANSYS. Вона проводить розрахунки фізичних характеристик за заданими параметрами та обмеженнями, виконує роль «полігону» для випробування конструкцій моста, тобто є «віртуальною лабораторією» для експериментів над дизайном споруди та побудови сценаріїв.

- *Модуля дизайну конструкції* – системи автоматизованого пошуку раціонального проєкту конструкції. Вона дозволяє задати цільову функцію, обмеження та параметри, які оптимізуватимуться алгоритмом рою частинок із сурогатними моделями. Містить базу попередньо випробуваних дизайнів, на якій навчаються сурогатні моделі.

- *Модуля динамічного зважування* – системи для створення штучного інтелекту (ШІ), який за показниками сенсорів механічної напруги мосту оцінює навантаження на осі транспортних засобів, що його перетинають.

2. ОСНОВНІ МЕТОДИ, МОДЕЛІ

2.1. Модель із застосуванням методу скінченних елементів

У будівництві основним інструментом високоточного моделювання фізичних процесів є метод скінченних елементів (Finite Element Method, FEM) – *чисельний метод розв'язку диференціальних рівнянь в частинних похідних на основі дискретизації.* На об'єкт накладається двовимірна або тривимірна (для рівнянь у просторі) сітка довільної форми, поділяючи складну область на прості підобласті (скінченні елементи), у межах яких відбувається апроксимація простими функціями [3]. Приклад сітки показано на рисунку 1.

Скінченні елементи перетинаються в деяких площинах, лініях, точках. У двовимірному просторі скінченні елементи можуть бути трикутниками, квадратами; у тривимірному – кубами, пірамідами, паралелепіпедами та ін. У результаті здійснюється перетворення до системи лінійних алгебраїчних рівнянь з розрідженою матрицею [3].

FEM дозволяє з високою точністю оцінювати ключові характеристики конструкції: строк безвідмовного функціонування, деформацію під навантаженнями, запас міцності тощо. Варіюючи параметри проводять порівняльний аналіз проєктів споруди: матеріали, форми та розміри деталей і т.д. На етапі експлуатації конструкції за допомогою FEM будують «What-if» сценарії, роблять прогнози, проводять моніторинг стану споруди [4].

FEM застосовуватиметься як при проєктуванні споруди, так і її обслуговуванні. Універсальність FEM робить доцільним її використання в комп'ютерній моделі ЦД мосту.

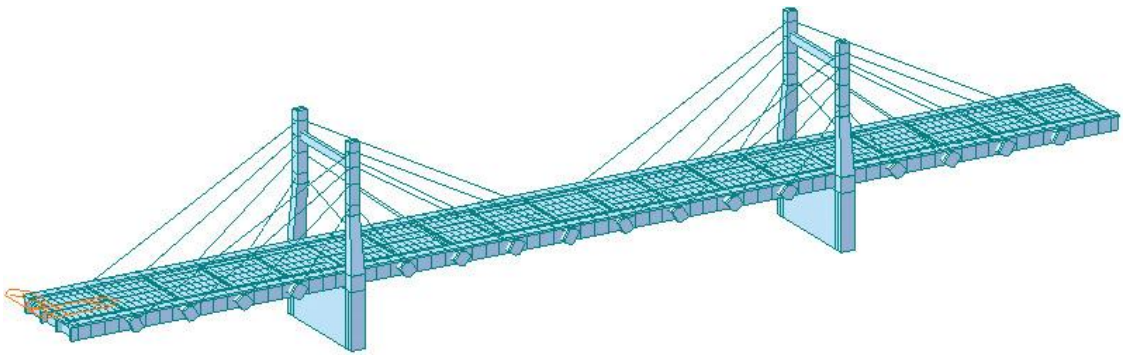


Рисунок 1. Модель із застосуванням FEM вантового мосту. Видно накладену тривимірну сітку. Джерело [5].

2.2. Алгоритм рою частинок, сурогатні моделі у модулі дизайну

Функцією модулю дизайну є пошук оптимального проєкту конструкції мосту за заданих вимог та обмежень, що формалізується як задача (1).

$$f(\vec{x}) \to min$$
$$g_i(\vec{x}) \le 0, i = \overline{1,m} \quad (1)$$
$$\vec{x} \in S,$$

де $f(\vec{x})$ – цільова функція, яка мінімізується;

$\vec{x} \in R^n$ – вектор параметрів (наприклад, розміри, матеріал і т.д.);

$g_i(\vec{x}), i = \overline{1,m}$ – функції-обмеження у формі нерівностей (вимоги до надійності: запасу міцності, строку функціонування та ін.);

S – простір оптимізації (заданий діапазон параметрів).

Значення функцій-обмежень обчислюються FEM, тому вони не представлені в аналітичному вигляді, що виключає застосування традиційних методів оптимізації. Альтернативою є метаевристичні алгоритми, в основі яких лежать математичні моделі природніх явищ. Вони не гарантують знаходження глобального мінімуму, втім, на практиці значно зменшують цільову функцію [6].

Метаевристичним алгоритмом у роботі обрано *алгоритм рою частинок з топологією*, заснований на моделі соціальної поведінки зграї птахів при пошуку їжі. Аналогом зграї є рой частинок, які шукають мінімум та обмінюються між собою інформацією [7].

Частинки задаються векторами:

- $\overrightarrow{x_k}(t) \in R^n$ – позиція, яка визначає поточний розв'язок в просторі оптимізації S;
- $\overrightarrow{v_k}(t) \in R^n$ – швидкість, що керує зміною позиції.

Частинка k має пам'ять про власний кращий виявлений розв'язок та кращий розв'язок, знайдений множиною частинок S_k, із якими у неї є зв'язок. Вектори $\overrightarrow{x_k}(t), \overrightarrow{v_k}(t)$ змінюються за стохастичною формулою, яка враховує ці два види пам'яті, в дискретному часі t. Мережа інформаційних зв'язків рою має назву топології [7].

Моделі із застосуванням FEM обчислювально витратні, ситуація погіршується тим, що метаевристичні алгоритми під час своєї роботи перебирають тисячі варіантів. Для прискорення процесу оптимізації використовують *сурогатні моделі – методи машинного навчання, які апроксимують високоточну модель, маючи вищу швидкість обчислень* [8].

Тобто матимемо два способи розрахунку значень: високої точності за допомогою FEM, нижчої точності – сурогатними моделями. Останні здатні виявити загальні тренди і тільки наближають алгоритм до глобального мінімуму, не знаходячи його. Їх задачею є прискорення процесу оптимізації.

Роль сурогатних моделей на практиці виконують регресійні моделі, штучні нейронні мережі, кригінг та ін. У роботі в якості сурогатних моделей використані:

- радіально-базисні нейронні мережі;
- випадкові ліси;
- машини опорних векторів;
- мережі Колморогова-Арнольда (Kolmogorov-Arnold Network, KAN).

Гіперпараметри моделей налаштовуються баєсівською оптимізацією [9].

2.3. Методи штучного інтелекту для модулю динамічного зважування

Системи динамічного зважування будують прогноз на основі сигналів множини сенсорів: вхідними даними є багатовимірні часові ряди. Тому архітектура нейронних мереж будувалась на основі рекурентних нейронних мереж. У роботі застосовувались прості комірки RNN та Gated Recurrent Unit (GRU).

Використана також технологія векторного представлення (embedding) часових рядів Time2Vec. Вона перетворює параметр часу t на вектор $\overrightarrow{t2v}(t)$, що задається формулою (2).

$$\overrightarrow{t2v}(t) = \left(\omega_0 t + \varphi_0, F(\omega_1 t + \varphi_1), \dots, F(\omega_i t + \varphi_1), \dots, F(\omega_n t + \varphi_n)\right), \quad (2)$$

де $\omega_s, \varphi_s, s = \overline{0, N}$ – параметри, що здатні навчатися;

F – функція активації, яка є періодичною (наприклад, синус);

Вектор (2) подається поміж інших вхідних параметрів в модель ШІ. Перша координата здатна виявити лінійні тренди, інші – періодичні. Під час навчання ω_s, φ_s коригуються, тому досліднику нема необхідності задавати параметри трендів у явному вигляді. Розробники методу стверджували, що застосування Time2Vec в тандемі з рекурентними нейронними мережами покращувало результати [10].

Однак на практичних задачах перетворення Time2Vec також стало застосовуватись не до часу t, а безпосередньо до часового ряду $\overrightarrow{x_t}$ (відповідно, перетворення проходять по кожній координаті вектору $\overrightarrow{x_t}$). Цей підхід використано в роботі.

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Результатом проведеного дослідження є комп'ютерна модель цифрового двійника мосту, яка використовує сучасні математичні методи системного аналізу та технології штучного інтелекту. Програмний продукт розроблено на мові програмування Python із залученням середовища ANSYS. Взаємодія з користувачем забезпечується інтерфейсом.

Експеримент складався з двох частин, які відповідали етапам проєктування та експлуатації.

3.1. Експеримент із модулем дизайну

Перша частина – розв'язання модулем дизайну задачі мінімізації маси мосту на балках коробкового типу, що підтримуються двома колонами. Колони зроблені з бетону, інші елементи (балки, настил) – з конструкційної сталі. Довжина мосту – 60 метрів з колонами через кожні 20 метрів. Конструкція має по дві автомобільні та одній пішохідній смузі в обидві сторони. Задача загалом мала 12 параметрів та три функції-обмеження (запас міцності, максимальна деформація, механічна напруга в колонах).

Вимоги до автомобільних та пішохідних смуг споруди, модель тестового навантаження взяті зі державного стандарту України ДБН В.1.2-15:2009 «Мости та труби. Навантаження і впливи» [11]. Використовувався метод оптимізації на основі алгоритму рою частинок з топологією (три типи: кільце, повнозв'язна, два кластери) та сурогатних моделей.

Сурогатні моделі дали суттєву перевагу порівняно з алгоритмом без їх участі. У ході проведення досліду з оптимізації було накопичено 4184 варіантів конструкції мосту. Усі типи сурогатних моделей досягли на цій вибірці достатньої точності, що доводить їх адекватність. Показано, що збільшення обсягу даних підвищує точність моделей. Найкращі результати мали KAN та випадковий ліс.

У роботі на прикладі однієї з функцій-обмежень (механічної напруги в колонах) продемонстровано здатність KAN будувати модель, яка має зручну графічну інтерпретацію, виводити аналітичну формулу для оцінки характеристик залежно від параметрів. Формула може бути корисною при аналізі фізичних процесів у конструкції.

3.2. Модуль динамічного зважування

Друга частина експерименту – створення системи динамічного зважування на основі моделі ШІ. Моделлю із застосуванням FEM згенеровано навчальні вибірки для трьох типів вантажівок (з різною кількістю осей), розміром 1955, 2034, 2039 елементів, відповідно. Кожен тип транспортного засобу мав свою нейронну мережу для оцінки навантаження на осі (тобто система мала три нейронні мережі). Вибірка включала різний розподіл навантаження по автомобільним осям та швидкість транспортних засобів.

Запропонована в роботі архітектура нейронної мережі для системи динамічного зважування складається з:

- послідовності RNN-комірок;
- шару Time2Vec;
- послідовності повнозв'язних шарів.

У досліді перебирались варіанти архітектури: кількість комірок RNN, їх тип та розмірність векторів прихованого стану, число повнозв'язних шарів. Набір даних поділено на навчальну та перевірочну вибірку як 9:1. Кращі архітектури було перевірено додатково методом крос-валідації, для них середня абсолютна відносна похибка становила до 2.16% від навантаження на вісь.

4. ВИСНОВКИ

Результатом дослідження є розроблений програмний продукт на мові Python із залученням середовища ANSYS, який являє собою комп'ютерну модель цифрового двійника мосту. Система складається з моделі на основі методу скінченних елементів в ANSYS, модулю дизайну та модулю динамічного зважування.

Модуль дизайну здійснює автоматизовану модифікацію проєкту мосту алгоритмом рою частинок з топологією, із використанням сурогатних моделей та калібруванням їх гіперпараметрів. Для його випробування сформульовано задачу мінімізації вагу мосту із 12 параметрами та трьома функціями-обмеженнями на характеристики конструкції. Досліди показали, що сурогатні моделі суттєво покращують результати оптимізації, найкращими виявились запропоновані випадкові ліси та мережі Колмогорова-Арнольда.

Модуль динамічного зважування створює штучний інтелект, що оцінює навантаження від транспортних засобів за сигналами сенсорів. У ході його випробування моделлю на основі FEM згенерована синтетична вибірка розміром 6028 елементів, на якій навчені моделі ШІ. Запропонована архітектура нейронних мереж дає високу точність (середня абсолютна відносна похибка становила до 2.16% від навантаження на вісь).

Робота доводить ефективність запропонованої стратегії розробки комп'ютерної моделі цифрових двійників для супроводу функціонування мостів як кіберфізичних систем. У подальшому доцільно її розвивати використанням складніших моделей та більших обчислювальних потужностей.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Pankratova N. D., Grishyn K. D. The Strategy of Digital twin Implementation at the Cyber-Physical Enterprises. Global Journal of Computer Science and Technology. 2024. Vol 24., No. H1. P. 1–12. DOI: 10.34257/GJCSTHVOL24IS1PG1.

2. Pankratova N. D., Grishyn K. D., Barilko V. E. Digital twins: stages of concept development, areas of use, prospects. System research and information technologies. 2023. No. 2. P. 7–21. DOI: 10.20535/SRIT.2308-8893.2023.2.01.

3. Logan D. L. A First Course in the Finite Element Method. Canada; Toronto: Thomson, 2007. 808 p. URL: https://kntu.ac.ir/DorsaPax/userfiles/file/Mechanical/OstadFile/dr_nakhodchi/DarylL.LoganAFirstCourse.pdf

4. Zanjani Zadeh V., Patnaik A. Finite element modeling of the dynamic response of a composite reinforced concrete bridge for structural health monitoring. International Journal of Advanced Structural Engineering (IJASE). 2014. Vol. 6, No. 2. P. 14. DOI: 10.1007/s40091-014-0055-4

5. Zhang K., Qi T., Li D., Xue X. та ін.;. Health Monitoring-Based Assessment of Reinforcement with Prestressed Steel Strand for Cable-Stayed Bridge. SDHM Structural Durability and Health Monitoring. 2022. Vol. 16, No. 1. P. 53–80. DOI: 10.32604/sdhm.2022.016130

6. Tsai C.-W., Chiang M.-C. Handbook of Metaheuristic Algorithms. United States; Cambridge: Academic Press, 2023. 622 p. DOI: 10.1016/C2022-0-00638-3.

7. Li X.-L., Serra R., Olivier J. An Investigation of Particle Swarm Optimization Topologies in Structural Damage Detection. Applied Sciences. 2021. Vol. 11, No. 11. P. 15. DOI: 10.3390/app11115144.

8. Ma Y., Song C., Wang Z., Jiang Z. та ін.;. Efficient Design Optimization of Cable-Stayed Bridges: A Two-Layer Framework with Surrogate-Model-Assisted Prediction of Optimum Cable Forces. Applied Sciences. 2024. Vol. 14, No. 5. P. 22. DOI: 10.3390/app14052007

9. Garnett R. Bayesian Optimization. England; Cambridge: Cambridge University Press, 2023. 358 p. DOI: 10.1017/9781108348973.

10. Time2Vec: Learning a vector representation of time. / Kazemi S. M., Goel R., Eghbali S., Ramanan J. та ін.; arXiv, 2019. 16 p. DOI: 10.48550/arXiv.1907.05321.

11. ДБН В.1.2-15:2009 "Мости та труби. Навантаження і впливи". URL: https://e-construction.gov.ua/laws_detail/3074218647841932431?doc_type=2

ІНСТРУМЕНТАРІЙ РОЗРОБКИ ПІДХОДІВ ЩОДО ДЕТЕКЦІЇ ДРОНІВ ЗА ДОПОМОГОЮ КАМЕРИ ТА ДАТЧИКІВ

Майборода О.І.[1], Савастьянов В.В.

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] kavabanga200188@gmail.com

Розробка підходів для детекції дронів за допомогою камери та датчиків є важливою задачею в сучасних дослідженнях в області безпеки, автоматизації та штучного інтелекту. Використання камери та різноманітних датчиків дозволяє реалізувати системи спостереження, які можуть виявляти дрони в реальному часі, навіть в складних умовах. Ключовими технологіями, що використовуються для цієї мети, є комп'ютерний зір, сенсори руху, радіолокація та інфрачервоні технології. Завдяки застосуванню алгоритмів машинного навчання та глибинного навчання можна підвищити точність детекції та знизити кількість хибних спрацьовувань. Метою цієї роботи є розробка нових підходів до детекції дронів, які інтегрують дані з камери та датчиків, а також порівняльний аналіз ефективності різних методів виявлення та відслідковування дронів.

Ключові слова: детекція дронів, комп'ютерний зір, датчики, машинне навчання, YoLo10, автоматизація, системи спостереження.

1. ВСТУП

У сучасному світі стрімкий розвиток технологій безпілотних літальних апаратів (дронів) створює нові можливості, проте водночас викликає низку викликів, пов’язаних із забезпеченням безпеки та контролю. Зокрема, проблема виявлення дронів у реальному часі набуває критичного значення у сфері громадської безпеки, охорони об’єктів стратегічного значення, а також у військовій галузі.

Сучасні підходи до детекції дронів базуються на використанні різноманітних сенсорних технологій, таких як оптичні камери, акустичні сенсори, радіочастотні аналізатори та інфрачервоні датчики. Особливу увагу привертає інтеграція даних з цих джерел за допомогою алгоритмів комп’ютерного зору та машинного навчання, які дозволяють здійснювати автоматизовану обробку інформації та підвищувати точність виявлення.

Метою цієї роботи є дослідження сучасних підходів до детекції дронів із використанням камер і датчиків, розробка інноваційних методів інтеграції даних із різних джерел, а також проведення порівняльного аналізу ефективності запропонованих рішень. Це дозволить оцінити потенціал поєднання традиційних та інтелектуальних технологій у завданнях моніторингу повітряного простору.

2. СИСТЕМНИЙ ПІДХІД ЩОДО ПОБУДОВИ ІНСТРУМЕНТАРІЮ

Системний підхід щодо побудови інструментарію представлено на рис. 1.

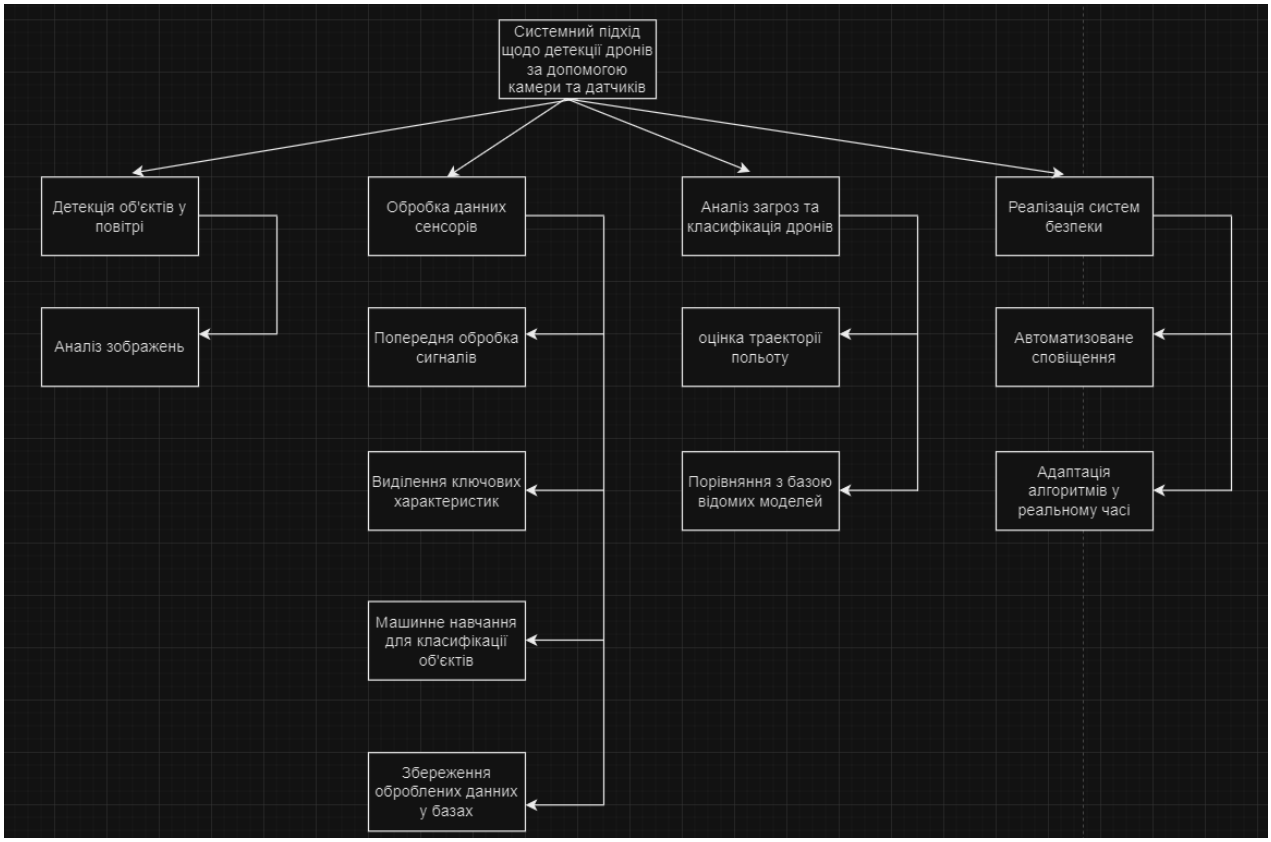


Рисунок 1. Структура підходу побудови інструментарію

Ціль системи.

- Забезпечення своєчасного виявлення дронів у повітряному просторі.
- Оцінка ризиків, пов’язаних із діяльністю дронів.
- Автоматизація процесу моніторингу та аналізу.

Ключові компоненти моделі

Сенсорна система

- **Оптичні камери:**
  Використовуються для аналізу зображень за допомогою комп’ютерного зору.

Алгоритми обробки даних

- **Попередня обробка сигналів:**
  Фільтрація шумів та виділення ключових характеристик.
- **Машинне навчання:**
  Використання класифікаційних моделей для ідентифікації об’єктів.
- **Аналіз траєкторій:**
  Виявлення закономірностей у поведінці дронів.

База знань

- Накопичення даних про відомі моделі дронів.
- Зберігання результатів детекції для подальшого аналізу.

Інтерфейси взаємодії

- Система сповіщення (автоматичне інформування операторів).
- Аналітичний модуль для прогнозування ризиків.

Сценарії використання

- **Оптимістичний сценарій:**
  Система виявляє дрон на ранній стадії, автоматично визначає його тип і сповіщає відповідальні служби.
- **Нейтральний сценарій:**
  Дрон ідентифіковано, але тип і мета не визначені, вживаються запобіжні заходи.
- **Песимістичний сценарій:**
  Неправильна ідентифікація об’єкта або запізніла реакція на вторгнення.

Очікувані результати

- Зниження ризиків, пов’язаних із неконтрольованим використанням дронів.
- Автоматизація процесів моніторингу та аналізу повітряного простору.
- Підвищення ефективності системи безпеки.

Ця модель може слугувати основою для розробки сценаріїв впровадження систем детекції дронів та їх оцінки.

3. МОДЕЛЮВАННЯ ТА РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для дослідження був використаний дата сет з Kaggle, у якому зібрані та розмічені фотографії з дронами. Як видно з рис. 2, розроблена модель добре справляється з поставленою задачею детекції дрона. Далі було реалізовано розрахування дистанції до дрона (рис. 3).

Рисунок 2. Приклад детекції дрона

Рисунок 3. Приклад з розрахованою дистанцією

4. ВИСНОВКИ

У ході виконання роботи було розглянуто і реалізовано підхід до детекції дронів за допомогою камери та сенсорів, що базується на використанні сучасних технологій комп'ютерного зору, машинного навчання та інтеграції багатьох джерел даних. Отримані результати дозволяють зробити такі висновки:

Ефективність методів детекції

Застосування моделей глибинного навчання, таких як YOLO, дозволяє досягти високої точності ідентифікації дронів навіть у складних умовах.

Розрахунок дистанції до дрона

Розроблено методику розрахунку відстані до дрона на основі перспективної геометрії та калібрування камери. Це дозволяє точно визначати місцезнаходження об'єкта у просторі, що є важливим для завдань безпеки та контролю.

Модульний підхід до розробки системи

Використання модульної архітектури (детекція, обробка даних, аналіз загроз, формування сценаріїв) забезпечує гнучкість і масштабованість системи. Це дозволяє легко адаптувати систему до нових умов або інтегрувати додаткові компоненти.

Можливості інтеграції з існуючими системами безпеки

Розроблена система може бути інтегрована з іншими технологіями моніторингу для забезпечення комплексного підходу до захисту об'єктів та повітряного простору. Зокрема,

автоматизоване сповіщення та прогнозування ризиків можуть значно знизити час реакції на потенційні загрози.

Обмеження та подальші перспективи

Система демонструє високу ефективність за умови належного калібрування камери та точності попередньо заданих параметрів дронів. У майбутньому доцільно дослідити використання нейромереж для автоматичного калібрування та підвищення точності в реальних умовах, а також інтеграцію з алгоритмами відстеження траєкторій руху.

Таким чином, розроблений підхід до детекції дронів за допомогою камери та сенсорів є перспективним інструментом для розв'язання завдань безпеки та моніторингу. Подальший розвиток системи може включати розширення функціоналу, оптимізацію алгоритмів і застосування у різних сферах, таких як військова безпека, охорона критичних об’єктів, моніторинг територій і боротьба з несанкціонованими польотами.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Zhang, X., Wang, H., & Li, L. (2020). "Real-Time UAV Detection and Tracking Using Deep Learning Approaches."

2. Kumar, V., & Roshan, A. (2021). "A Survey on UAV Detection Techniques: Challenges and Opportunities."

3. Erkmen, I., & Kilic, K. (2019). "Vision-Based UAV Detection Using Convolutional Neural Networks."

4. Mohamed, A. A., & Farag, M. M. (2022). "Sensor Fusion for UAV Detection: Integrating Acoustic and Visual Data."

5. Gupta, P., & Singh, A. (2021). "Drone Detection Using YOLO and Distance Estimation Based on Perspective Geometry."

ПРОГНОЗУВАННЯ ЕКОНОМІЧНИХ ПРОЦЕСІВ НА ОСНОВІ НОВИН З ВІДКРИТИХ ДЖЕРЕЛ ЗА ДОПОМОГОЮ НЕЙРОННИХ МЕРЕЖ

Машталяр Ю.О.[1], Данилов В.Я.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] mashtaliaryurii@gmail.com

Сучасні методи прогнозування економічних процесів базуються на аналізі часових рядів і врахуванні зовнішніх факторів, зокрема новинного контексту. У роботі запропоновано підхід, що об'єднує модель LSTM із текстовими аналізаторами Word2Vec, GloVe та GPT для врахування впливу новин. Дані про валютний курс CHF/USD за період 2024 року використано для навчання та тестування моделей. Отримані результати показали, що інтеграція новинного контексту знижує похибки прогнозування, покращуючи точність. Наукова новизна полягає в поєднанні моделей часових рядів із сучасними методами обробки тексту, що підвищує адаптивність до ринкових змін. Практична значимість підтверджена застосуванням у фінансовій аналітиці.

Ключові слова: прогнозування, нейронні мережі, текстовий аналіз, LSTM.

1. ВСТУП

У сучасних умовах глобалізації та цифровізації економічних процесів роль оперативного та точного прогнозування фінансових показників набуває особливого значення. Коливання валютних курсів, цін на сировину чи фондових індексів безпосередньо впливають на прийняття рішень у бізнесі, інвестиціях і державній політиці. Традиційні методи аналізу та прогнозування, що ґрунтуються виключно на історичних даних, поступово втрачають свою ефективність через зростаючу складність і динамічність економічних систем. У цьому контексті новини з відкритих джерел стають важливим джерелом інформації, що здатне суттєво доповнити традиційні підходи.

Метою представленої роботи є розробка комплексної моделі прогнозування економічних процесів, що інтегрує аналіз історичних даних із оцінкою новинного контексту. Для цього було реалізовано модель LSTM, яка здійснює прогнозування на основі часових рядів валютних курсів, та створено три текстові аналізатори — Word2Vec, GloVe та GPT. Аналізатори оцінюють новини, формуючи векторні репрезентації тексту, що дозволяє отримати семантичну значущість кожної новини. Надалі ці оцінки інтегруються з результатами LSTM, створюючи модель комплексного прогнозування, яка враховує як історичні дані, так і поточний новинний фон.

Запропонований підхід дозволяє враховувати широкий спектр факторів, які впливають на економічні процеси, та оцінювати їхню значущість у реальному часі. Експериментальні дослідження демонструють, що поєднання традиційних моделей часових рядів із текстовими аналізаторами новин значно покращує точність прогнозування. Такий підхід відкриває нові можливості для прогнозування економічних показників та прийняття обґрунтованих рішень на їх основі.

2. МОДЕЛІ ПРОГНОЗУВАННЯ

У сучасних дослідженнях прогнозування економічних процесів активно застосовуються нейронні мережі завдяки їхній здатності ефективно аналізувати великі обсяги даних і знаходити складні залежності між ними. Однією з найбільш перспективних архітектур у задачах обробки часових рядів є рекурентні нейронні мережі (RNN) та їхні вдосконалені версії, такі як LSTM і GRU. Водночас інтеграція економічних новин до моделей прогнозування дозволяє враховувати зовнішні події, що впливають на ринкову ситуацію, і отримувати більш точні результати. У межах даної роботи було реалізовано декілька підходів до аналізу текстових новин і їхнього інтегрування до моделі прогнозування на основі LSTM.

Основою комплексного підходу став алгоритм **Long Short-Term Memory (LSTM)** – різновид рекурентних нейронних мереж, розроблений для роботи з довгими часовими послідовностями. Завдяки спеціальній структурі комірок LSTM здатна зберігати та використовувати контекстні залежності, що є критично важливим у задачах прогнозування валютних курсів. У межах роботи модель була навчена на часових рядах історичних даних про курси валют, що дозволило створити базовий механізм прогнозування, який враховує патерни та сезонні коливання.

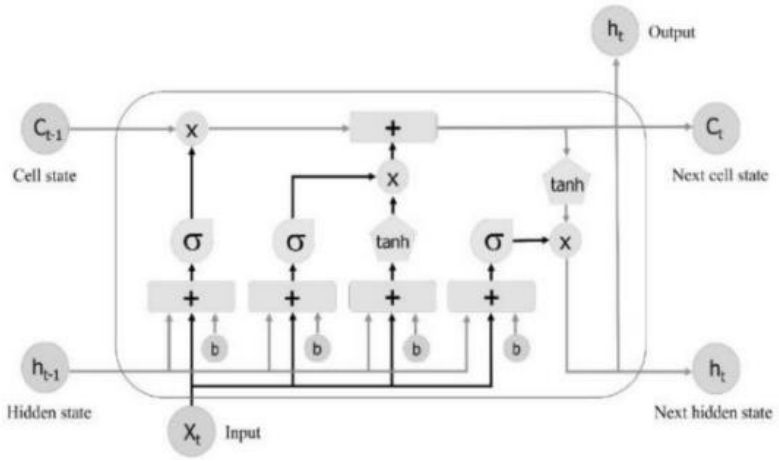


Рисунок 1. Структура LSTM

Для аналізу текстових новин були використані три підходи до векторизації тексту: **Word2Vec**, **GloVe** та **GPT**. Кожен із цих аналізаторів виконує роль генератора векторного представлення тексту, яке надалі використовується як вхідні дані для моделі прогнозування.

Word2Vec — алгоритм, що створює щільні векторні репрезентації слів, враховуючи їх контекст у реченні. Було застосовано Continuous Bag of Words (CBOW) і Skip-Gram підходи для тренування моделей на новинному датасеті. Вектори, створені Word2Vec, використовувалися для подання семантичного значення новин.

GloVe (Global Vectors for Word Representation) – метод, що будує векторні репрезентації, враховуючи глобальні статистичні характеристики тексту. На відміну від Word2Vec, GloVe враховує не лише локальний контекст, а й спільну частотність слів у всьому корпусі даних, що дозволяє отримувати більш узагальнені та точні вектори.

GPT (Generative Pre-trained Transformer) – трансформерна модель, що надає сучасні результати в обробці природної мови. Завдяки своїй багаторівневій архітектурі GPT здатна не лише визначати значення окремих слів, а й оцінювати тональність та смислові зв'язки в тексті. У рамках роботи GPT використовувалась для отримання векторів, які враховують контекст і загальний вплив новин на економічну ситуацію.

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для оцінки ефективності запропонованих моделей було проведено серію експериментів, спрямованих на прогнозування валютного курсу швейцарського франка (CHF) до долара США (USD). Ця валютна пара була обрана через її високу стабільність і значення в міжнародних фінансових операціях, що робить її ідеальним об'єктом для аналізу складних економічних процесів. Дані про історичні курси валют охоплюють період із 01.01.2024 по 01.11.2024, який включає 10 місяців активних змін на валютному ринку. Такий часовий інтервал дозволив моделі врахувати як короткострокові, так і довгострокові тренди, а також реакцію ринку на ключові економічні події. Історичний курс швейцарського франка до долара США наведено на рис. 2.

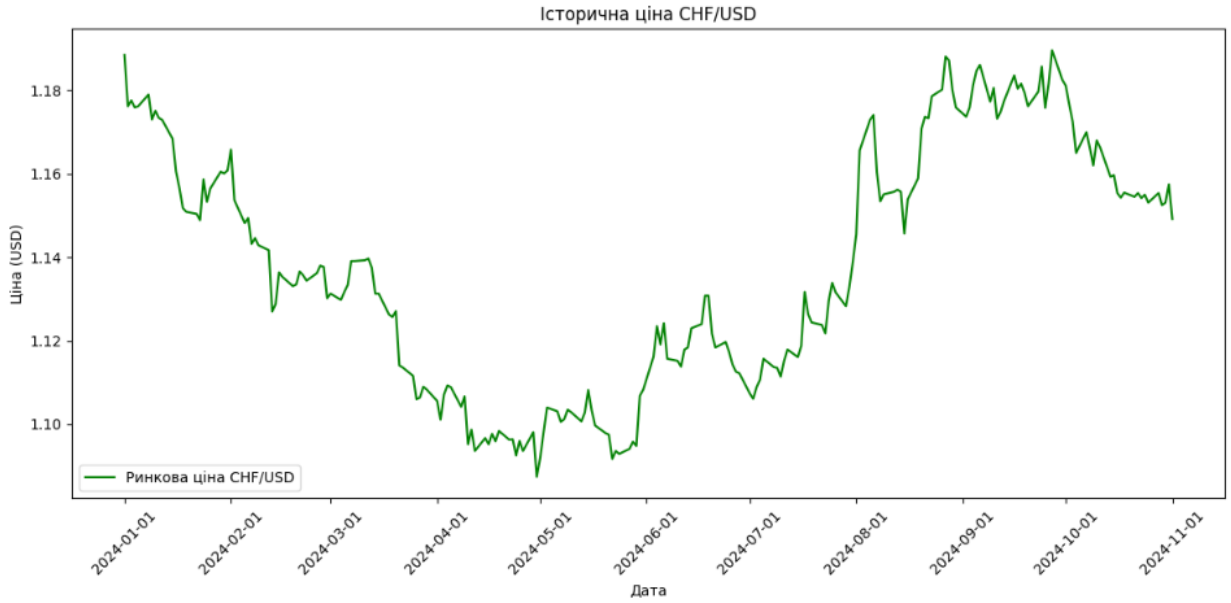


Рисунок 2. Історична ціна CHF/USD

Для покращення якості прогнозування модель була доповнена зовнішніми даними у вигляді новинного контексту. Джерелом новин стали авторитетні англомовні платформи, такі як Bloomberg, Investing.com, Reuters, Financial Times, Yahoo Finance та інші. Ці ресурси забезпечують високу якість, актуальність і достовірність інформації, що є важливим для моделювання впливу подій на фінансові ринки. Новини охоплювали широкий спектр тем, зокрема економічні звіти, аналітику фінансових ринків, макроекономічні прогнози, а також важливі політичні події, які можуть впливати на динаміку валютного курсу. Зібраний новинний контекст був попередньо оброблений для видалення шуму, таких як дублікат новин або несуттєва інформація, що дозволило покращити точність моделі.

На початковому етапі моделі прогнозування були навчені виключно на часових рядах історичних курсів CHF/USD. Для навчання моделей було використано 80% всіх доступних даних. Базова модель LSTM продемонструвала високу здатність до виявлення закономірностей у даних, таких як сезонні коливання та короткострокові тренди (рис. 3).

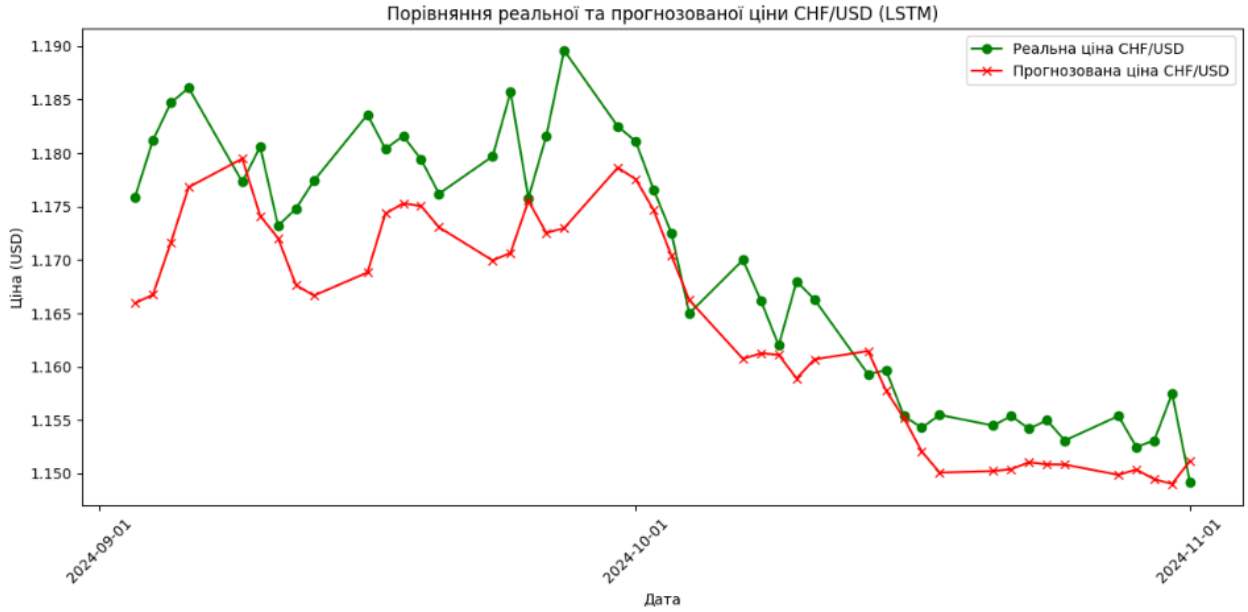


Рисунок 3. Прогнозування на основі тільки історичних даних

Модель Word2Vec показала покращення прогнозування у порівнянні з базовою LSTM. Проте її ефективність залежала від того, наскільки новини були прямо пов'язані з економічною ситуацією чи валютними ринками (рис. 4).

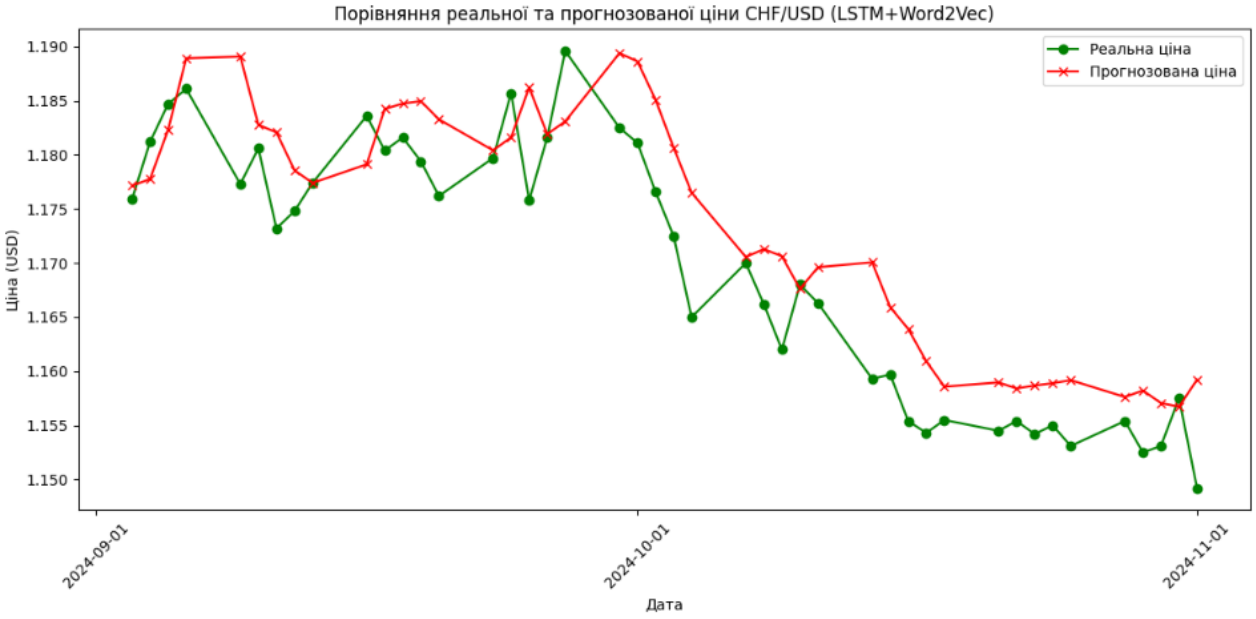


Рисунок 4. Прогнозування з залученням новинних даних з аналізатором Word2Vec

GloVe дозволив врахувати глобальні зв'язки між словами, що позитивно вплинуло на розуміння загальних ринкових тенденцій. Ця модель забезпечила краще узагальнення впливу новинного контексту (рис. 5).

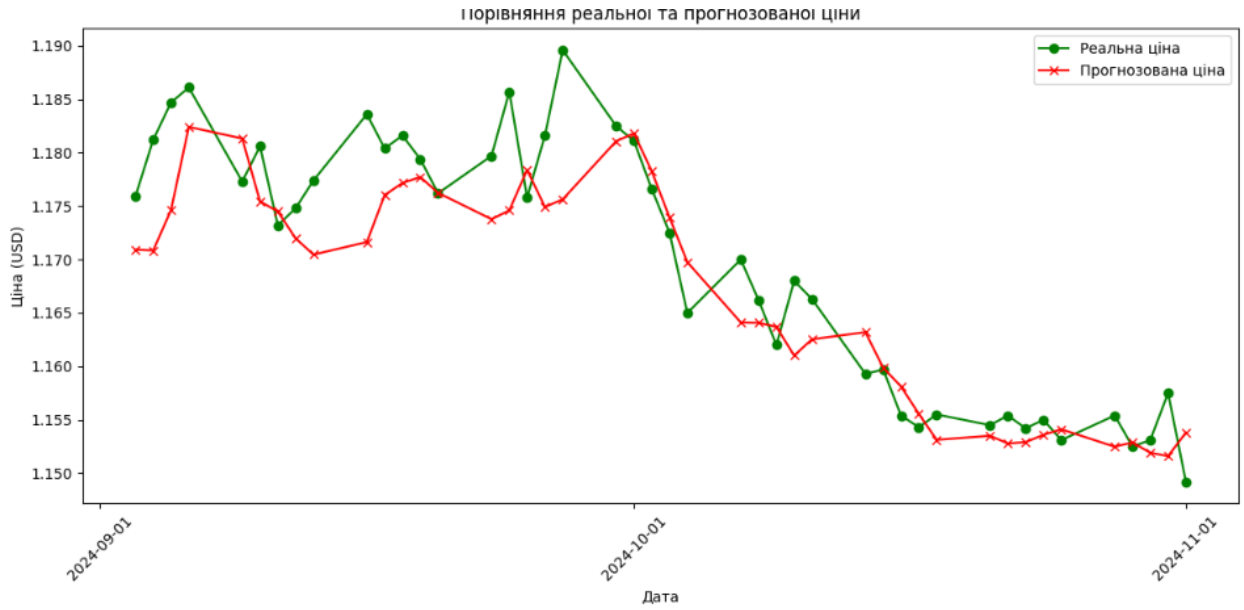


Рисунок 5. Прогнозування з залученням новинних даних з аналізатором GloVe

GPT забезпечив найкращі результати завдяки здатності обробляти складні текстові залежності та враховувати широкий контекст новин. Це дозволило моделі найбільш точно оцінювати вплив подій на валютний курс (рис. 6).

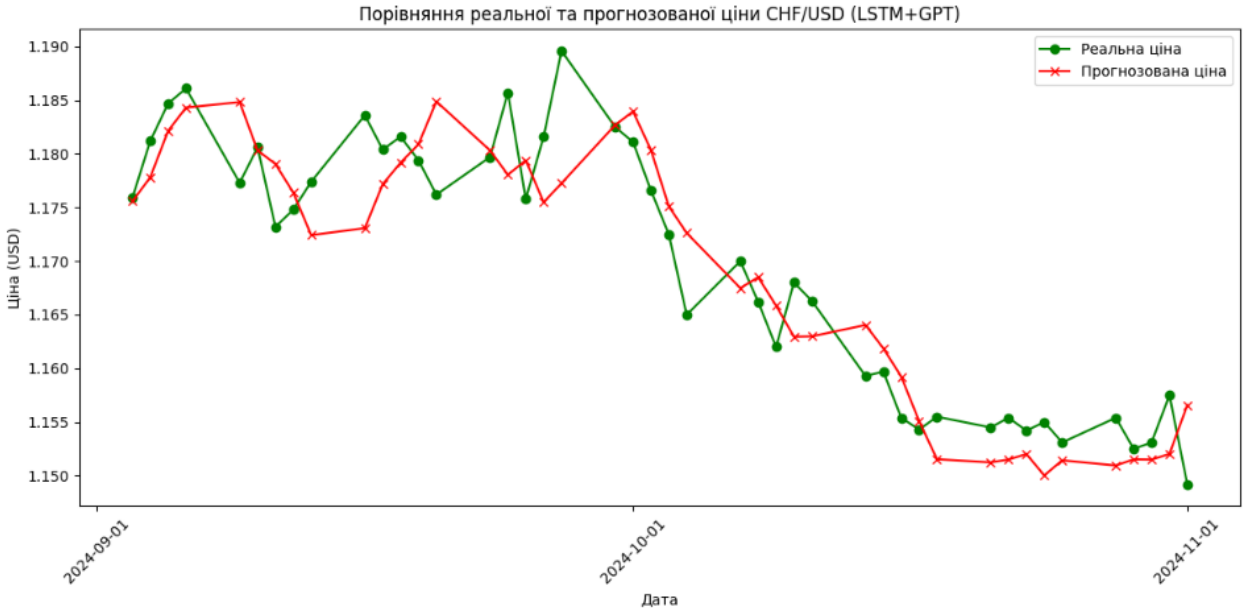


Рисунок 6. Прогнозування з залученням новинних даних з аналізатором GPT

Таблиця 1. Характеристики моделей

Модель	MSE	MAE	RMSE
LSTM	5.195e-05	0.00579	0.00721
LSTM+Word2Vec	3.594e-05	0.00507	0.00599
LSTM+GloVe	2.683e-05	0.00398	0.00518
LSTM+GPT	2.227e-05	0.00387	0.00472

Порівняння результатів, наведених у таблиці 1, показує, що інтеграція текстових аналізаторів із базовою моделлю LSTM суттєво знижує похибки прогнозування. Найбільшу похибку продемонструвала базова модель LSTM, яка працювала лише з історичними даними, що підтверджує важливість врахування зовнішніх факторів, таких як новини. З додаванням текстового аналізу продуктивність моделей покращилась: Word2Vec знизив похибки в порівнянні з базовою LSTM, проте він дещо поступається іншим методам через лінійність векторного представлення. GloVe продемонстрував ще кращі результати, особливо з точки зору MAE, завдяки врахуванню глобальних залежностей у тексті. Найнижчі значення похибок забезпечила модель GPT, яка завдяки своїй трансформерній архітектурі змогла найефективніше обробити складні текстові залежності. Це свідчить про перевагу глибоких контекстуальних моделей у задачах інтегрованого прогнозування.

4. ВИСНОВКИ

Проведені експерименти підтвердили, що базова модель LSTM, яка працює лише з історичними даними, демонструє обмежену точність. Інтеграція текстових аналізаторів із цією моделлю дозволила знизити похибки завдяки додатковій інформації про економічні та політичні події, представлені в новинах. Аналізатор Word2Vec покращив точність прогнозування, проте його лінійний підхід до побудови векторів обмежив результати. GloVe продемонстрував значні переваги, враховуючи глобальні зв’язки між словами, що дозволило краще узагальнювати новинний контекст. Найкращі результати забезпечила модель GPT завдяки здатності до глибокого розуміння тексту, аналізу контексту та виявлення складних залежностей.

Робота показала, що поєднання часових рядів і текстових даних створює потужний інструмент для прогнозування економічних процесів. Запропонований підхід дозволяє враховувати як внутрішні ринкові патерни, так і зовнішні впливи, зокрема політичні та економічні новини. Отримані результати відкривають перспективи для подальших досліджень, таких як оптимізація гіперпараметрів, розширення спектра текстових джерел та використання інших трансформерних моделей. Це підтверджує, що комплексний аналіз даних із застосуванням сучасних методів машинного навчання є ефективним рішенням для прогнозування економічних процесів у реальному часі.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. S. Hochreiter, Y. Bengio, and J. Schmidhuber, “Gradient flow in recurrent nets: the difficulty of learning long-term dependencies”. [Online]. Available: http://www.bioinf.jku.at/ publications/older/ch7.pdf . Accessed on: Dec. 12, 2018.

2. R. Hyndman, G. Athanasopoulos, Forecasting: Principles and Practice. Melbourne, Australia: OTexts, 2013.

3. F. Chollet, and J. Allaire, Deep Learning with R. New York, USA: Manning, 2018.

4. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, USA: MIT Press, 2016.

УДК 378.147

СИСТЕМА СТВОРЕННЯ ЕТАПІВ ДИПЛОМНОГО ПРОЕКТУВАННЯ

Омельченко Г.О.[1], Бандурка О.І.[2], Дацюк О.А.[3]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] glebomelchenko911@gmail.com, [2] o.i.bandurka@ukr.net, [3] datsiuk@gmail.com

У даній роботі розроблено систему для оптимізації етапів дипломного проектування, що дозволяє студентам обирати тему під конкретний напрям або пропонувати власну, а також обирати наукового керівника. Система інтегрує інструменти спілкування та обміну файлами, сприяючи ефективній взаємодії й підвищенню якості підготовки студентів.

Ключові слова: дипломне проектування, система створення етапів дипломного проектування, вибір теми, вибір наукового керівника, обмін файлами.

Дипломне проектування є одним із найважливіших етапів навчального процесу для студентів вищих навчальних закладів. Воно вимагає значної організації, зокрема розподілу завдань, контрольних точок і взаємодії між студентом і науковим керівником. Однак у сучасних реаліях цей процес часто зустрічається з низкою проблем, пов'язаних із неефективною комунікацією, нестачею централізованого управління та труднощами в моніторингу прогресу студентів. Для вирішення цих питань пропонується створити систему "Система створення етапів дипломного проектування", яка автоматизує та оптимізує цей процес.

Основною метою розробки системи є автоматизація етапів дипломного проектування, що дозволить викладачам та студентам ефективніше керувати проектами [2], покращить комунікацію між ними та підвищить загальний рівень організації роботи. Система повинна надавати інструменти для створення та управління дипломними проектами, що включатимуть різні етапи – від вибору теми до надсилання готових звітів та щоденників.

Першим кроком у роботі системи є розробка етапів проекту. Викладачі повинні мати можливість створювати теми під напрямками для дипломного проекту – це означає що в системі будуть присутні такі речі як вибір теми, додавання матеріалів, написання звіту, його перевірка, редагування та фінальна здача. Кожен етап повинен бути чітко визначений із вказаними дедлайнами та завданнями, які студент повинен виконати.

Другий важливий аспект системи – це керування процесом виконання дипломного проекту. Викладачі можуть призначати студентам конкретні завдання для кожного етапу. Наприклад, під час початкового етапу студент може отримати завдання на вибір теми, підготовку пропозиції та плану дослідження. Кожен етап повинен містити інструкції та вимоги щодо виконання, а також інструменти для завантаження документів та інших матеріалів, необхідних для виконання завдань.

Крім того, система повинна забезпечувати моніторинг прогресу студентів. Викладачі мають можливість контролювати виконання завдань на кожному етапі, перевіряти матеріали, відслідковувати стан готовності магістерської дисертації. Це дозволить уникнути проблем з несвоєчасним виконанням завдань, що часто є причиною затримок у дипломному процесі.

Студенти, зі свого боку, також можуть бачити свій прогрес і розуміти, на якому етапі вони знаходяться, що допоможе їм краще планувати час і виконувати завдання вчасно.

Важливим аспектом роботи системи є забезпечення комунікації між викладачами та студентами. Система повинна мати функції обміну повідомленнями, що дозволить студентам швидко зв'язуватися зі своїми науковими керівниками, ставити запитання або отримувати зворотній зв'язок щодо виконаних завдань. Окрім того, викладачі можуть коментувати завантажені студентами матеріали, вносити правки або надавати рекомендації щодо подальшої роботи.

Не менш важливою функцією системи є генерація звітів про дипломне проектування.

Запровадження такої системи дозволить значно покращити організацію дипломного проектування, зробити його прозорим та більш контрольованим як для студентів, так і для викладачів. Автоматизація рутинних процесів, зокрема розподілу завдань, моніторингу прогресу та обміну інформацією, значно зменшить адміністративні витрати наукових керівників і дозволить їм більше зосередитися на змістовній частині дипломного проекту.

Система створення етапів дипломного проектування також допоможе студентам краще організувати свою роботу, розподіляти час та відповідальніше підходити до виконання завдань. Наявність чітких дедлайнів, завдань та можливості отримувати оперативний зворотний зв'язок дозволить студентам уникнути затримок та проблем із захистом магістерської дисертації.

Загалом, така система сприятиме підвищенню якості підготовки дипломних проектів, що позитивно вплине на загальний рівень знань студентів та поліпшить процес їхньої підготовки до реальної роботи в професійній сфері. Автоматизація процесів дипломного проектування — це важливий крок у напрямку модернізації вищої освіти та підвищення ефективності навчання.

Розроблена система містить кілька ключових компонентів. Система створення етапів дипломного проектування спрямована на оптимізацію та автоматизацію процесів підготовки студентів до виконання дипломних робіт. Вона враховує потреби різних категорій користувачів, зокрема студентів, викладачів і адміністрації, та забезпечує зручний інструментарій для взаємодії між ними (рис. 1).

Одним із ключових елементів системи є забезпечення гнучкого вибору тем для студентів. Вони можуть обирати тему з наявного переліку, запропонованого викладачами, або подавати власні варіанти. Це дає змогу максимально врахувати індивідуальні інтереси студента та його професійні цілі. Крім того, система дозволяє закріплювати наукових керівників, забезпечуючи прозорий та ефективний процес вибору.

Для викладачів система створює можливості управління темами дипломних робіт, контролю виконання завдань студентів, а також тісної взаємодії зі студентами через комунікаційні інструменти. Завдяки вбудованому функціоналу обміну документами у форматі Word, викладачі можуть перевіряти проміжні версії звітів і щоденників, а також давати рекомендації щодо їх удосконалення.

Адміністрація отримує інструменти для організації та координації роботи системи. Вона може керувати даними про групи, користувачів, напрями підготовки, етапи проектування та навчальні матеріали. Крім того, адміністративні працівники можуть здійснювати документообіг, надсилати повідомлення студентам і викладачам, а також формувати звіти щодо обраних тем і студентів без тем.

Гнучка структура системи забезпечує можливість відслідковування календарних планів студентів, управління версіями документів і спрощує організацію зворотного зв’язку між усіма учасниками процесу. Це не лише оптимізує роботу студентів і викладачів, але й сприяє підвищенню якості дипломного проектування, впроваджуючи сучасні цифрові технології.

Рисунок 1. Створені компоненти системи

Завдяки інтеграції інструментів комунікації, управління даними та документообігу система створює сприятливі умови для організації самостійної роботи студентів у цифровому середовищі. Вона спрямована на спрощення виконання кожного етапу дипломного проектування та забезпечення якісної взаємодії між студентами, викладачами й адміністраторами.

У рамках розробки сучасної освітньої платформи було реалізовано комплексну архітектуру бази даних, що забезпечує ефективне управління навчальним процесом. Представлена модель даних базується на реляційному підході з використанням принципів нормалізації та оптимізації структур даних.

Центральним компонентом архітектури є сутність "Користувачі" (Users), що реалізує багатокористувацьку модель взаємодії. Кожен користувач характеризується унікальним ідентифікатором та набором атрибутів, що включають аутентифікаційні дані (логін, пароль), контактну інформацію (електронна адреса, контактний номер) та персональні дані. Важливим аспектом є імплементація механізму типізації користувачів, що дозволяє диференціювати ролі та рівні доступу в системі (рис. 2).

Архітектурне рішення передбачає групову модель організації навчального процесу через сутність "Групи" (Groups). Реалізовано багато-до-багатьох зв'язок між користувачами та групами через проміжну таблицю "Учасники груп" (Group Members), що забезпечує гнучкість у формуванні навчальних колективів. Кожна група має власні атрибути, включаючи назву, опис та часові характеристики функціонування.

Особлива увага приділена темпоральним аспектам системи через впровадження календарного модуля. Сутність "Календарний план" (Calendar Plan) забезпечує планування та відстеження навчальних активностей з чітко визначеними часовими рамками. Кожна активність характеризується датами початку та завершення, що дозволяє ефективно організовувати навчальний процес та контролювати дотримання дедлайнів.

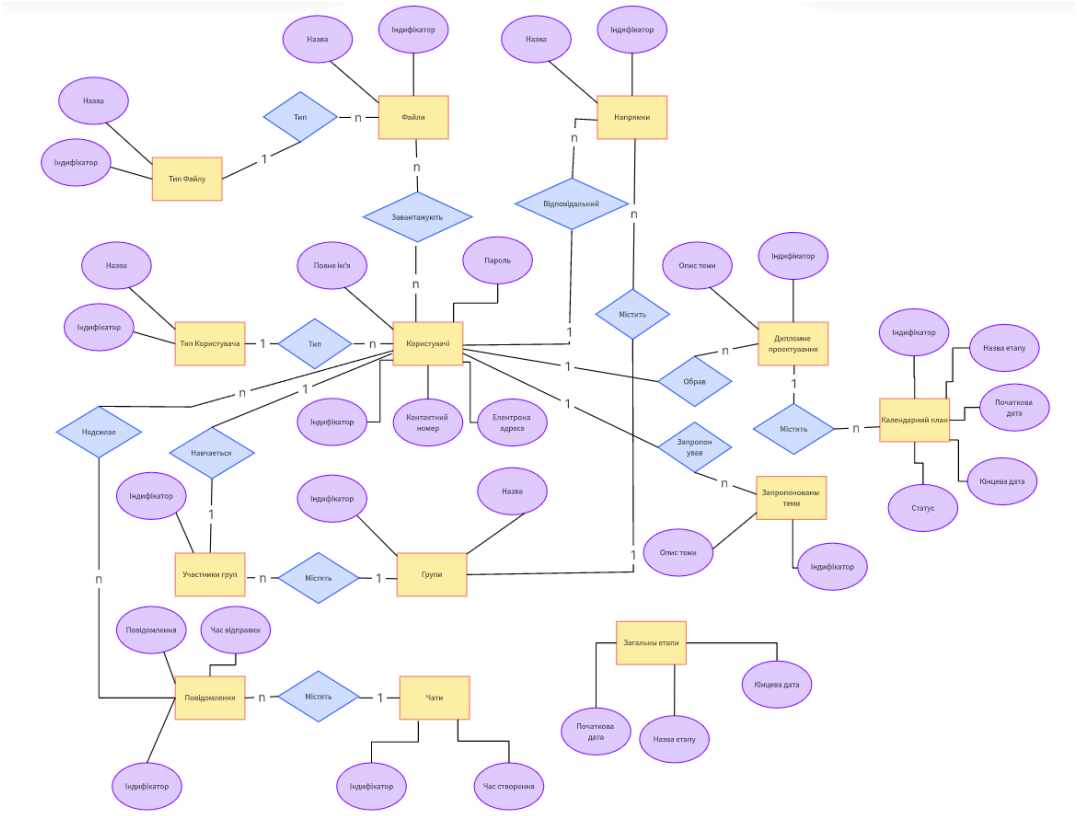


Рисунок 2. ER діаграма бази даних

Контентна складова системи реалізована через взаємопов'язані сутності управління навчальними матеріалами. Впроваджено механізм категоризації контенту, що дозволяє структурувати навчальні матеріали за темами та підтемами. Кожен елемент контенту супроводжується метаданими, що включають опис, тип матеріалу та статус.

Важливим компонентом є система відстеження прогресу та статусів. Реалізовано механізм фіксації станів для різних елементів системи, що дозволяє моніторити прогрес користувачів та ефективність навчального процесу. Кожна зміна статусу супроводжується часовою міткою та ідентифікацією ініціатора змін.

Архітектура передбачає розширену систему аудиту через впровадження механізму логування змін. Кожна сутність містить службові поля для відстеження часу створення та модифікації записів, що забезпечує можливість аналізу історії змін та відновлення даних при необхідності.

В контексті масштабованості система спроектована з урахуванням можливості горизонтального розширення. Використання зовнішніх ключів та індексів оптимізовано для забезпечення ефективної роботи при збільшенні обсягу даних. Передбачено можливість додавання нових типів контенту та активностей без суттєвої модифікації існуючої структури.

Особлива увага приділена аспектам безпеки даних. Впроваджено механізми валідації вхідних даних на рівні бази даних, реалізовано систему обмежень цілісності та передбачено можливість шифрування чутливих даних.

Представлена архітектура успішно впроваджена та протестована в реальних умовах експлуатації, демонструючи високу надійність та ефективність у забезпеченні навчального процесу. Модульний підхід та чітка структуризація компонентів забезпечують можливість подальшого розвитку системи відповідно до зростаючих потреб користувачів та еволюції освітніх технологій.

Інноваційним аспектом розробленої архітектури є імплементація механізму адаптивного навчання через систему динамічних зв'язків між контентом та користувацькою активністю. Кожна взаємодія користувача з системою аналізується та зберігається в спеціалізованих агрегаційних таблицях, що дозволяє формувати персоналізовані навчальні траєкторії. Система автоматично адаптує складність матеріалу та темп навчання на основі прогресу користувача, його успішності та патернів взаємодії з контентом. Такий підхід забезпечує оптимальне засвоєння матеріалу та підвищує ефективність навчального процесу. Важливим елементом цього механізму є впроваджена система метрик та індикаторів, що дозволяє кількісно оцінювати ефективність навчального процесу та своєчасно вносити корективи в навчальні програми. Збір та аналіз цих даних також надає цінну інформацію для подальшого вдосконалення освітньої платформи та розробки нових функціональних можливостей.

Наукова новизна полягає у розробці цифрової системи, яка вперше інтегрує індивідуальний підхід до організації етапів дипломного проектування. Система забезпечує автоматизований вибір теми, наукового керівника та оптимізує взаємодію студентів і викладачів у цифровому середовищі, сприяючи підвищенню якості підготовки.

Практичне значення дослідження полягає у створенні системи, що оптимізує етапи дипломного проектування та сприяє самостійності студентів у виборі теми і наукового керівника. Вона забезпечує зручну взаємодію і обмін файлами між студентами та керівниками, підвищуючи ефективність і якість організації проектної діяльності.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Андріяш В. К. Дейт. Введення в систему бази даних. М.: Наука, 1980, с. 463.

2. Зінчук Грассіг І., Кемерер К. Основи управління програмними проектами. М.: Видавництво ІД «Вільямс», 2004. — 432 с.

3. Гель Коваль Т.І. Автоматизація процесів управління проектами у закладах вищої освіти. Київ: КНУ, 2023. — 286 с.

4. Пономаренко В.С. Проектування інформаційних систем. Харків: ХНЕУ, 2023. — 442 с.

5. Управління проектами в умовах цифрової трансформації / За ред. С.Д. Бушуєва. Київ: КНУБА, 2023. — 386 с.

ОПТИМІЗАЦІЯ ВИБОРУ ІНВЕСТИЦІЙНОГО ПОРТФЕЛЮ

Павлюк С.В.[1], Мілявський Ю.Л.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] sunzil@gmail.com

Основною метою дослідження є розробка математичної моделі управління інвестиційним портфелем, яка інтегрує витрати на транзакції, обмеження ліквідності, похибки оцінювання та інші практичні фактори для підвищення ефективності. У роботі досліджуються класичні середньо-дисперсійні моделі, модифіковані підходи до врахування транзакційних витрат, факторні моделі ризику, а також інтеграція цих методів у загальну стратегію оптимізації. Проведено емпіричне тестування запропонованої моделі на основі даних про галузеві портфелі, індекси фондових ринків та факторні портфелі. Для оцінювання переваг моделі використовуються такі показники ефективності: коефіцієнт Шарпа, еквівалент гарантованого доходу (CEQ) та оборотність портфеля.

Ключові слова: оптимізація портфеля, середньо-дисперсійна модель, транзакційні витрати, ліквідність, факторні моделі, коефіцієнт Шарпа.

1. ВСТУП

В умовах сучасної економіки, яка характеризується високим ступенем нестабільності та невизначеності, ефективне управління інвестиційними портфелями набуває критичного значення. Вибір оптимального портфеля є ключовим завданням інвесторів, яке вимагає врахування не лише прибутковості, а й ризику, витрат на транзакції, а також обмежень ліквідності. Класична теорія портфеля, розроблена Гаррі Марковіцем, визначає процес побудови оптимального портфеля як задачі мінімізації ризику за заданого рівня доходності. Проте реальні ринки значно складніші, ніж передбачено теоретичними моделями. Фактори, такі як витрати на транзакції, ліквідність активів, регуляторні обмеження, суттєво впливають на ефективність прийнятих рішень.

2. ПОСТАНОВКА ЗАДАЧІ

Модифікувати середньо-дисперсійний підхід, включивши в нього витрати на транзакції, обмеження на розміри позицій та параметри ліквідності. Визначити аналітичний вигляд функції корисності, яка враховує фактори витрат та ризику, а також інтегрує модель оцінки коваріаційної матриці. Оцінити ефективність моделі за допомогою коефіцієнта Шарпа, еквівалента гарантованого доходу (CEQ) та інших ключових показників. Порівняти ефективність запропонованої моделі з класичними підходами.

3. РОЗРОБКА АНАЛІТИЧНОЇ МОДЕЛІ

Одним з обмежень класичної моделі Марковіца є ігнорування транзакційних витрат. Для реалістичної оптимізації портфеля необхідно враховувати торгові витрати, які можуть

включати як пропорційні (спред між ціною купівлі та продажу), так і індуковані витрати. Функція корисності з урахуванням таких витрат може бути представлена наступним чином:

$$U(w) = f^T w - \frac{\lambda}{2} w^T C^2 w - \gamma c_T (b + s) - \delta d_T (b + s)^2$$

де λ, γ, δ — коефіцієнти, які визначають вагу ризику, пропорційних і квадратичних витрат у функції корисності; b і s – кількість куплених і проданих активів відповідно, c – пропорційні витрати, а d — квадратичні витрати.

Оптимізація портфеля з урахуванням витрат і обмежень перетворюється на задачу квадратичного програмування, де метою є максимізація функції корисності з урахуванням витрат на торгівлю. У випадку з обмеженнями на розмір угод і позицій оптимізаційна задача записується як:

$$max_{b,s} \left(f^T (b - s) - c^T (b + s) - (b - s)^T (2C_2 + d)(b - s)\right)$$

при обмеженнях на розмір покупок і продажів:

$$0 \leq b_k \leq \min(B_k, u_k - (w_0)_k)$$

$$0 \leq s_k \leq \min(S_k, (w_0)_k - l_k)$$

4. МЕТОДИ ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ

Для того щоб перевірити ефективність запропонованого методу, проаналізуємо чотири критерії та порівняємо з існуючими моделями для оптимізації вибору інвестиційного портфелю.

Коефіцієнт Шарпа (Sharpe Ratio) використовувався для порівняння прибутковості портфелів з урахуванням ризику. Коефіцієнт Шарпа визначається як співвідношення середнього перевищення прибутковості над безризиковою ставкою до стандартного відхилення доходностей. Це допомагає визначити, наскільки ефективною є кожна модель у створенні портфеля з високим співвідношенням доходності до ризику.

Формально, коефіцієнт Шарпа в межах вибірки для стратегії k визначається як:

$$\widehat{SR}_k^{IS} = \frac{Mean_k}{Std_k} = \frac{\hat{\mu}_k^{IS\top} \widehat{w}_k}{\sqrt{\widehat{w}_k^\top \widehat{\sum}_k^{IS} \widehat{w}_k}}$$

Показник «еквівалент гарантованого доходу» (Certainty-Equivalent Return, CEQ) дозволяє оцінити, яку фіксовану дохідність готовий обрати інвестор замість ризикового портфеля, сформованого кожною моделлю. Цей показник показує той рівень гарантованого доходу, який інвестор готовий прийняти замість ризикованої інвестиції.

Формально, ми розраховуємо дохідність CEQ для стратегії k за формулою:

$$CEQ_k = \hat{\mu}_k - \frac{\gamma}{2} \hat{\sigma}_k^2$$

де $\hat{\mu}_k$ і $\hat{\sigma}_k^2$ – це середнє значення і дисперсія позавибіркових надлишкових доходностей для портфелів k, а γ – коефіцієнт ризикової антипатії інвестора (рівень ризику, якого він уникає).

Оборотність портфеля відображає, наскільки часто доводиться проводити торги для підтримки оптимального складу портфеля. Показник важливий для розуміння обсягу витрат, пов’язаних із реалізацією кожної стратегії, оскільки часті торги можуть призводити до підвищення витрат, що, у свою чергу, впливає на реальну прибутковість портфеля.

Щоб зрозуміти обсяг торгівлі, необхідний для реалізації кожної стратегії портфеля, було обчислено оборот портфеля, що визначається як середнє значення суми абсолютних значень операцій для всіх N доступних активів:

$$\text{Оборот} = \frac{1}{T-M}\sum_{t=1}^{T-M}\sum_{j=1}^{N}\left|\widehat{w}_{k,j,t+1} - \widehat{w}_{k,j,t+}\right|$$

де $\widehat{w}_{k,j,t}$ – це вага активу j у портфелі в момент tдля стратегії k; $\widehat{w}_{k,j,t+}$ – вага портфеля до перебалансування в момент $t+1$; а $\widehat{w}_{k,j,t+1}$ — цільова вага портфеля на момент $t+1$ після перебалансування. Показник характеризує частоту торгів у портфелі, тобто, як часто портфель перебалансовується шляхом купівлі-продажу активів. Для стратегії-еталона mv ми наводимо її абсолютний оборот, а для всіх інших стратегій — їх оборот відносно еталонної стратегії.

Крім значення обороту для кожної моделі, наводиться економічний показник, що демонструє, як пропорційні витрати на операції, викликані оборотом, впливають на доходність конкретної стратегії. Пропорційну вартість транзакції встановлено на рівні 50 базисних пунктів за операцію, згідно з припущенням [1], та на основі досліджень витрат на операції для окремих акцій на NYSE [2–4].

Для кожної моделі обчислюється втрата доходності відносно стратегії mv-c. Втрата доходності визначається як додатковий дохід, необхідний для того, щоб стратегія *k* забезпечила таку ж ефективність, як і модель mv-c за коефіцієнтом Шарпа. Для обчислення втрати доходності за місяць припускається, що μ_{ew} та σ_{ew} – це середнє значення та волатильність позавибіркових чистих доходів стратегії mv-c за місяць, а μ_k та σ_k – відповідні показники для стратегії *k*. Формула для обчислення втрати доходності стратегії *k* має вигляд:

$$return - loss_k = \frac{\mu_{ew}}{\sigma_{ew}} \times \sigma_k - \mu_k$$

5. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Таблиця 1. Коефіцієнти Шарпа для емпіричних даних

Модель	S&P Sectors	Галузеві портфелі	Міжнародні	MKT/SMB/HML	FF-1 фактор	FF-4 фактор
	N = 11	N = 11	N = 9	N = 3	N = 21	N = 24
mv (класична)	0.3848	0.2124	0.2090	0.2851	0.5098	0.5364
mv (з урахуванням похибки)	0.0794	0.0679	-0.0332	0.2186	0.0128	0.1841
min	0.0820	0.1554	0.1490	0.2493	0.2778	-0.0183
vw (camp)	0.1444	0.1138	0.1239	0.1138	0.1138	0.1138
mv – c	0.0892	0.0678	0.0848	0.1084	0.1977	0.2024

Згідно з Таблицею 1, класична модель середньо-дисперсійної оптимізації (mv) показує найвищі результати у більшості випадків, наприклад, для набору даних «FF-4 фактор» коефіцієнт Шарпа становить 0.5364. Однак врахування похибки оцінювання суттєво знижує ефективність моделі: для того ж набору даних «FF-4 фактор» коефіцієнт Шарпа падає до 0.1841. Це підкреслює важливість врахування факторів ризику та похибок.

Оптимізована модель (mv-c) демонструє стабільні результати, наприклад, 0.1977 для набору «FF-1 фактор» і 0.2024 для «FF-4 фактор». Це свідчить про те, що врахування витрат та обмежень дозволяє значно покращити стабільність портфелів.

Таблиця 2. Еквівалент гарантованого доходу (CEQ)

Модель	S&P Sectors	Галузеві портфелі	Міжнародні	MKT/SMB/HML	FF-1 фактор	FF-4 фактор
	N = 11	N = 11	N = 9	N = 3	N = 21	N = 24
mv (класична)	0.0478	0.0106	0.0096	0.0047	0.0300	0.0304
mv (з урахуванням похибки)	0.0031	-0.7816	-0.1365	0.0045	-2.7142	-0.0829
min	0.0024	0.0052	0.0054	0.0039	0.0100	-0.0002
vw (camp)	0.0053	0.0042	0.0044	0.0042	0.0042	0.0042
mv – c	0.0040	0.0023	0.0032	0.0030	0.0090	0.0075

Як показано у Таблиці 2, класична модель mv забезпечує найвищу еквівалентну дохідність у теоретичних умовах (0.0478 для «S&P Sectors»). Однак при врахуванні похибок, наприклад, для «FF-1 фактор», дохідність стає від'ємною (-2.7142), що свідчить про суттєві втрати.

Модель mv-c, навпаки, демонструє стабільніші значення CEQ у всіх наборах, наприклад, 0.0090 для «FF-1 фактор» і 0.0075 для «FF-4 фактор». Це підкреслює її адаптивність у реальних умовах ринку

Таблиця 3. Оборотність портфелів для емпіричних даних

Модель	S&P Sectors	Галузеві портфелі	Міжнародні	MKT/SMB/HML	FF-1 фактор	FF-4 фактор
	N = 11	N = 11	N = 9	N = 3	N = 21	N = 24
mv (класична)	-	-	-	-	-	-
mv (з урахуванням похибки)	38.99	606594.36	4475.81	2.83	10466.10	3553.03
min	6.54	21.65	7.30	1.11	45.47	6.83
vw (camp)	0	0	0	0	0	0
mv – c	4.53	7.17	7.23	4.12	17.53	13.82

Оборотність портфелів у Таблиці 3 свідчить про рівень витрат, необхідних для підтримки оптимальної структури портфеля. Модель mv (з урахуванням похибки) демонструє високу оборотність, наприклад, 606594.36 для «Галузевих портфелів», що призводить до значних транзакційних витрат.

Модель mv-c показує значно нижчу оборотність, наприклад, 4.53 для «S&P Sectors», що забезпечує її ефективність у зниженні витрат на торгівлю та підтримці стабільності.

Таблиця 4. Втрата доходності відносно mv-c

Модель	S&P Sectors	Галузеві портфелі	Міжнародні	MKT/SMB/HML	FF-1 фактор	FF-4 фактор
	N = 11	N = 11	N = 9	N = 3	N = 21	N = 24
mv (класична)	-	-	-	-	-	-
mv (з урахуванням похибки)	0.0145	23.8504	1.1689	0.0003	7.4030	1.5740
min	0.0048	0.0015	0.0000	-0.0004	-0.0008	0.0024
vw (camp)	-0.0001	0.0037	0.0012	0.0157	0.0021	0.0028

Модель mv (з урахуванням похибки) у Таблиці 4 демонструє суттєві втрати доходності, наприклад, 23.8504 для «Галузевих портфелів». Модель mv-c, навпаки, має мінімальні втрати, що свідчить про її перевагу у реальних умовах із високими витратами на торгівлю.

6. ВИСНОВКИ

Проведені дослідження підтверджують, що класичний середньо-дисперсійний підхід Гаррі Марковіца є ефективним лише у теоретичних умовах, але суттєво втрачає свою практичну цінність без урахування реальних обмежень, таких як транзакційні витрати та ліквідність активів. Модифіковані моделі, які враховують ці аспекти, забезпечують кращу адаптивність до сучасних ринкових умов.

Включення пропорційних і квадратичних транзакційних витрат до функції корисності дозволяє уникнути надмірної торгівлі та знизити загальні витрати. Це позитивно впливає на стабільність портфеля, зокрема завдяки введенню обмежень на обсяги торгів і позицій активів.

Експериментальні дані продемонстрували, що модель із врахуванням витрат (mv-c) має значно нижчий рівень оборотності порівняно з класичною моделлю. Це дозволяє знизити операційні витрати та забезпечити більш стабільний дохід, особливо в умовах високих витрат на транзакції.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Balduzzi, P., & Lynch, A. W. (1999). Transaction Costs and Predictability: Some Utility Cost Calculations. The Journal of Financial and Quantitative Analysis, 34(2), 209–225.

2. Stoll, H. R., & Whaley, R. E. (1983). Transaction Costs and the Small Firm Effect. Journal of Financial Economics, 12(1), 57–79.

3. Bhardwaj, R. K., & Brooks, L. D. (1992). The January Anomaly: Effects of Low Share Price, Transaction Costs, and Bid-Ask Bias. The Journal of Finance, 47(2), 553–575.

4. Lesmond, D. A., Ogden, J. P., & Trzcinka, C. A. (1999). A New Estimate of Transaction Costs. The Review of Financial Studies, 12(5), 1113–1141.

РОЗРОБКА ІНСТРУМЕНТАРІЮ АНАЛІЗУ ДАНИХ ДЛЯ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ ТЕКСТОВИХ ПОВІДОМЛЕНЬ

Паутинка О.В.[1], Савастьянов В.В.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] pautynka.oleksandra@lll.kpi.ua

Ефективний аналіз текстових даних набуває дедалі більшої важливості для прийняття стратегічних рішень у різних галузях. В цій роботі досліджуються моделі Bi-LSTM та Gemma для аналізу текстових повідомлень, орієнтованих на екстракцію фактів, що дозволяє ефективно виділяти релевантні дані для оцінки ризиків. Запропоновано багаторівневу систему ранжування загроз, яка базується на якісно-кількісному порівнянні та просторово-динамічному оцінюванні, що враховує історичні дані та просторові взаємозв'язки між подіями, підвищуючи точність аналізу загроз. Розроблена архітектура експертної системи забезпечує надійне зберігання й обробку даних та включає в себе експертну систему для гнучкої оцінки загроз. Такий підхід оптимізує процеси аналізу великих обсягів даних, підвищуючи якість прийняття рішень у сферах безпеки, управління інфраструктурою та маркетингу, забезпечуючи систематичний підхід до управління інформацією.

Ключові слова: Bi-LSTM, модель Gemma, NLP, текстові повідомлення, підтримка прийняття рішень, оцінка ризиків, ранжування загроз.

1. ВСТУП

У сучасному світі обсяги текстових даних стрімко зростають, створюючи нові виклики для їх аналізу та ефективного використання. Повідомлення, звіти, документи та інші форми текстової інформації часто містять критично важливі дані, які необхідно швидко ідентифікувати для прийняття обґрунтованих рішень. Стрімке поширення інформаційно-комунікаційних технологій, економічне зростання та суспільна трансформація спричинили появу величезної кількості інформаційних потоків, які людина не завжди здатна опрацювати самостійно [1].Тому автоматизація аналізу текстових даних стає критично важливою для багатьох сфер, зокрема безпеки та управління кризовими ситуаціями.

Проте більшість текстових даних є неструктурованими й містять складні конструкції природної мови, що ускладнює вилучення конкретної інформації для подальшого аналізу. Це стимулює розвиток досліджень у сфері глибинного навчання та класифікації текстів на основі природної мови [2]. У цьому контексті особливої значущості набуває інформаційна екстракція – автоматичне вилучення структурованої інформації з неструктурованих текстів, що є однією з ключових складових обробки природної мови (NLP) [3]. Традиційні методи класифікації текстів, засновані на словниках і базових методах машинного навчання, були замінені потужними методами глибинного навчання, серед яких моделі LSTM (long short-term memory) займають особливе місце завдяки здатності обробляти послідовності текстів із тривалими залежностями [4].

Саме тому, використання в цій роботі сучасних моделей обробки природної мови, таких як Bi-LSTM і Gemma, дозволяє ефективно аналізувати текстові повідомлення, враховуючи не лише локальні характеристики даних, але й їх просторово-часові зв’язки. Також пропонується інноваційний підхід до аналізу текстових повідомлень, який об'єднує якісно-кількісне порівняння та метод просторово-динамічного оцінювання. Ця методика враховує історичні події, їхній вплив на поточну ситуацію та можливість розширення зон ризику, що є важливим для ранжування загроз і оцінки ризиків.

Це дослідження спрямоване на вдосконалення алгоритмів аналізу текстових даних та створення ефективної системи, здатної враховувати просторові та часові аспекти подій. Запропоновані підходи є важливими для розв'язання сучасних проблем інформаційного управління та прийняття рішень у критичних ситуаціях

2. СИСТЕМНИЙ ПІДХІД ДО ПОБУДОВИ ІНСТРУМЕНТАРІЮ АНАЛІЗУ ДАНИХ НА ОСНОВІ ТЕКСТОВИХ ПОВІДОМЛЕНЬ

Системний підхід до обробки текстових даних базується на інтеграції кількох методів та інструментів, що дозволяють оптимізувати процес екстракції та обробки інформації з текстових повідомлень. У сучасних умовах аналіз текстових даних відіграє критичну роль у підтримці прийняття рішень, особливо в сферах, де своєчасне отримання достовірної інформації є ключовим фактором. Основні етапи системного підходу в цій роботі продемонстровано на рис. 1:

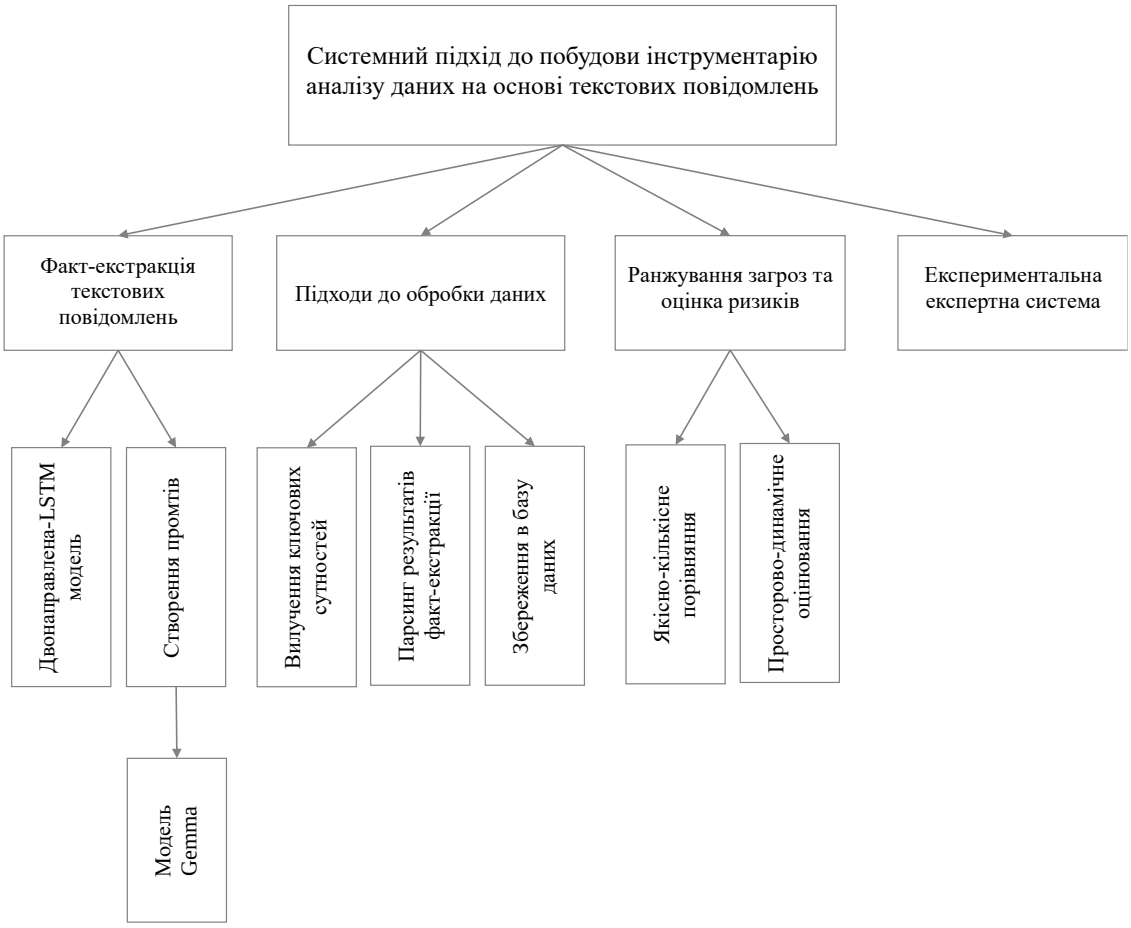


Рисунок 1. Системний підхід до побудови інструментарію аналізу даних на основі текстових повідомлень

Факт-екстракція є основним етапом аналізу текстових даних, оскільки здійснюється виділення основної інформації з великого обсягу тексту. Текстові повідомлення містять величезну кількість інформації, але ця інформація, як правило, неструктурована. Щоб зробити її придатною для подальшого аналізу, необхідно виділити ключові елементи, такі як події, локації, дати, об’єкти. На цьому етапі застосовуються дві основні моделі:

1. Bi-LSTM (двонаправлена довготривала короткочасна пам’ять) – це рекурентна нейронна мережа, яка використовується переважно для обробки природної мови. LSTM – це модель, яка може зберігати та використовувати важливу інформацію навіть у довгих послідовностях даних, завдяки чому вирішує проблему втрати контексту під час тривалого аналізу На відміну від стандартного LSTM в Bi-LSTM вхідні дані надходять в обох напрямках, тобто вона здатна використовувати інформацію з обох сторін, що робить її потужним інструментом для моделювання послідовних залежностей між словами та фразами в обох напрямках послідовності [5]. Такий підхід забезпечує більш точне розуміння контексту та дозволяє моделі визначати сутності та їхні відносини в тексті.

2. На етапі ідентифікації ключових слів важливим аспектом є створення ефективних промптів, що спрямовують роботу моделей обробки тексту та підвищують релевантність отриманих результатів (рис. 2). Формування правильних запитів дозволяє адаптувати систему під специфіку оброблюваних даних та забезпечити високу точність у процесі аналізу текстів. В основі цієї системи лежить модель Gemma – з сімейства легких відкритих моделей, розроблених Google на основі тієї ж технології, що і моделі Gemini [6]. Ця сучасна модель обробки природної мови була обрана як альтернатива Bi-LSTM завдяки своїй високій точності та здатності ефективно обробляти великі обсяги даних із мінімальною кількістю помилок. Це робить модель ідеальною для завдань, які потребують максимальної точності.

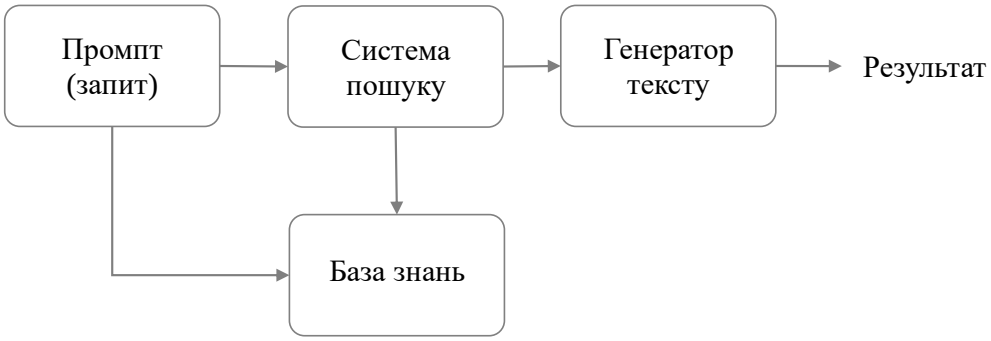


Рисунок 2. Схема роботи промптів (запитів)

Після виділення фактів із текстових повідомлень дані проходять через кілька етапів обробки для їхнього структурування та збереження. Обробка даних включає такі етапи:

1. Вилучення ключових сутностей – відбувається на основі результатів роботи моделей і передбачає виділення з тексту найважливіших елементів, таких як локації, дати, події, особи та інші об’єкти, що можуть мати значення для аналізу. Це дозволяє формувати структуру даних для подальшого аналізу.

2. Парсинг результатів факт-екстракції – на цьому етапі отримані дані перетворюються у структурований формат, придатний для зберігання в базі даних. Це дозволяє систематизувати дані, створюючи єдину базу, яка легко доступна для подальших аналітичних операцій. Структуровані дані спрощують процес подальшого аналізу та мінімізують ризик втрати важливих деталей.

3. Збереження в базу даних – після вилучення та обробки дані зберігаються у спеціальній базі даних. Цей етап дозволяє підтримувати історичні записи та забезпечує легкий доступ до інформації для подальшого аналізу та порівняння.

Оцінка ризиків і ранжування загроз є важливою складовою інструментарію, що дозволяє визначити рівень небезпеки у певних локаціях на основі проаналізованих даних. Для цього використовується два методи:

1. Якісно-кількісне порівняння – метод, що базується на порівнянні якісних та кількісних показників для оцінки рівня загрози. Цей підхід дозволяє визначати рівень небезпеки на основі специфічних характеристик об'єктів та їхньої кількості, що дає змогу обґрунтовано приймати рішення про заходи безпеки. Кожному типу об'єктів присвоюються вагові коефіцієнти, які визначаються експертним шляхом.

Кількісний та якісний показники розраховується відповідно:

$$\sum_{i=1}^{n} N_i,$$

де n – загальна кількість різних типів технічних об'єктів у локації, N_i – кількість $i-$ их об'єктів

$$\sum_{i=1}^{n} (\omega_i \cdot N_i),$$

де n – загальна кількість різних типів технічних об'єктів у локації, ω_i – ваговий коефіцієнт для i-го об'єкта, N_i – кількість i-х об'єктів

Цей підхід дозволяє об'єктивно оцінити небезпеку, виходячи з кількісних показників, але він не враховує історичний контекст та динаміку подій, що є його основним недоліком.

2. Просторово-динамічне оцінювання – альтернативний метод якісно-кількісному порівнянню, який враховує не лише поточний стан, а й історичні дані про події у певній зоні. Це забезпечує більш точне виявлення потенційних загроз завдяки аналізу накопичення подій та взаємозв’язків між різними повідомленнями. Основним етапом цього підходу є ідентифікація повідомлень (подій), які мають значення для аналізу загроз. Ці повідомлення визначаються як релевантні, якщо вони задовольняють наступні умови:

$$S = \{P_i | D_i \leq d, |t_{cur} - t_i| \leq T\},$$

де S – множина релевантних повідомлень, P_i – попередні повідомлення, D_i – відстань між поточною точкою та точкою повідомлення P_i, d – задана відстань, t_{cur} – час поточного повідомлення, t_i –час повідомлення P_i, T – заданий часовий інтервал.

Таким чином, релевантними вважаються повідомлення (події), які відповідають просторовій близькості та знаходяться у визначеному часовому інтервалі. Якщо в зоні раніше були зафіксовані події, що демонстрували підвищений рівень небезпеки, це збільшує вагу нового повідомлення за допомогою коефіцієнта підвищення ваги.

$$K \cdot \sum_{i=1}^{n} (\omega_i \cdot N_i),$$

де n – загальна кількість різних типів технічних об'єктів у локації, ω_i – ваговий коефіцієнт для i-го об'єкта, N_i – кількість i-х об'єктів, K – коефіцієнт підвищення ваги.

Коефіцієнт підвищення ваги відображає ступінь зростання небезпеки і визначається експертним шляхом. Він може варіюватися залежно від ступеня загрози, що дозволяє гнучко адаптувати оцінку до конкретної ситуації.

Останнім компонентом системного підходу є створення експериментальної експертної системи, яка інтегрує в собі всі вищеописані елементи, створюючи комплексний підхід до аналізу та підтримки прийняття рішень. Вона поєднує автоматизований аналіз текстових даних, моделі обробки природної мови та адаптивні можливості на основі експертних знань. Система функціонує як централізована платформа, яка обробляє вхідні дані, аналізує їх у

контексті просторово-часових взаємозв'язків і надає користувачеві чіткі рекомендації. Завдяки здатності враховувати історичні дані, моделювати зміни у динаміці загроз і коригувати вагові коефіцієнти відповідно до експертних оцінок, система забезпечує точність і надійність аналізу. Її гнучкість і адаптивність роблять її ключовим елементом, здатним ефективно підтримувати процеси прийняття рішень у складних динамічних умовах.

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Дані, використані в дослідженні, представлені у вигляді текстових повідомлень, що були написані користувачами через месенджери. Цей тип даних характеризується високим ступенем неструктурованості, що значно ускладнює їх автоматизований аналіз. Повідомлення часто подаються у вільній формі, містять неоднозначності та різноманітні деталі, які вимагають попередньої обробки для виділення ключової інформації.

На першому етапі дослідження було проведено навчання моделі Bi-LSTM, результати продемонстровано на рисунку 3.

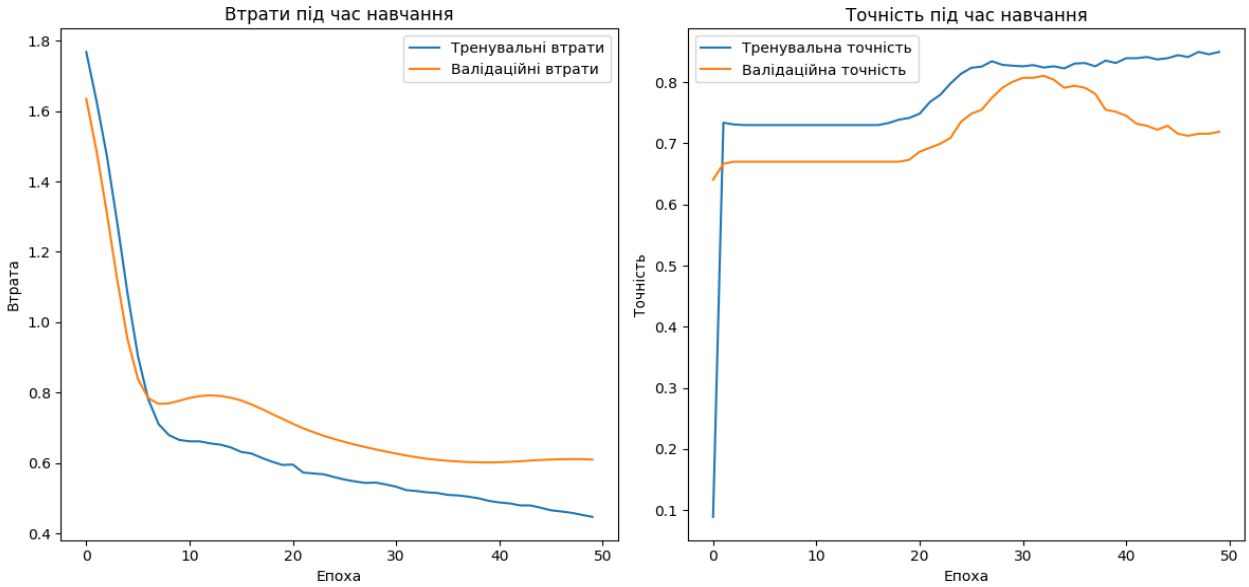


Рисунок 3. Результат навчання моделі Bi-LSTM

Для цього використовувався спеціально підготовлений набір даних, де повідомлення були попередньо протеговані за категоріями: локація, дата, об’єкт, дія об’єкта, стан об’єкта та тип об’єкта. Такий підхід дозволив моделі навчитися розпізнавати слова та вирази відповідно до заданих тегів, що забезпечило правильність розпізнавання повідомлень на рівні 80%. Це підтвердило ефективність Bi-LSTM у задачах автоматизованої екстракції даних. Ручна перевірка результатів навчання продемонструвала, що модель здатна успішно розпізнавати текст із задовільною точністю. Проте, при роботі з неоднозначними даними або повідомленнями зі складними структурами, точність моделі знижувалася.

Середній час обробки одного повідомлення, що містить один запис, склав 2 секунди. Це робить Bi-LSTM придатною для аналізу великих обсягів даних у реальному часі.

Попри високу ефективність у базових задачах, модель виявила певні обмеження при обробці даних із високим ступенем неоднозначності. Це вказує на необхідність пошуку альтернативних підходів.

Наступним етапом дослідження стало впровадження моделі Gemma для аналізу текстових повідомлень. Модель Gemma, завдяки своїй передовій архітектурі, забезпечує високу точність ідентифікації ключових даних. Однак для досягнення оптимального

результату необхідно було визначити найбільш ефективний спосіб формування запиту (промпту). З цією метою було проведено серію експериментів (рис. 4), під час яких запити формувалися з урахуванням раніше визначених категорій: локація, дата, об’єкт, дія об’єкта, стан об'єкта та тип об’єкта.

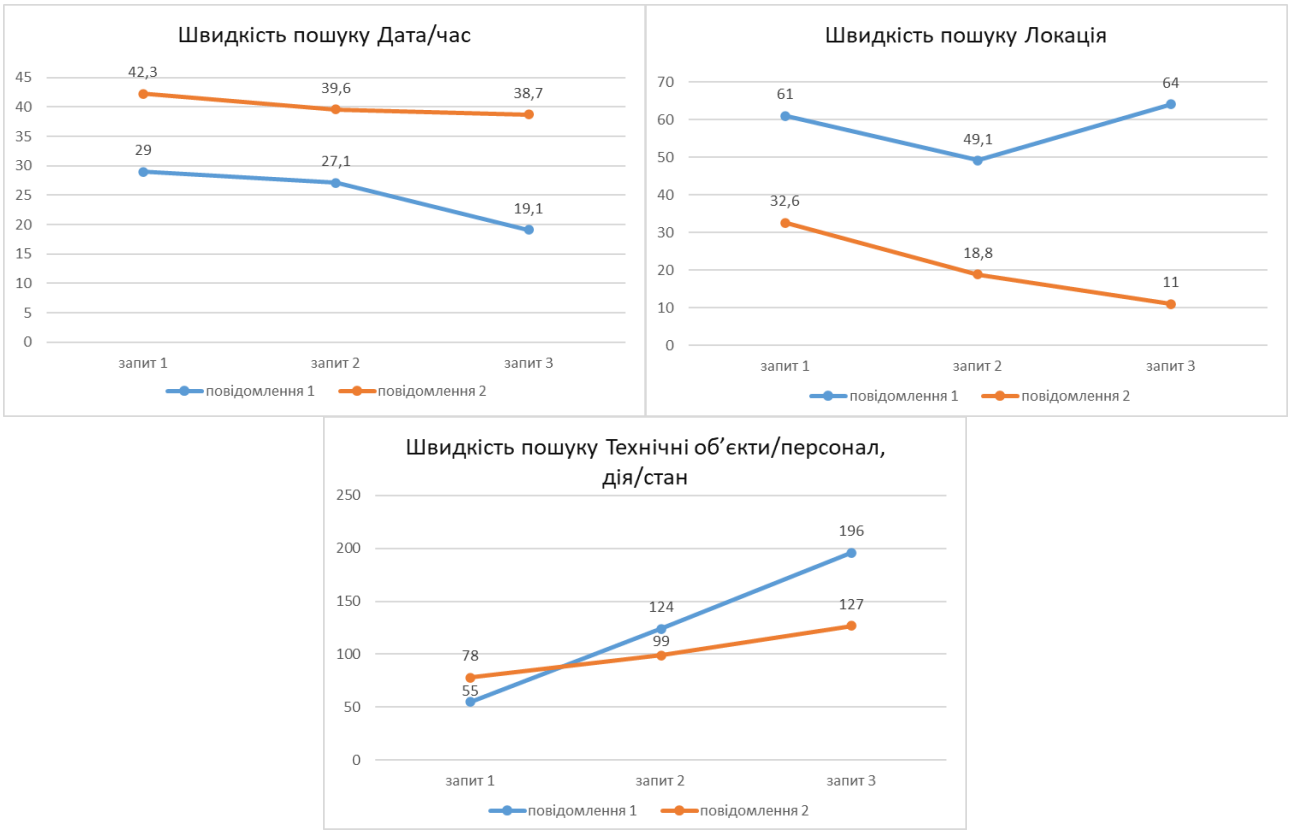


Рисунок 4. Результат експериментів запитів з пошуку фактів в повідомлені

Ці експерименти допомогли визначити структуру промпту, яка одночасно забезпечує правильність отриманих результатів і оптимальну швидкість обробки.

У результаті досліджень було сформовано оптимізовану структуру запиту, яка дозволяє повноцінно обробляти текстові повідомлення з мінімальними часовими витратами (рис. 5).

```
{"role": "system", "content":
'''Витягни з тексту наступні дані і напиши їх у такому форматі:
Дата/час: Виведи дату та час, якщо знайдено, інакше -.
Точки: Виведи точки, якщо знайдено, інакше -.
Якщо є декілька точок, виводь кожну окремо і повторюй всю інформацію
для кожної точки, поділивши кількість технічних об'єктів та персоналу.
Область: Виведи відповідні значення або -.
Місто: Виведи відповідні значення або -.
Село: Виведи відповідні значення або -.
Населений пункт: Виведи відповідні значення або -.
Технічні об'єкти: Виведи тип технічних об'єктів, інакше -.
Якщо є кілька типів технічних об'єктів, виводь кожен тип окремо і повторюй
всю інформацію для кожного типу
Кількість технічних об'єктів: Виведи кількість або -.
Персонал: Виведи кількість або -.
Дія: Визначи як активна чи пасивна, інакше пасивна.'''}
```

Рисунок 5. Робочий промт для моделі Gemma

Такий підхід забезпечив високий рівень точності ідентифікації ключових слів. Ручна перевірка результатів показала, що рівень точності розпізнавання досягнув 97%, що значно перевищує показники моделі Bi-LSTM (рис. 6). Це свідчить про перевагу Gemma у задачах автоматизованої екстракції даних.

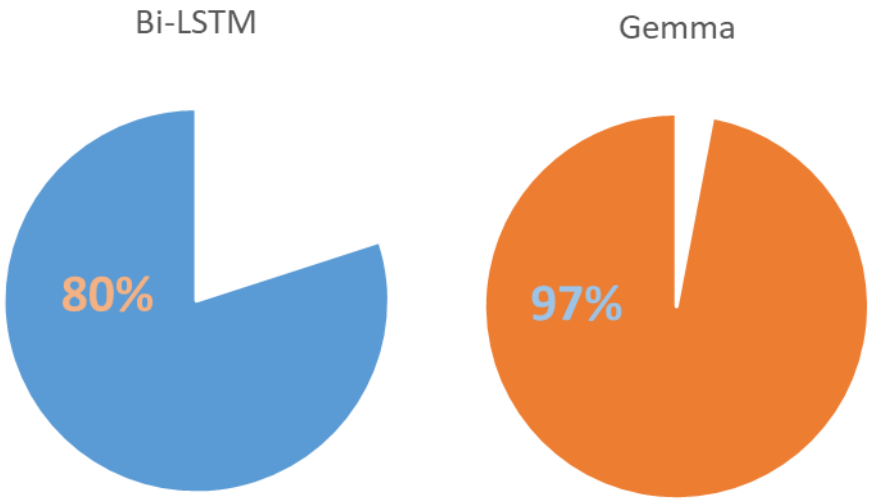


Рисунок 6. Результати точності розпізнавання текстів моделями Bi-LSTM та Gemma

Проте, незважаючи на високу точність, час обробки одного повідомлення виявився більшим, ніж у Bi-LSTM, що слід враховувати при оцінці ефективності моделі в реальному застосуванні. Зокрема, тривалість обробки повідомлень із використанням Gemma перевищувала очікувані значення (рис. 7), що може бути критичним у задачах із часовими обмеженнями.

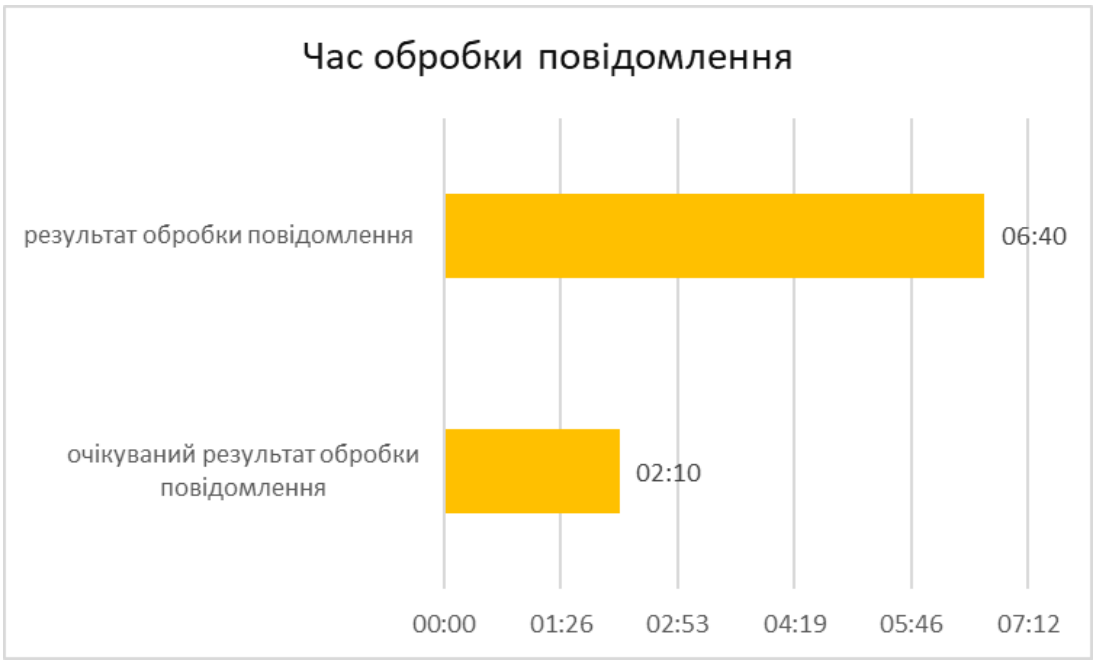


Рисунок 7. Результат та очікуваний результат швидкості обробки повідомлення моделлю Gemma

Однак, порівнюючи результати екстракції фактів між двома моделями, слід зазначити, що саме цей аспект має вирішальне значення в межах цього дослідження. Точність та якість екстракції даних є ключовими показниками ефективності, і в цьому контексті Gemma демонструє перевагу. Отже, модель Gemma була визнана більш придатною для задач, де головним критерієм є точність аналізу текстових даних.

Наступним етапом роботи стало ранжування загроз на основі зібраних та оброблених даних. Цей етап має вирішальне значення для практичного застосування отриманих результатів, адже він дозволяє не лише аналізувати наявну інформацію, але й визначати пріоритети у прийнятті рішень.

Для цього було використано дві основні методики оцінювання загроз: якісно-кількісне порівняння та просторово-динамічне оцінювання. Обидва підходи дозволяють оцінити рівень небезпеки, але використовують різні методологічні принципи, що робить їх ефективними у різних контекстах.

Якісно-кількісне порівняння базується на обчисленні якісного показника, що враховує вагові коефіцієнти для різних типів об'єктів та їх кількість. Загальний ризик визначається як добуток вагового коефіцієнта та кількості об'єктів певного типу. Далі результати порівнюються з кількісним показником, який відображає загальну кількість об'єктів (рис. 8). У випадках, коли якісний показник перевищує кількісний, зона вважається підвищеної небезпеки.

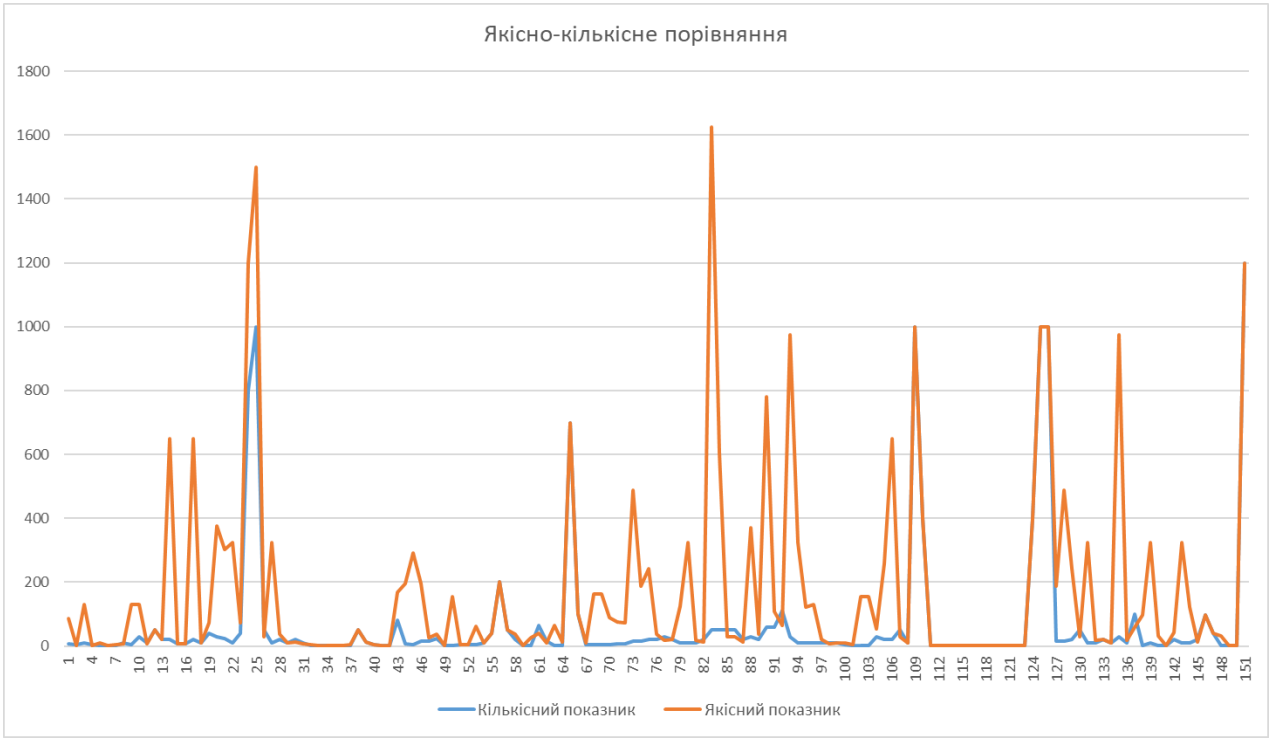


Рисунок 8. Якісно-кількісне порівняння

Цей підхід є швидким та ефективним для аналізу великих обсягів даних, але має обмеження у врахуванні просторово-часових характеристик подій.

Просторово-динамічне оцінювання дозволяє враховувати не лише поточні дані, але й історичну динаміку подій у визначеній зоні. Аналіз нових повідомлень проводився з урахуванням їхнього впливу на вже існуючі загрози у радіусі, визначеному експертним шляхом, а також у межах заданого часового інтервалу (рис. 9).

Рисунок 9. Просторово-динамічне оцінювання

Як можна побачити, в просторово-динамічному оцінюванні попередні якісні оцінки набули інших, більших ваг. Це свідчить про те, що певні зони, можливо, були недооцінені як зони підвищеної небезпеки, оскільки повідомлення, отримані раніше, не несли відповідної інформації або не мали достатньо високої базової оцінки.

Це підкреслює важливість врахування не лише окремих повідомлень, а й їхнього контексту в часі та просторі. Тому просторово-динамічне оцінювання виявляється більш адаптивним і здатним виявляти приховані загрози, які могли залишитися поза увагою у рамках якісно-кількісного порівняння. Застосування такого підходу дозволяє запобігти потенційним помилкам у розподілі пріоритетів загроз і забезпечити більш повне розуміння загальної ситуації в зоні аналізу. Ранжування загроз стало логічним продовженням роботи з обробленими даними. Воно дозволило створити ефективний інструмент для оцінки ризиків, який може бути використаний у реальних умовах для прийняття обґрунтованих рішень.

Отже, проведені дослідження та отримані результати не лише підтвердили ефективність застосованих методів, але й створили базу для розробки експериментальної експертної системи. Ця система інтегрує методи екстракції даних, ранжування загроз і просторово-динамічного оцінювання, забезпечуючи комплексний підхід до аналізу та оцінки ризиків. Вона має потенціал значно підвищити точність і оперативність прийняття рішень у складних та динамічних умовах.

4. ВИСНОВКИ

Проведене дослідження було спрямоване на вдосконалення методів аналізу текстових даних для підтримки прийняття рішень у критичних ситуаціях. Основна увага була приділена розробці підходів до автоматизованої екстракції ключових даних із текстових повідомлень та оцінки загроз. У результаті роботи вдалося визначити найбільш ефективну модель для аналізу тексту – Gemma, яка, порівняно з Bi-LSTM, продемонструвала значно вищу точність розпізнавання даних. Експериментальне визначення структури запитів дозволило оптимізувати процес взаємодії з моделлю та покращити результати екстракції фактів.

Також у дослідженні було запропоновано два підходи до ранжування загроз – якісно-кількісне порівняння та просторово-динамічне оцінювання. Останній метод показав свою перевагу за рахунок врахування історичних подій та їхнього впливу на поточну ситуацію, що дозволило підвищити точність визначення зон ризику.

Зібрані результати та запропоновані методики створили основу для подальшого розвитку експериментальної експертної системи, яка може стати потужним інструментом у задачах оцінки ризиків, ранжування загроз та підтримки прийняття рішень. Отримані висновки відкривають перспективи для інтеграції таких систем у реальні процеси управління ризиками та підвищення безпеки у критичних галузях.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. А.І. Вавіленкова Теоретичні основи аналізу електронних текстів, 2016. с. 5-9. URL: https://er.nau.edu.ua/bitstream/NAU/42695/10/All.pdf

2. Beakcheol Jang , Myeonghwi Kim, Gaspard Harerimana, Sang-ug Kang and Jong Wook Kim Bi-LSTM Model to Increase Accuracy in Text Classification: Combining Word2vec CNN and Attention Mechanism. Appl. Sci. 2020. 1-5 p.

3. The Complete Guide to Information Extraction from Texts with Spark NLP and Python. URL: https://medium.com/john-snow-labs/the-complete-guide-to-information-extraction-from-texts-with-spark-nlp-and-python-c862dd33995f

4. Pouyanfar S., Sadiq S., Yan Y., Tian H., Tao Y., Reyes M.P., Shyu M.-L., Chen S.-C., Iyengar S.S. A survey on deep learning: Algorithms, techniques, and applications. ACM Comput. Surv. (CSUR) 2018. 1–36.

5. Complete Guide to RNN, LSTM, and Bidirectional LSTM. URL: https://dagshub.com/blog/rnn-lstm-bidirectional-lstm/

6. Exploring Gemma: Google open-source AI model. URL: https://medium.com/pythoneers/exploring-gemma-google-open-source-ai-model-812e71b539c0#036a

ІНСТРУМЕНТАРІЙ АНАЛІЗУ КОНТЕКСТУАЛЬНИХ ВЗАЄМОЗВ'ЯЗКІВ ТЕХНОЛОГІЙ У ТЕКСТОВИХ ДАНИХ З ВИКОРИСТАННЯМ LLM І ГРАФОВИХ СТРУКТУР

Петренко М.М.[1], Савастьянов В.В.

Національний технічний університет України
"Київський політехнічний інститут ім. Ігоря Сікорського", Київ, Україна

[1] 1256f13@gmail.com

Аналіз контекстуальних взаємозв'язків технологій у текстових даних є важливою задачею в сучасних дослідженнях, що стосуються обробки природної мови та штучного інтелекту. У цьому контексті важливо оцінити структуру і значення зв'язків між різними технологіями, а також їх роль у текстових даних. Технології аналізу контекстуальних взаємозв'язків, зокрема з використанням великих мовних моделей (LLM) та графових структур, дозволяють здійснювати глибокий аналіз та побудову моделей, що відображають складні взаємозв'язки. Метою цієї роботи є розробка інструментів для аналізу контекстуальних взаємозв'язків між технологіями в текстових даних за допомогою LLM і графових структур, а також проведення порівняльного аналізу різних підходів до аналізу таких взаємозв'язків.

Ключові слова: контекстуальні взаємозв'язки, LLM, графові структури, аналіз текстових даних, технології, семантичний аналіз, класифікація текстів.

1. ВСТУП

У сучасному світі аналіз та оптимізація контекстуальних взаємозв'язків між технологіями в текстових даних набувають все більшої актуальності. Розвиток штучного інтелекту, зокрема великих мовних моделей (LLM), значно розширює можливості обробки та аналізу великих обсягів текстової інформації, допомагаючи виділяти важливі патерни та взаємозв'язки між різними технологіями. Одним із основних завдань є виявлення та аналіз контекстуальних зв'язків, що дозволяє отримувати більш точні інсайти та покращувати процеси прийняття рішень у різних сферах, від науки до бізнесу.

Технології аналізу текстових даних, засновані на використанні LLM і графових структур, надають потужні інструменти для побудови моделей, здатних ідентифікувати складні взаємозв'язки та контексти. Водночас, інтеграція різних підходів, таких як графи знань, семантичний аналіз і методи машинного навчання, дозволяє створювати адаптивні та ефективні системи для обробки неструктурованої інформації. Моделювання контекстуальних зв'язків у текстових даних є важливим аспектом для досягнення більш глибокого розуміння технологій та їх взаємодії в рамках різних доменів.

2. СИСТЕМНИЙ ПІДХІД ЩОДО ПОБУДОВИ ІНСТРУМЕНТАРІЮ

Основна структура підходу щодо побудови інструментарію зображена на рис 1.

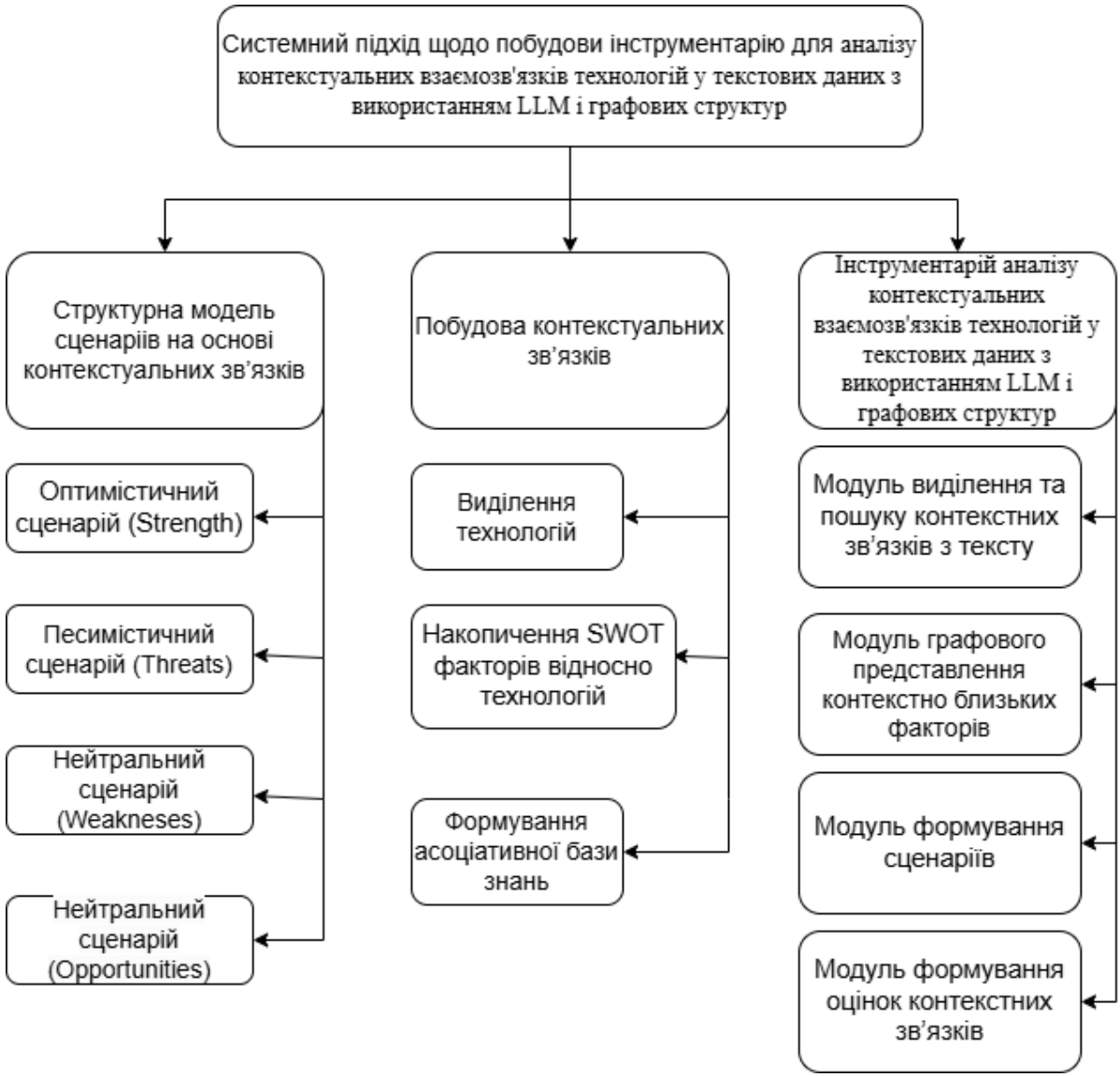


Рисунок 1. Структура підходу побудови інструментарію

Приклад структурної моделі сценаріїв на основі контекстуальних зв'язків виглядає наступним чином.

Множини, які використовуються для моделювання.

1. T – множина технологій.

T = {дрони, штучний інтелект, системи РЕБ, системи наведення, сенсори, тощо}.

2. F_Threats – множина загроз.

F_Threats = {високий ризик втрат внаслідок одночасного ураження, підвищена точність ураження цілей, розробка протидронових засобів, складність розробки, вимоги до інформаційної безпеки та захисту від зламу, тощо}.

3. Y – множина часових етапів:

Y = {y1, y2, y3, y4}, де y_i – рік виконання.

4. D – множина можливих дій:

D = {d1, d2, ..., d_k}, де d_k — конкретна дія.

Для опису взаємодії множин використовуються наступні функції:

1. Функція ризику R:

R: T × F_Threats → R+, де R(t_i, f_j) – рівень впливу загрози f_j на технологію t_i.

2. Функція впливу часу T_Impact.

T_Impact: R × Y → R+, де T_Impact(R(t_i, f_j), y) – посилення впливу загрози f_j на t_i з часом y.

3. Функція ефективності дій E.

E: D × (T × F_Threats) → R, де E(d_k, (t_i, f_j)) – ефективність дії d_k для зменшення впливу f_j на t_i.

4. Функція вибору дії C.

C: (T × F_Threats × Y) → D, де C(t_i, f_j, y) визначає оптимальну дію d_k для технології t_i, враховуючи загрозу f_j і часовий етап y.

Для вибору оптимального рішення використовується функція цільової оптимізації:

min_{d_k ∈ D} Σ_{i=1}^n Σ_{j=1}^m T_Impact(R(t_i, f_j), y) · (1 - E(d_k, (t_i, f_j))).

Мета: мінімізувати негативний вплив загроз на технології за всі часові етапи.

1. Вхідні дані: технології T, загрози F_Threats, набір дій D, модель ризику R(t_i, f_j).
2. Розрахунок впливу загроз: визначити T_Impact(R(t_i, f_j), y) для кожного t_i, f_j, y.
3. Оцінка ефективності дій: обчислити E(d_k, (t_i, f_j)) для кожного можливого d_k.
4. Оптимізація: знайти d_k, що мінімізує цільову функцію.
5. Призначення дій: обрати C(t_i, f_j, y) = d_k для кожної комбінації t_i, f_j, y.

Негативний сценарій формалізовано наступним чином.

Негативний сценарій = ⋃_{y_i ∈ Y} D(y_i, G(F_Threats, T, y_i)).

За аналогічним принципом створено моделі інших сценаріїв на основі контекстуальних зв’язків

В ході роботи побудовано експертну систему підтримки прийняття рішень. Нижче розглянута формалізація експертної системи на прикладі моделі для побудови песимістичних сценаріїв.

Основні множини для експертної системи:

1. T – множина технологій: T = {t1, t2, ..., tn}, де t_i – окрема технологія.

2. F_Threats – множина факторів загроз: F_Threats = {f1, f2, ..., fm}, де f_j – окрема загроза.

3. Y – множина часових етапів: Y = {y1, y2, y3, y4}, де y_i – часовий етап (рік).

4. D – множина можливих дій: D = {d1, d2, ..., dk}, де d_k – конкретна дія, спрямована на зменшення впливу загроз.

Функції

Для моделювання взаємодії множин використовуються наступні функції:

1. Функція ризику R:

R: T × F_Threats → R+, де R(t_i, f_j) — рівень впливу загрози f_j на технологію t_i.

2. Функція впливу часу T_Impact:

T_Impact: R × Y → R+, де T_Impact(R(t_i, f_j), y) — посилення впливу загрози f_j на t_i з часом y.

3. Функція ефективності дій E:

E: D × (T × F_Threats) → R, де E(d_k, (t_i, f_j)) — ефективність дії d_k для зменшення впливу f_j на t_i.

4. Функція вибору дії C:

C: (T × F_Threats × Y) → D, де C(t_i, f_j, y) визначає оптимальну дію d_k для технології t_i, враховуючи загрозу f_j і часовий етап y.

Критерій оптимальності

Цільова функція для оптимізації дій:

$$\min_{d_k \in D} \sum_{i=1}^{n} \sum_{j=1}^{m} T\_Impact(R(t_i, f_j), y) \cdot (1 - E(d_k, (t_i, f_j))).$$

Мета: мінімізувати негативний вплив загроз на технології за всі часові етапи.

Алгоритм прийняття рішень

1. Вхідні дані: технології T, загрози F_Threats, набір дій D, модель ризику R(t_i, f_j).
2. Розрахунок впливу загроз: визначити T_Impact(R(t_i, f_j), y) для кожного t_i, f_j, y.
3. Оцінка ефективності дій: обчислити E(d_k, (t_i, f_j)) для кожного можливого d_k.
4. Оптимізація: знайти d_k, що мінімізує цільову функцію.
5. Призначення дій: обрати C(t_i, f_j, y) = d_k для кожної комбінації t_i, f_j, y.

Експертна система прийняття рішень базується на формалізованій моделі взаємодії загроз, технологій, часових етапів і дій. Запропонована структура дозволяє мінімізувати вплив загроз через оптимізацію стратегій реагування для кожного етапу часу. Ця модель є універсальною і може бути адаптована до різних сфер застосування залежно від специфіки вхідних даних та пріоритетів системи.

3. МОДЕЛЮВАННЯ ТА РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для дослідження були зібрані текстові дані з Telegram-чату “Victory Drones”. Важливими етапами стали очищення, нормалізація, анонімізація текстів і форматування їх у формат .csv. Це забезпечило створення якісного корпусу текстів для роботи з великими мовними моделями. Приклад вилучених повідомлень зображено на рис. 2.

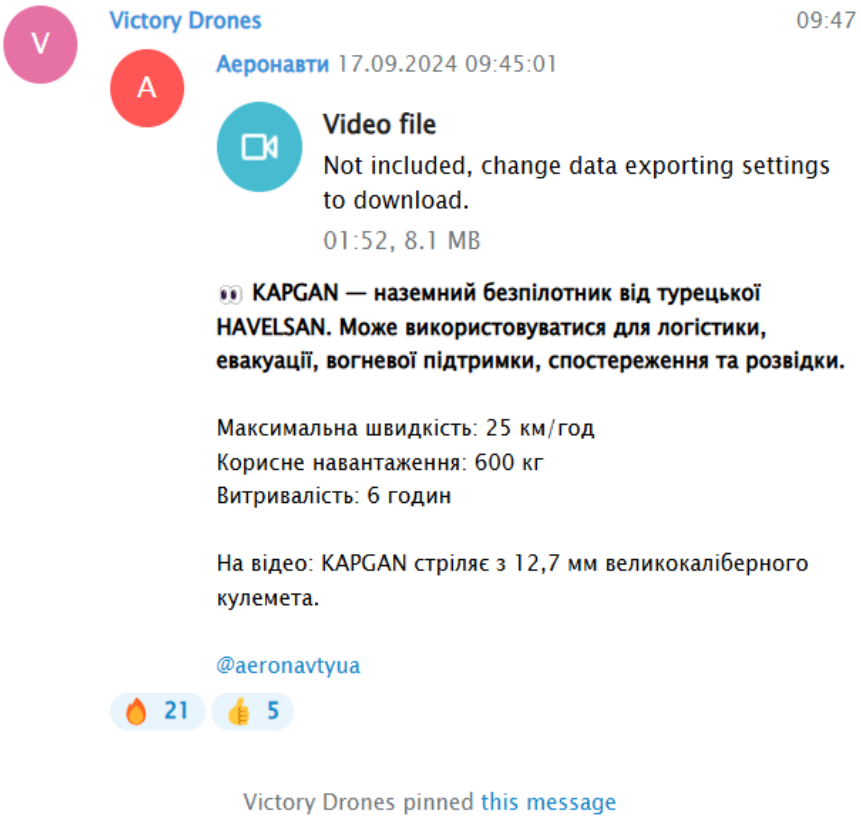


Рисунок 2. Приклад вилучених повідомлень

Для вилучення знань використано модель Gemma 2 9B з 9. Було створено промпти для категоризації дронів, виділення їх технічних характеристик і аналізу сфер застосування. Тестування показало високу точність класифікації даних.

Автоматизований парсинг тексту дозволив виділити типи дронів (квадрокоптери, гексакоптери) та їх ключові характеристики. Результати були класифіковані і додані в базу знань для подальшого аналізу.

SWOT-аналіз дронів охопив їхні сильні та слабкі сторони, можливості і загрози. Використання Gemma 2 9B забезпечило структурований підхід до оцінки кожного типу дронів і створення аналітичної бази. Приклад вмісту бази знань після проведеного аналізу зображено на рис. 3.

	Strengths	Weaknesses	Opportunities	Threats
0	Збільшена дальність та тривалість польоту: За ...	Вразливість до перехоплення та перешкод: Дрони...	Подальший розвиток технології: Прогрес в облас...	Протягнення збройової гонки: Прориви в техноло...
1	Збільшена дальність та тривалість польоту: За ...	Вразливість до перехоплення та перешкод: Дрони...	Подальший розвиток технології: Прогрес в облас...	Збільшення ризику злочинного використання:
2	Збільшена дальність та тривалість польоту: За ...	Вразливість до перехоплення та перешкод: Дрони...	Подальший розвиток технології: Прогрес в облас...	дрони можуть бути використані для незаконни...
3	Збільшена дальність та тривалість польоту: За ...	Вразливість до перехоплення та перешкод: Дрони...	Подальший розвиток технології: Прогрес в облас...	Непередбачувані наслідки: Швидкий розвиток те...
4	Збільшена дальність та тривалість польоту: За ...	Вразливість до перехоплення та перешкод: Дрони...	Зміщення фокусу від наземних операцій: Збільш...	Протягнення збройової гонки: Прориви в техноло...

Рисунок 3. Приклад вмісту бази знань

Розроблена система пошуку дозволила знаходити контекстуально близькі факти, що полегшує аналіз бази знань. Графове представлення візуалізувало зв'язки між типами дронів, їх характеристиками та сферами застосування, забезпечуючи інтуїтивне уявлення. Приклади пошуку контекстно близьких фактів з бази знань та графічного представлення взаємозв'язків зображено на рис. 4 та рис. 5.

	Факт	Категорія	Схожість
0	Opportunities: Подальший розвиток технології: ...	Strengths	0.957830
1	Strengths: Збільшена дальність та тривалість п...	Strengths	0.949989
2	Opportunities: Зміщення фокусу від наземних о...	Strengths	0.949794
3	Opportunities: Протягнення збройової гонки: П...	Strengths	0.949273
4	Threats: Протягнення збройової гонки: Прориви ...	Strengths	0.946476

Рисунок 4. Приклад пошуку контекстно близьких фактів

4. ВИСНОВКИ

У цій роботі було продемонстровано ефективність використання великих мовних моделей (LLM) та графових структур для аналізу контекстуальних взаємозв'язків у текстових даних. Запропонований підхід дозволяє виявляти складні взаємозв'язки між технологіями, оцінювати вплив загроз і пропонувати оптимальні стратегії для мінімізації ризиків.

У ході дослідження було розроблено інструменти, які надають можливість аналізувати ризики, вплив часу на загрози, ефективність дій та обирати оптимальні рішення для зменшення негативного впливу загроз. Це дало змогу провести структурований SWOT-аналіз, враховуючи сильні та слабкі сторони, можливості використання та загрози. Система пошуку контекстуально близьких фактів і графове представлення зв'язків полегшують доступ до інформації, забезпечуючи глибше розуміння зв'язків між об'єктами дослідження.

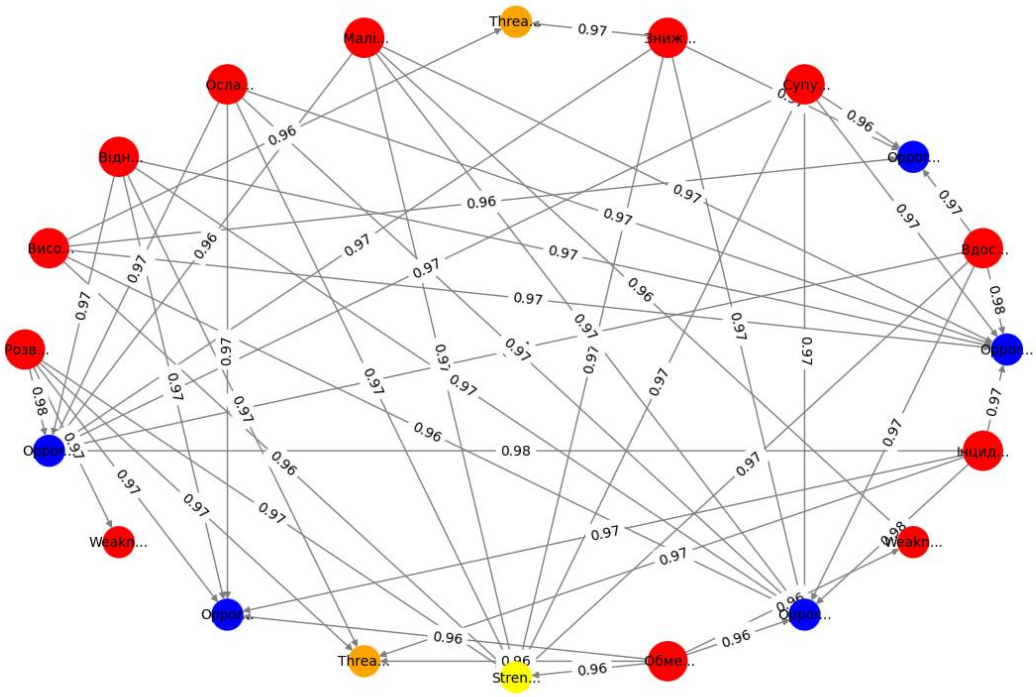


Рисунок 5. Приклад графічного представлення взаємозв’язків

Отримані результати демонструють, що запропонований підхід є універсальним і може бути адаптований для аналізу даних у різних сферах. Використання моделі Gemma 2 9B показало високу точність класифікації даних і забезпечило надійну базу для аналітичної роботи. Розроблена система є цінним інструментом для покращення процесів прийняття рішень, дозволяючи мінімізувати вплив загроз на технології через оптимізацію стратегій реагування.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Brown T., Mann B., Ryder N., et al. Language Models are Few-Shot Learners. *Proceedings of the 2020 Conference on Neural Information Processing Systems (NeurIPS 2020)*, 2020.

2. Mikolov T., Chen K., Corrado G., Dean J. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*, 2013.

3. Hamilton W.L., Ying R., Leskovec J. Representation Learning on Graphs: Methods and Applications. *arXiv preprint arXiv:1709.05584*, 2017.

4. Vaswani A., Shazeer N., Parmar N., et al. Attention Is All You Need. *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

5. Blei D.M., Ng A.Y., Jordan M.I. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 2003.

6. Rong X. Word2vec Parameter Learning Explained. *arXiv preprint arXiv:1411.2738*, 2014.

7. Chen Z., Wang P., Hu Y. Knowledge Graphs for Natural Language Processing. *Computational Linguistics Journal*, 2019.

ANALYSIS OF NON-LINEAR AND NON-STATIONARY PROCESSES IN ECONOMY AND FINANCE

Bidyuk P.I.[1], Tymoshchuk O.L.[2], Levenchuk L.B.[3]

National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine

[1] pbidyuke_00@ukr.net [0000-0002-7421-3565], [2] oxana.tim@gmail.com [0000-0003-1863-3095], [3] levenchuk.liudmyla@lll.kpi.ua [0000-0002-8600-0890]

Most of the process we analyze in economy, finance and many other areas of activity belong to the class of non-linear and non-stationary (NNP). Their structures are more sophisticated than linear ones what requires substantial efforts for their estimation and adaptation. That is why they need more elaborate procedures for modeling and forecasting. Analysis of both types of processes has similar features what allows to discuss them in the frames of one context. The study is directed towards development of convenient and to some extent generalized methodology of modeling the processes using some known procedures like identification of non-linearity and non-stationarity, structure and parameter estimation etc. We propose to start with identification and taking into consideration possible data uncertainties to prepare data for modeling. Then model structure is estimated using all available knowledge about process under study. On this stage statistical analysis of available data, knowledge of known model structures and expert estimates are used. Experience of researcher plays substantial role in estimating (selecting) appropriate model structure for a specific case. Then parameters are estimated for candidate models using alternative parameter estimation techniques like nonlinear LS, maximum likelihood, Markov chain Monte Carlo or other suitable method. The last stage of modeling supposes selection of best model among constructed candidates by making use of appropriate sets of statistics. The whole model constructing procedure is better to perform in the frames of specialized intellectual decision support system (IDSS) containing necessary computing procedures and means necessary for simulation. The high quality of final result is achieved thanks to appropriate tracking of computing procedures at all stages of data processing: preliminary data analysis, model constructing, and forecasts estimation. It is supposed that such system will find its applications as an extra tool for decision making when developing the strategies for economic and financial processes evolution and analysis.

Keywords: non-linear and non-stationary processes, economy and finances, mathematical modeling and forecasting, system analysis approach, intellectual decision support system.

1. INTRODUCTION

The problems of modeling, forecasting and management of non-linear and non-stationary processes in economy and finance are available practically in most countries. Especially urgent are

the problems of mathematical modeling, estimation, and forecasting possible directions of future evolution for developing economies and economies in transition. These problems arise due to outdated industrial and agricultural technologies and management, strong international competition, influence of various disturbances (mostly stochastic), and switching of functioning regimes in economy and finance. To describe formally non-linear and non-stationary processes there is a set of ideologically different approaches based upon probabilistic-and-statistical methods and intellectual data analysis. The available known fundamental models are applied not so often as empirical models based upon statistical data and expert estimates. The principal objective of this study is to reveal key issues that researchers confront in attempt to construct statistical, probabilistic or other models of the process mentioned. Wide applications have found such models as logistic and linear/nonlinear regression, support vector machine, neural nets, neuro-fuzzy models, Bayesian data analysis, theory of distributions, generalized linear models (GLM), radial basis functions and combinations of the methods mentioned [1–3].

Selection of a method for the formal description of non-linear and non-stationary processes (NNP) is mainly determined by availability of statistical data and instrumental platform for performing computations, staff qualification, and appropriate material support. These processes can be characterized with the following special features: sophisticated structure, availability of uncertainties (statistical, structural, and parametric), necessity for availability of several methods for parameter estimation; often unknown parameters distribution; necessity for applying simulation procedures. The features require as a rule constructing specialized intellectual decision support systems for obtaining high quality results of NNP analysis. Remind that IDSS design is based upon latest developments of system analysis ideas, information system theory, estimation theory, mathematical and statistical-and-probabilistic modeling and forecasting, intellectual data analysis, decision theory and other results of theory and practical data as well as expert estimates processing methods [4, 5].

The study is aiming to development of methodology for modeling non-linear and non-stationary processes that should be helpful from the point of view of refining the model constructing procedure, enhancing model adequacy and increasing quality of forecasts. The next step of the study is implementation of the methodology in the frames of specialized decision support system using system analysis principles.

2. PROBLEM STATEMENT

For a class of economic and financial processes, $\{y(\cdot)\}$, having generalized type of probability distribution

$$f(y,\theta,\varphi) = exp\left[\frac{y\theta - b(\theta)}{a(\varphi)} + c(y,\varphi)\right],$$

where *a*, *b*, *c* are functions that correspond to specific distribution function; y, is dependent variable; θ is canonic parameter or function of some parameter of specific distribution; ϕ, is parameter of variance. It is necessary to construct adequate model applicable for short- and middle-term forecasting and control. Various distribution laws are possible, more specifically the following: normal, Poisson, binomial, inverse Gaussian, gamma, and exponential. The processes may contain non-linearity in variables of the following type: polynomial, bilinear, and some non-linearity in parameters and non-stationarity, for example:

$$y(k) = a_0 + \sum_{i=1}^{p} a_i y(k-i)\, y(k-j) + \sum_{j=1}^{q} b_j v(k-i) + cx^m(k) + \varepsilon(k), j = 1,2,\ldots,$$

or generalized bilinear models:

$$y(k) = a_0 + \sum_{i=1}^{p} a_i y(k-i) + \sum_{j=1}^{q} b_j v(k-j) + \sum_{i=1}^{m}\sum_{j=1}^{s} c_{i,j} y(k-i)v(k-j) + \varepsilon(k),$$

where *m* is integer that determines degree of non-linearity regarding explaining variable; *p*, *q*, *s* characterize order of corresponding model components; $\{\varepsilon(k)\}$ is random process taking into

consideration external disturbances, errors of computations and measurement errors. It should be possible to construct mentioned or other models of this class.

The methodology for constructing nonlinear processes models should provide for the possibility of achieving required model adequacy, and high quality of forecasts based upon them. The methodology should fit applications considered in the frames of specialized decision support system that is formally defined as follows:

$$DSS = \{DKB, PDP, DT, MSE, MPE, DQ, MQ, FEQ, AQ\},$$

where *DKB* is data and knowledge base; *PDP* is a set of procedures for preliminary data processing; *DT* is a set of statistical tests for determining possible effects contained in data (like nonlinearity, integration or heteroscedasticity); *MSE* is a set of procedures for estimation of mathematical model structure; *MPE* is a set of procedures for estimation of mathematical model parameters; *DQ*, *MQ*, *FEQ*, *AQ* are the sets of statistical quality criteria for estimating quality of data, model quality, forecasts estimates quality, and decision alternatives, accordingly [5, 6].

3. SOME TYPES OF NONLINEAR MODELS

Some known nonlinear processes models in economy and finances are given below [6].

– Nonlinear autoregression that contains only functions of dependent variable and stochastic process that approximates influence of non-measurable external disturbances, measurement errors and estimates of model structure and parameters. Typical functional elements of the models are the following: *sin* or *cosine*, sign function, functions of integer degrees, logarithms of modules and relations of low order polynomials.

– Nonlinear transfer functions that include functions of time delayed dependent variable, current and time delayed independent (explaining) variables.

– Bilinear models like this: $y(k) = \sum_{j,k} \beta_{jl} y(k-j)\, \varepsilon(k-l) +$ other components that contain products of $x(k)$ and time delayed residuals, $\varepsilon(k-i)$, where i determines estimated allowable lags. Such equations usually are a part of multiple bilinear system.

– Models with nonlinear moving average that contain time delayed functions of residuals, $\varepsilon(k), e(k)$, and other possible random processes.

– Double stochastic models (stochastic variables and stochastic parameters) that contain products of time delayed dependent variables, $y(k)$, and current or time delayed independent components, $x_i(k)$, or stochastic parametric processes [1].

Most existing nonlinear models can be extended with linear autoregression. However, analysis of such models usually meets substantial difficulties. As a result of the difficulties the values of lags (time delays) usually accept rather low values. The problem of lag estimation for nonlinear models is usually considered separately though the ideas regarding estimation algorithms are close to the ideas used for analysis of linear system.

Some examples of models that are nonlinear in parameters.

– Power function:
$y(k) = a[x(k)]^b$, and transformed: $log[\, y(k)] = log\, a + b\, log[\, x(k)]$.

– Exponential function:
$y(k) = a \cdot b^{x(k)}$, and transformed: $log[\, y(k)] = log\, a + x(k) \cdot log\, b$.

– Exponential and power function: $y(k) = a \cdot [x(k)]^b \cdot c^{x(k)}$, and transformed:
$$log[\, y(k)] = log\, a + b\, log[\, x(k)] + x(k)\, log\, c.$$

– Ecological function: $y(k) = a \cdot exp(-b^2(x-c))^2$, and transformed:
$$log\, y = log\, a - b^2 c^2\, log\, e + 2b^2 c\, log(\, e)x - b^2 (log\, e)x^2.$$

– Logistic function: $y = \frac{a}{1+be^{-cx}}$, and transformed: $log\left(\frac{a}{y} - 1\right) = log\, b - cx\, log\, e$;

$y = \frac{a}{1+e^{b-cx}}$, and transformed: $log\left(\frac{a}{y} - 1\right) = b\, log\, e - cx\, log\, e.$

Such linearization provides a possibility for application of LS technique for parameter estimation. However, the parameters estimated this way may not correspond completely to statistical properties of LS estimates.

4. ESTIMATION OF NON-LINEAR MODEL STRUCTURE

Generally, estimation of nonlinear model structure using available data is performed in several stages shown below.

1. Identification and elimination of possible data uncertainties. Select (estimate) the structure of linear model on the basis of possible correlation analysis. As a rule, this is low order model that is necessary for testing the data under study for linearity (non-linearity); to solve the problem well known methodology of modeling linear processes can be applied [2, 6].

2. Perform testing of the process under study on linearity using the linear model constructed on previous step as a null hypothesis against smooth transition regression (STR); if the hypothesis regarding linearity is not accepted, then determine the transition variable on the basis of available data.

3. Select for further study the logistic smooth transition regression (LSTR) or exponential regression with smooth transition (exponential smooth transition regression (ESTR) for further testing of residuals for linearity.

4. Select (estimate) candidate models and perform analysis of the models adequacy for two variants of possible final model: with additive combination of linear and nonlinear parts and with multiplicative combination. If necessary (for reaching higher adequacy) add to the structures of candidate models some of the following nonlinear components:

$x^2(k), x^3(k), x(k)s(k), x^2(k)s(k)$, where, $x(k), s(k)$ are regressors.

5. Estimate candidate model parameters using correctly selected methods of estimation. Among possible methods are ordinary LS, nonlinear LS, maximum likelihood; Markov chain Monte Carlo [7, 8]. Examining the graph of basic (depending) variable and computed forecast can help to select nonlinear components of a model, say logistic regression to which will be added linear part, for example, linear regression.

6. Select the best model among the candidates constructed using appropriately chosen statistical adequacy criteria.

Testing according step 3 is based on the following model:

$$\hat{v}(k) = \beta_0^T \boldsymbol{w}(k) + \beta_1^T \boldsymbol{w}(k) z_d(k) + \beta_2^T \boldsymbol{w}(k) z_d^2(k) + \beta_3^T \boldsymbol{w}(k) z_d^3(k) + \eta(k), \qquad (1)$$

where $z_d(k)$ is transition variable; $\hat{v}(k)$ are residuals of linear regression estimated with ordinary LS: $y(k) = \beta^T \boldsymbol{w}(k) + v(k)$. If $z_d(k)$ is component of vector $\boldsymbol{w}(k)$, then vector, $\boldsymbol{w}(k) = [1, \widetilde{\boldsymbol{w}}^T k)]^T$ in (1) should be replaced by $\widetilde{\boldsymbol{w}}(k)$, except for the first component in right hand side of the equation. The null hypothesis regarding linearity is formulated in this case as follows: $H_{0d}: \beta_1, \beta_2, \beta_3 = 0$.

The equation (1) is also used for selecting transition variable, $z_d(k)$. The testing is performed for all possible candidates for the transitional variable, $z_d(k)$, and the variable is left for which p-value accepts minimum value. If the condition is not satisfied then the model (process) is considered as a linear one.

Correctness of this testing procedure is motivated with the following considerations. Suppose, that there exists such smooth transition regression (STR-model) with transitional variable, $z_d(k)$, that is characterized by appropriate statistical data (according to the null hypothesis). In this case application of Lagrange multiplier test or LM-test against this hypothesis is characterized by maximum power [1]. The main point in constructing model of nonlinear process is its correct

structure estimation. To fulfill the task any appropriate information should be used: historical studies, expert estimates, preliminary studying of a process, correct identification of non-linearity, analysis of known cases of modeling similar processes, simulation etc.

5. CONCLUSIONS

The methodology is proposed for constructing models for non-linear and non-stationary processes. The main point of the methodology is correct estimation of a model structure. The model structure is estimated using all available knowledge about process under study. It is performed statistical analysis of available data, knowledge of known model structures (for similar processes) and expert estimates are also used. Experience of a researcher plays substantial role in estimating (selecting) appropriate model structure for a specific case. For a class of processes considered it is convenient to have appropriate set of models for selecting candidates in the process of modeling.

It should be stressed that the methodology can be better implemented in the frames of specialized intellectual decision support system designed on system analysis principles. Such approach will be helpful with performing the following tasks: identification and taking into consideration possible probabilistic-and-statistical data uncertainties; application of optimization and adaptation procedures for model structure and parameter estimation; generating of forecasts for selected horizon using candidate models; application of appropriate sets of test statistics for data, models and forecasts quality analysis; monitoring of computational procedures at all stages of data processing; alternatives generating regarding decision making according to problem statement. Such system provides the possibility for reaching high quality of intermediate and final results of computations thanks to applying appropriate quality criteria at each stage of available data analysis and hiring appropriate simulation procedures.

In future studies it is planned enhancing automatizing data processing and model constructing procedures, active usage of intellectual data processing techniques and constructing models of Bayesian type [8, 9], neural nets, combined models, as well as appropriate simulation procedures directed towards refinement of models and forecasts.

REFERENCES

1. Tsay R.S. 2010. Analysis of Financial Time Series. – New York: John Wiley & Sons, Inc. 715 p.
2. Dovgyj S.O., Bidyuk P.I., Trofymchuk O.M., Savenkov O.I. 2011. Methods of Forecasting in DSS. – Kyiv: «Azymut-Ukraine». 608 p.
3. Harvey A.C. 1990. Econometric Analysis of Time Series. – Cambridge, MA: MIT Press. 402 p.
4. Gupta J.N.D., Forgionne G.A., Mora M. Intelligent Decision-making Support Systems. – London: Springer-Verlag, 2006. 522 p.
5. Bidyuk P.I., Gozhyj O.P., Korshevnyuk L.O. Computerized Decision Support Systems. – Mykolaiv: Black Sea Petro Mogyla State University, 2012. 380 p.
6. Bidyuk P.I., Romanenko V.D., Tymoshchuk O.L. Time Series Analysis. – Kyiv: Polytechnika, 2010. 317 p.
7. Gilks W.R., Richardson S., Spiegelhalter D.J. Markov Chain Monte Carlo in practice. – New York: Chapman & Hall/CRC, 2000. 486 p.
8. Zgurovsky M.Z., Bidyuk P.I., Terentyev O.M. Method of constructing Bayesian networks based on scoring functions // Cybernetics and System Analysis, 2008, No.2, pp. 219-224.
9. Jensen F.V., Nielsen Th.D. Bayesian Networks and Decision Graphs. – New York: Springer, 2007. 457 p.

ВИЯВЛЕННЯ ТА ЗАХОПЛЕННЯ ШВИДКІСНИХ РУХОМИХ ЦІЛЕЙ ГЛИБОКИМИ НЕЙРОННИМИ МЕРЕЖАМИ

Бєздєтний Д.Д., Данилов В.Я.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

Сучасні методи виявлення та захоплення швидкісних рухомих об'єктів базуються на використанні глибоких нейронних мереж та алгоритмів відстеження. У роботі представлено підхід до розпізнавання та трекінгу дронів на зображеннях і відеоряді за допомогою моделей сімейства YOLO та трекерів на основі фільтра Калмана. Навчання та тестування проводилося на спеціалізованих наборах даних із реальними сценаріями. Отримані результати демонструють високу точність і стабільність в умовах швидкого руху об'єктів. Наукова новизна полягає в інтеграції методів детекції та трекінгу для забезпечення точного відстеження об'єктів у режимі реального часу. Практична значимість підтверджується можливістю застосування в системах спостереження та контролю.

Ключові слова: виявлення об'єктів, трекінг, глибокі нейронні мережі, дрони, YOLO.

1. ВСТУП

У сучасному світі автоматизація процесів контролю та моніторингу вимагає використання передових технологій штучного інтелекту. Особливого значення набуває завдання виявлення та трекінгу швидкісних рухомих об'єктів у різних умовах. Такі задачі знаходять застосування в обороні, безпеці, логістиці та системах спостереження.

Метою даної роботи є розробка інтегрованого підходу до виявлення, трекінгу та вимірювання дистанції до швидкісних об'єктів за допомогою глибоких нейронних мереж і стереозору. У роботі використано модель YOLOv8 для детекції об'єктів на зображеннях і відео та трекери, такі як CSRT, для відстеження траєкторії руху. Вимірювання дистанції реалізовано з використанням стереоскопічного вимірювання, що дозволяє оцінювати тривимірні координати об'єктів у просторі.

Запропонований підхід поєднує швидку та точну обробку візуальних даних з можливістю отримання глибинної інформації про об'єкти. Для навчання моделей були використані спеціалізовані набори даних, що включають відеозаписи дронів у різних умовах, зокрема зі зміною освітлення, швидкості руху та напрямку польоту. Експериментальні дослідження продемонстрували, що інтеграція глибоких нейронних мереж із методами стереоскопічного вимірювання забезпечує високу точність виявлення та відстеження об'єктів, а також дозволяє ефективно оцінювати їхню відстань до камери.

Цінність роботи полягає в поєднанні сучасних методів виявлення та слідкування із тривимірною реконструкцією сцени для задачі виявлення швидкісних рухомих об'єктів. Практична значимість підтверджена можливістю використання запропонованого підходу в системах автоматизованого спостереження та аналізу руху.

2. РОЗГЛЯНУТІ МЕТОДИ

У сучасних дослідженнях виявлення та відстеження швидкісних об'єктів значну увагу приділяють використанню сучасних нейронних мереж і алгоритмів машинного зору. Зокрема, методи розпізнавання об'єктів, такі як Zero-Shot Detectors та Two-Shot Detectors, дозволяють

ефективно розпізнавати рухомі цілі, навіть у складних умовах. Одночасно алгоритми слідкування, такі як DeepSORT, ByteTrack, KCF і CSRT, забезпечують надійне відстеження траєкторій руху об'єктів, що рухаються з високою швидкістю. У межах даної роботи реалізовано кілька підходів до детекції та відстеження об'єктів, а також розрахунку відстані до них за допомогою монокулярних і стереоскопічних методів.

Основою комплексного підходу стало використання YOLOv8 — однієї з найсучасніших моделей глибокого навчання для розпізнавання об'єктів. Завдяки своїй архітектурі YOLOv8 здатна виконувати аналіз зображень у реальному часі, забезпечуючи високу точність навіть у складних умовах з мінливим освітленням або заднім планом. У межах роботи ця модель використовувалася для розпізнавання дронів у відеоряді.

Для задач трекінгу застосовувалися наступні алгоритми:

- DeepSORT використовує калманівський фільтр і апаратний аналіз ознак об'єктів, що дозволяє відстежувати об'єкти навіть за умов тимчасових втрат детекції.
- ByteTrack ефективно інтегрує інформацію про детекції об'єктів і їхнє пересування, зберігаючи точність трекінгу навіть у ситуаціях із великою кількістю об'єктів у кадрі.
- KCF (Kernelized Correlation Filters) забезпечує трекінг на основі кореляційних фільтрів із ядровими функціями, що дає змогу швидко й точно відстежувати об'єкти за умови стабільного освітлення та малої кількості змін у сцені.
- CSRT (Discriminative Correlation Filter with Channel and Spatial Reliability) перевершує інші кореляційні методи завдяки використанню просторово-канального аналізу, що дозволяє більш точно відстежувати об'єкти в умовах складних змін фону чи освітлення.

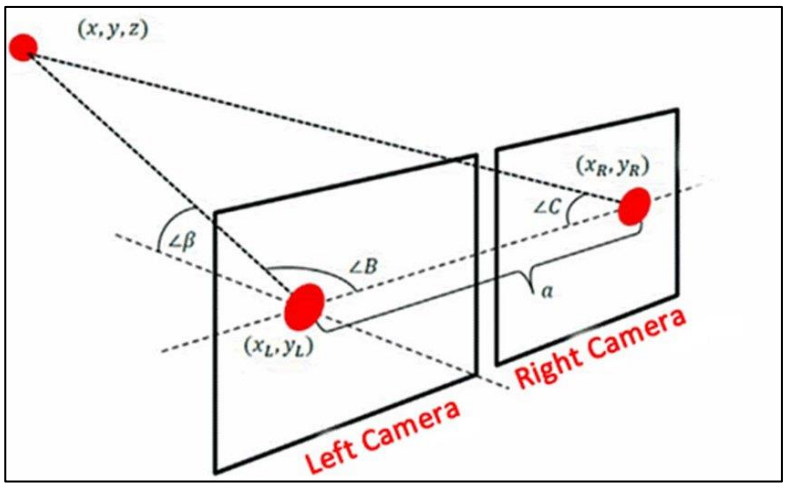


Рисунок 1. Схема стереоскопічного вимірювання

Для визначення відстані до об'єктів використано два підходи:

- Монокулярний аналіз, що базується на обчисленні відстані до об’єкта за відомими характеристиками камери та розмірами об’єкта.
- Стереоскопічний аналіз (рис. 1), який за допомогою двох камер дозволяє обчислювати тривимірні координати об'єктів у просторі, забезпечуючи високу точність при роботі в динамічних умовах.

Експериментальні результати показали, що інтеграція алгоритмів детекції та трекінгу дозволяє досягти високої ефективності виявлення швидкісних об'єктів. Застосування стереоскопічного та монокулярних методів вимірювання дозволило отримати точні дані про відстань до об'єктів, що є ключовим для практичних завдань, таких як моніторинг руху дронів чи автомобілів у реальному часі.

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для успішної побудови системи одним із ключових етапів був збір релевантних даних у належній якості. Для тренування моделі необхідно було сформувати якісний датасет із великою кількістю семплів, а для тестування системи — знайти відповідні валідаційні відео. Зібрані дані включали не лише зображення дронів (рис. 2), але й фотографії без них (рис. 3). Це дозволило моделі запам’ятати патерни інших об’єктів, що сприяло підвищенню точності виявлення саме дронів.

Рисунок 2. Приклад зображення дрона

Рисунок 3. Приклад додаткових зображень

Далі усі дані були правильно розмічені та підготовані до навчання моделі. Навчання моделі проходило 200 епох та було зупинено по досягненню вказаного порогу метрик функцій втрат. На рис. 4 наведемо приклад результату роботи моделі.

Рисунок 4. Приклад результату роботи детекції

Після цього слід було перевірити роботу трекінгових алгоритмів та визначитися, який саме краще запроваджувати для швидкісних цілей. Багато з них показали схожі високі результати, однак слід було сфокусуватися на тому, який найкраще працює із швидкими змінами вектору руху об’єктів та має гарну швидкодію, що відслідковуються. Було обрано алгоритм CSRT (рис. 5).

Рисунок 5. Результат роботи алгоритму CSRT

Після цього було зібрано фінальну комбінацію роботи моделі з визначенням відстані. Для визначення відстані методом стереозору ми маємо отримувати зображення з двох джерел, а саме камер, для яких ми знаємо відстань між ними та значення їх фокусної відстані. На рис. 6 приведемо приклад роботи системи.

Рисунок 6. Результат комбінації моделі моделі з алгоритмом визначення відстані

4. ВИСНОВКИ

У результаті проведеного дослідження було розроблено систему виявлення, відстеження та визначення відстані до швидкісних об’єктів, таких як дрони, з використанням сучасних методів комп’ютерного зору та алгоритмів машинного навчання.

На першому етапі вдалося сформувати якісний датасет, який включав не лише зображення дронів, але й інші об’єкти для підвищення точності детекції. Ретельна розмітка та підготовка даних забезпечили ефективне навчання моделі, що дозволило досягти високих показників точності під час виявлення дронів.

Далі було протестовано кілька трекінгових алгоритмів, серед яких найкращі результати для швидкісних об’єктів показав алгоритм CSRT. Його було обрано для інтеграції в систему завдяки здатності ефективно працювати зі швидкими змінами траєкторії руху.

Також реалізовано модуль визначення відстані до об’єктів на основі стереоскопічного методу. Використання двох камер із заданими параметрами дозволило точно обчислювати відстань до цілей у реальному часі.

Таким чином, розроблена система демонструє високу точність і швидкодію, що є критично важливим для роботи зі швидкісними об’єктами. Отримані результати можуть бути використані в подальших дослідженнях та розробках, зокрема в сфері безпеки, моніторингу об’єктів та автономних систем.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ultralytics YOLO Documentation. Source: https://docs.ultralytics.com.
2. Zhang, Y., Sun, P., Jiang, Y., et al. ByteTrack: Multi-Object Tracking by Associating Every Detection Box. arXiv preprint arXiv:2110.06864, 2021.
3. Wojke, N., Bewley, A., Paulus, D. Simple Online and Realtime Tracking with a Deep Association Metric. In IEEE International Conference on Image Processing (ICIP), 2017.
4. Lukezic, A., Vojir, T., Zajc, L., Matas, J., Kristan, M. Discriminative Correlation Filter with Channel and Spatial Reliability. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
5. Hartley, R., Zisserman, A. Multiple View Geometry in Computer Vision. Cambridge University Press, 2003.

КОМПЛЕКСНА МОДЕЛЬ ЕКОНОМІЧНОЇ БЕЗПЕКИ УКРАЇНИ

Биченков М.В., Мілявський Ю.Л.

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

Вирішене завдання оцінки рівня економічної безпеки держави. Економічна безпека України представлена у вигляді двох складових: стан економіки всередині країни та оцінка взаємодії з економіками країн-конкурентів. В основу комплексної моделі оцінки рівня економічної безпеки України покладено метод сценарного прогнозування. Комплексна модель враховує вплив зовнішньоекономічного і внутрішньоекономічного аспектів, які змодельовані з використанням відомих математичних методів: Хольта, АРІКС та МГУА. В якості вихідних даних використані закордонний The legatum prosperity index та вітчизняний індекс економічної безпеки.

Ключові слова: економічна безпека України, комплексна модель, сценарії економічного розвитку, пріоритетний ряд заходів, ефективний розвиток.

1. ВСТУП

Згідно указу Президента України від 2022 року була створена робоча група з питань розроблення плану заходів з післявоєнного відновлення та розвитку України. Однією з основних цілей даної робочої групи є підвищення рівня Національної безпеки України. На основі Стратегії національної безпеки України, в державі були розроблені стратегії розвитку складових національної безпеки. Однією з найважливіших складових національної безпеки – є економічна безпека .

На даний момент економічна безпека України розраховується за допомогою методики, яка розроблена Мінекономіки у 2013 році (далі – Методика 2013) [1]. В ній інтегральний показник економічної безпеки розраховується виходячи зі значень дев’яти його складових, а саме: продовольчої, виробничої, фінансової, макроекономічної інвестиційно-інноваційної, соціальної, демографічної, енергетичної та зовнішньоекономічної безпеки. Після детального розгляду методики розрахунку інтегрального показника економічної безпеки були виявлені деякі недоліки:

- методики носять рекомендаційний характер;
- в Стратегії економічної безпеки України на період до 2025 року аналіз здійснений лише за п’ятьма з дев’яти складових;
- для розрахунку економічної безпеки та її складових запропоновані усталені вагові коефіцієнти, механізм розрахунку яких відсутній;
- довоєнні вагові коефіцієнти оцінки рівня економічної безпеки України потребують перерахунку;
- методика, що розглядається, розраховує інтегральний показник економічної безпеки без врахування зовнішніх факторів та впливу економік конкурентів.

Таким чином, актуальним завданням є створення комплексної моделі економічної безпеки України, що була б позбавлена означених недоліків.

2. МЕТОДИКА РОЗРОБЛЕННЯ КОМПЛЕКСНОЇ МОДЕЛІ ЕКОНОМІЧНОЇ БЕЗПЕКИ УКРАЇНИ

Комплексна модель економічної безпеки України розроблена з врахуванням факторів, які б враховували не тільки розвиток національної економіки всередині країни (як в Методиці 2013), але і врахувала б фактор взаємного впливу економіки України та економік країн-конкурентів. Таким чином, в новій методиці враховано внутрішній і зовнішній аспекти економічної безпеки країни. Для відображення в комплексній моделі внутрішнього аспекту економічної безпеки було обрано індекс економічної безпеки [1], що запропонований фахівцями Мінекономіки. Для врахування зовнішньоекономічного аспекту – обраний The legatum prosperity index [2], який визнаний таким, що більш придатний серед інших за критеріями: наявність в складі рейтингу сукупності комплексних економічно впливових чинників, достатньо великий статистичний ряд оцінки економічних показників України, наявність оцінок ефективності розвитку економіки України в повоєнні роки.

Комплексну модель економічної безпеки України розроблено на основі методу сценарного прогнозування (рис. 1). Даний метод передбачає наступні кроки:

визначення горизонту прогнозу;
визначення ключових тенденцій і чинників;
ідентифікація екстремумів можливих результатів впливу відібраних чинників;
опис та оцінювання сценаріїв;
прогнозування розвитку сценаріїв;
вироблення рекомендацій щодо розвитку сценаріїв.

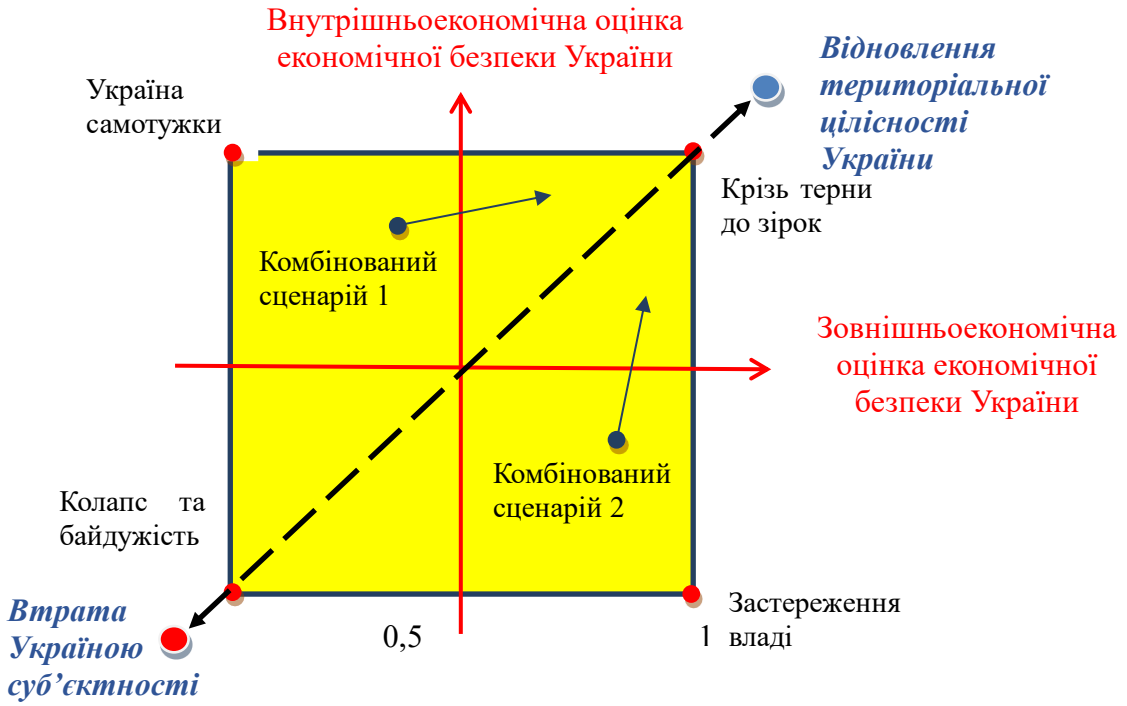


Рисунок 1. Зона можливих сценаріїв розвитку економічної безпеки України

В ході роботи було удосконалено метод сценарного прогнозування. Класичний метод сценарного прогнозування в залежності від кількості визначених комплексних впливових чинників має 2^n сценаріїв. Де n – кількість впливових комплексних чинників. В даному випадку за участі експертів можна сформувати 4 можливих сценарії. Але на стадії прогнозування можливого розвитку подій зрозуміло, що реальний розвиток подій скоріш за

все не відбудеться за одним з чотирьох сценаріїв. Розвиток подій буде відбуватись за певним комбінованим варіантом сценарію, який вбере в себе особливості декількох прогнозованих сценаріїв. Тому пропонується дані запропоновані сценарії економічного розвитку ситуації довкола країни вважати "опорними". Опорні сценарії є крайніми ідеалізованими варіантами розвитку подій.

Таким чином, між визначеними опорними сценаріями було отримано зону припустимих комбінованих варіантів сценаріїв, де з високою часткою ймовірності буде знаходитись реальний сценарій розвитку подій в майбутньому. Даний комбінований варіант сценарію буде враховувати дві оцінки рівня економічної безпеки України: внутрішньоекономічну та зовнішньоекономічну.

З іншого боку, з метою побудови моделей зовнішньоекономічної та внутрішньоекономічної складових рівня економічної безпеки, було проведено порівняльний аналіз індуктивних методів математичного моделювання, таких як: метод найменших квадратів, метод групового урахування аргументів [3], метод авторегресії з інтегрованим ковзним середнім [4]. Після аналізу зазначених методів прийняте рішення щодо їх використання в якості конкуруючих для побудови моделей внутрішньоекономічної та зовнішньоекономічної складових економічної безпеки України.

З метою порівняння якості побудованих моделей обрані метрики для їх перевірки. В якості метрик запропоновано використовувати: MAPE (Середня абсолютна похибка у відсотках), MSE (Середньоквадратична похибка), R^2 (Коефіцієнт детермінації), AIC (Інформаційний критерій Акаїке) та DW (Критерій Дарбіна-Уотсона) [5,6].

Крім того, в роботі для визначення опорних сценаріїв рівня економічної безпеки України, їх характеристик, для визначення пріоритетного ряду заходів щодо ефективного розвитку рівня економічної безпеки відповідно сценаріїв - були застосовані експертні оцінки. Для узгодження суджень експертів було застосовано метод редукторів ступенів свободи [7].

3. ПОБУДОВА КОМПЛЕКСНОЇ МОДЕЛІ ТА ОЦІНКА РІВНЯ ЕКОНОМІЧНОЇ БЕЗПЕКИ УКРАЇНИ

Для визначення моделей оцінки рівня економічної безпеки за вітчизняним та закордонним індексом розроблено програмний продукт: інтелектуальна система підтримки прийняття рішень ІСППР "Моделювання та прогнозування економічної безпеки". Для побудови моделей відновлення функціональної залежності в ІСППР застосовуються такі методи як: МГУА (метод групового урахування аргументів) та МНК (метод найменших квадратів). В якості критеріїв якості для проведення селекційного відбору моделей було застосовано наступні критерії якості: MSE (середня квадратична похибка), MAPE (середня абсолютна відсоткова похибка), AIC (інформаційний критерій Акаїке), DW (критерій Дарбіна-Уотсона) та коефіцієнт детермінації.

Приклад застосування зазначеного програмного продукту для визначення системи рівнянь за складовими внутрішньоекономічного рівня економічної безпеки показано в таблиці 1. Далі були визначені моделі інтегральних показників за вітчизняним та закордонним індексом, що зазначені в таблиці 2.

Після отримання значень прогнозу вітчизняного, так і закордонного індексів (моделі яких були побудовані з використанням наступних методів: МГУА, АРІКС та Хольта) необхідно визначити пріоритетність заходів для підвищення рівня економічної безпеки України враховуючи прогнозований стан рівня економічної безпеки України. Для цього була створена координатна площина, вісі якої відповідають, відповідно, за внутрішній та зовнішній аспект економічної безпеки. За допомогою експертного обговорення були визначені 4 опорні сценарії

економічного розвитку держави, що позначені синіми точками на рисунку 2 (в свою чергу червоними зазначені точки прогнозного економічного стану держави у 2025 та 2030 році).

Таблиця 1. Моделі відтворення функціональної залежності складових вітчизняного показника економічної безпеки України

Складова вітчизняного показника	Модель відновлення функціональної залежності складової
Продовольча	$y_1 = 70.27 + 17.18x_1 + 7.67x_2 + 19.25x_3 + 11.51x_5 - 7.23x_8 - -21.8x_1x_6 - 11.32x_2^2x_4 - 24.54x_1x_2^2$
Виробнича	$y_2 = 38.2 + 7.22x_1 + 8.41x_3 + 17.01x_4 - 23.61x_5 + 8.36x_7 - -6.69x_8 + 11.12x_2x_5 + 18.06x_5x_6$
Фінансова	$y_3 = 44.56 + 15.44x_1 - 1.02x_2 - 1.94x_3 + 30.7x_4 + 1.55x_5 - -3.79x_6 - 61.68x_1x_4 - 16.44x_3x_7 + 6.33x_8x_9$
Макроекономічна	$y_4 = 103.05 - 0.88x_1 + 0.91x_2 - 9.31x_3 - 51.63x_5 - 52.91x_6 - -82.36x_7 + 101.74x_5x_7 - 7.03x_4x_8 - 15.06x_7x_9$
Інвестиційно-інноваційна	$y_5 = 24.6 + 23.3x_1 - 2.25x_2 - 6.77x_4 + 3.14x_6 + 13.42x_7 - -27.12x_1x_7 - 9.03x_3x_4x_5 + 6.98x_3^2x_4x_5$
Соціальна	$y_6 = 63.1 - 9.32x_2 + 6.68x_4 - 1.64x_5 - 3.49x_1^2 + 8.08x_3x_5 - 8x_1x_7$
Демографічна	$y_7 = 35.07 + 1.96x_1 + 13.92x_2 + 25.39x_3 - 7.72x_4 - 7x_6 - -6.27x_7 - 5.83x_3x_4 - 5.22x_5x_7$
Енергетична	$y_8 = 67.26 - 13.73x_2 - 25.17x_3 - 13.3x_4 - 2.13x_5 - 13.49x_7 + +16.76x_2x_4 + 27.24x_4x_7 - 9.46x_6^2x_7 + 8.93x_1x_3x_7$
Зовнішньоекономічна	$y_9 = -1.32 - 3.31 * x_1 + 10.83x_2 + 37.13x_4 - 4.23x_6 + 14.4x_8 + +25.34x_3^2 - 21.79x_1x_7 - 9.33x_8^3 + 38.18x_5x_7^2$

Таблиця 2. Моделі інтегрального показника економічної безпеки як за вітчизняним, так і за закордонним індексом

	Модель інтегрального показника
Вітчизняний індекс	$z_1 = 27.72 - 0.28y_1 + 0.099y_2 + 0.11y_3 + 0.09y_4 - 0.23y_5 + 0.16y_7 + +0.12y_8 - 0.87y_9 + 0.013y_1y_9 + 0.0085y_5y_6$
Закордонний Індекс	$z_2 = 0.22 + 0.8y_1 - 0.28y_2 - 0.02y_4 - 0.42y_5 + 0.19y_6 + 0.1y_7 + 0.17y_8 - -1.42y_1y_4 - 0.27 * y_2^2y_4 + 2.57y_4^2y_5^2 + 4.61y_3y_5y_7y_8^2$

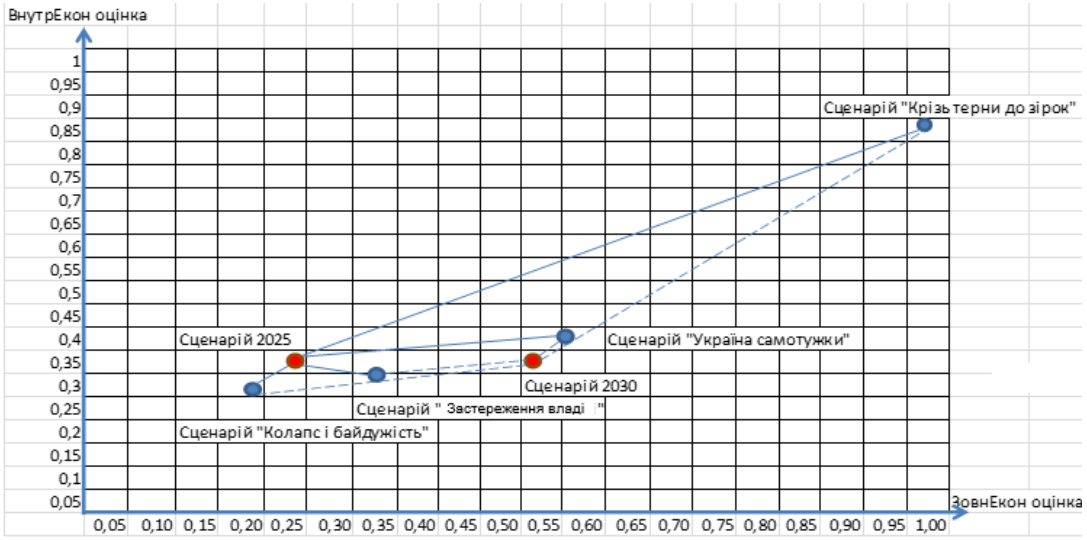


Рисунок 2. Координатна площина визначення пріоритетності заходів для підвищення рівня економічної безпеки України

Положення опорних сценаріїв на координатній вісі було визначено за допомогою узгоджених оцінок експертів, після чого виходячи з експертних оцінок та моделей оцінки інтегральних показників, що зазначені в таблиці 2 були визначені оцінки інтегральних показників для кожного сценарію. Результати приведених дій продемонстровані в таблиці 3.

Таблиця 3. Значення опорних сценаріїв за внутрішньоекономічним та зовнішньоекономічним показниками

Сценарії	Узгоджена зовнішньоекономічна оцінка за моделлю функціональної залежності інтегрального показника індексу процвітання України	Узгоджена внутрішньоекономічна оцінка за моделлю функціональної залежності економічної безпеки за вітчизняним показником
Крізь терни до зірок	1	0,86
Київ самотужки	0,59	0,42
Застереження владі	0,36	0,33
Колапс та байдужість	0,23	0,30

Для визначення пріоритетності економічних заходів за кожним з чотирьох опорних сценаріїв було проведено експертне опитування десяти експертів, що зазначили свою думку на рахунок пріоритетності заходів по кожному з опорних сценаріїв. Для визначення узгодженої пріоритетності групи експертів було застосовано метод редукторів ступенів свободи.

В подальшому, отримана узгоджена оцінка пріоритетності заходів по опорних сценаріях була використана для визначення важливості заходів по сценаріях прогнозного стану економічної безпеки держави, яка була змодельована з використанням комплексної моделі економічної безпеки України. Для цього було використано відомості щодо відстані змодельованого економічного положення держави до опорних сценаріїв та інформацію щодо пріоритетного ряду заходів зазначених опорних сценаріїв. При цьому була використана гіпотеза, що чим далі опорний сценарій до модельованого економічного стану держави, тим менше його вплив на кінцеве ранжування за пріоритетністю.

За результатами оцінки пріоритетності заходів станом на 2025 рік для поліпшення рівня економічної безпеки України за вутрішньоекономічною складовою пріоритетними заходами визначені: мобілізація трудових та творчих ресурсів; забезпечення верховенства закону в усіх сферах життя; модернізація державного сектору економіки. За зовнішньоекономічною складовою: відстоювання позиції невідворотності покарання з боку світової спільноти щодо застосування сили чи погрози силою; відстоювання відповідальності за втручання у внутрішні справи інших держав з боку світової спільноти; поліпшення міжнародного іміджу держави.

4. ВИСНОВКИ

Одним з основних питань післявоєнного ефективного розвитку держави – є забезпечення розвитку економічної безпеки України. Для вирішення поставленого завдання було розроблено комплексну модель економічної безпеки України, яка дозволила оцінити рівень економічної безпеки держави, а також встановити перелік пріоритетних заходів для поліпшення економічної безпеки держави.

В основу комплексної моделі економічної безпеки України було покладено метод сценарного прогнозування. Даний метод дозволив поєднати впливові показники економічної безпеки України, такі як: внутрішньоекономічна (оцінка рівня економічної розвитку

всередині країни), а також зовнішньоекономічна оцінка рівня економічної безпеки України (оцінка рівня економічної безпеки України з точки зору її конкурентоспроможності).

В роботі розроблено алгоритм визначення ряду пріоритетних заходів змодельованого з використанням комплексної моделі економічної безпеки України ймовірного сценарію розвитку країни з врахуванням відстаней між ним та опорними варіантами сценаріїв та наявної інформації щодо пріоритетного ряду заходів за опорними сценаріями.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Методичні рекомендації щодо розрахунку рівня економічної безпеки України, затверджені наказом Міністерства економічного розвитку і торгівлі України від 29.10.2013 року № 1277 URL: https://www.me.gov.ua/Documents/List?lang=uk-UA&id=d4c96730-ea46-4ebdba9260631a3e2e69&tag=MetodichniRekomendatsiiMakroekonomika&isSpecial=tre

2. LEGATUM PROSPERITY INDEX 2016 – METHODOLOGY REVIEW [Електронний ресурс] // LEGATUM INSTITUTE. – 2016. – Режим доступу до ресурсу: https://www.prosperity.com/application/files/1914/7819/5146/Legatum_Prosperity_Index_Methodology_Report.pdf.

3. Скакаліна О. Порівняльний аналіз прогнозуючих моделей на основі гібридів методу групового урахування аргументів [Електронний ресурс] / Олена Скакаліна // Полтавський політехнічний університет імені Юрія Кондратюка. – 2022. – Режим доступу до ресурсу: https://reposit.nupp.edu.ua/bitstream/PoltNTU/11303/1/%D0%A1%D0%9A%D0%90%D0%9A%D0%90%D0%9B%D0%86%D0%9D%D0%90_%D0%A2%D0%95%D0%97%D0%98_%D0%9A%D0%A3%D0%A1%D0%A12022.pdf.

4. A Review of ARIMA vs. Machine Learning Approaches for Time Series Forecasting in Data Driven Networks / V. I.Kontopoulou, A. D. Panagopoulos, I. Kakkos, G. K. Matsopoulos. // future internet. – 2023. – №15. – С. 31.

5. Chicco D. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation / D. Chicco, M. Warrens, G. Jyrman. // PeerJ Computer Science. – 2021. – С. 1–24.

6. Brewer M. J. The relative performance of AIC, AICC and BIC in the presence of unobserved heterogeneity / M. J. Brewer, A. Butler, S. L. Cooksley. // Methods in Ecology and Evolution. – 2016. – С. 679–692.

7. Voronin A. Multi-Criteria Decision Making for the Management of Complex Systems / Albert Voronin. – Pennsylvania: IGI Global, 2017. – 201 с. – (IGI).

МОДЕЛЮВАННЯ ТА АНАЛІЗ ТРАНСПОРТНИХ ПОТОКІВ ДЛЯ МІСЬКОГО ТРАФІКУ З ВИКОРИСТАННЯМ ТЕОРІЇ МАКСИМАЛЬНОГО ПОТОКУ

Боднар М.С.[1], Статкевич В.М.[2]

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Навчально-науковий інститут прикладного системного аналізу, кафедра математичних методів системного аналізу, Київ, Україна

[1] poit25@ukr.net, [2] mstatckevich@yahoo.com [0000-0001-5210-9890]

При моделюванні руху автомобілів в умовах великого міста важливо врахувати як загальні потоки транспорту та пропускну здатність доріг з урахуванням теорії максимального потоку, так і аспекти нижчого рівня: кількість смуг дороги і можливість зміни смуг під час руху, маневри на перехрестях. Ці аспекти можуть значно впливати на загальні показники потоку. Розроблене програмне забезпечення надає ефективний інструмент для моделювання транспортного потоку і формулювання практичних рекомендацій.

Ключові слова: максимальний потік, моделювання потоку автомобілів, задача про максимальний потік, алгоритм Едмондса-Карпа, карта, рух по смугам, маневр на перехресті.

1. ВСТУП

Оптимізація транспортних потоків є важливою складовою сучасної транспортної інженерії та управління, яка відіграє ключову роль у підвищенні ефективності транспортних систем. Історично, з розвитком промисловості та зростанням міських агломерацій, потреба в оптимізації транспортних потоків стала особливо актуальною. Від ефективності транспортних потоків залежить не тільки швидкість пересування та доставки товарів, але й економічне благополуччя регіонів, стан довкілля та якість життя населення.

Головною метою цього дослідження є розробка та аналіз алгоритмів і програмного забезпечення, які можуть допомогти в моделюванні транспортних потоків для надання можливості їх оптимізації, забезпечуючи більш ефективне та доцільне використання транспортних мереж. Це включає в себе знаходження максимального потоку, який може проходити через мережу без перевантаження, та визначення оптимальних маршрутів для різних видів транспорту.

Результати роботи можуть бути використані для прийняття рішень у реальних застосуваннях, таких як транспортні системи, логістика, мережева маршрутизація та розподіл ресурсів.

2. МОДЕЛЮВАННЯ ТРАНСПОРТНОГО ПОТОКУ

2.1. Задача про максимальний потік

Однією з основних задач в оптимізації транспортних потоків є задача про знаходження максимального потоку, яка є фундаментальною в теорії оптимізації та теорії графів. Вона полягає у знаходженні такого потоку за транспортною мережею, щоб сума потоків на ребрах, які виходять з джерела, або, що означає те ж саме, сума потоків на ребрах, які приходять до стоку, була максимальною. Ця задача має широке практичне застосування, включаючи

планування та управління транспортними системами, де вона допомагає визначити оптимальні шляхи для перевезення пасажирів та вантажів, зменшуючи час у дорозі та витрати на транспортування.

Задача про максимальний потік була вперше сформульована у 1954 році Т. Е. Гаррисом та Ф. С. Россом як спрощена модель радянського залізничного транспортного потоку [1]. У 1955 році Лестер Р. Форд-молодший та Делберт Р. Фалкерсон створили перший відомий алгоритм – алгоритм Форда-Фалкерсона [2], який базується на пошуку збільшуючих шляхів, як описано вище.

Задача про максимальний потік може бути сформульована так: дано орієнтований граф G (також називається мережею) з джерелом та стоком, де кожне ребро має додатну пропускну здатність, яка визначає максимальну кількість потоку, що може проходити через нього. Треба знайти спосіб надсилання якомога більшого потоку від джерела до стоку з урахуванням обмежень на пропускну здатність. Потік на кожному ребрі повинен бути невід'ємним і задовольняти умови збереження потоку: загальний потік, що входить у вузол (крім джерела та стоку), повинен бути рівним загальному потоку, що виходить з цього вузла. Більш формально, **потік** (англ. *flow*) у G – це двовимірна дійснозначна функція на декартовому квадраті множини вершин $f: V \times V \to \mathbb{R}$, яка визначає кількість потоку, що проходить по кожному ребру. Така функція задовольняє наступним умовам:

1) **обмеження пропускної здатності** (*capacity constraint)*:
$$\forall u, v \in V: f(u, v) \leq c(u, v),$$
2) **антисиметричність** (*skew symmetry*):
$$\forall u, v \in V: f(u, v) = -f(v, u),$$
3) **збереження потоку** (*flow conservation*):
$$\forall u \in V \setminus \{s, t\}: \sum_{v \in V} f(u, v) = 0.$$

Для знаходження максимального потоку можна використати алгоритм Едмондса-Карпа [3], який є вдосконаленою версією методу Форда-Фалкерсона, що використовує пошук у ширину (англ. Breadth-first search, BFS) для знаходження найкоротших збільшувальних шляхів у залишковій мережі. У кожній ітерації визначається шлях від джерела до стоку, уздовж якого збільшується потік, доки це можливо. Такий підхід забезпечує гарантовану збіжність алгоритму до оптимального рішення. Завдяки використанню пошуку в ширину алгоритм Едмондса-Карпа має часову складність $O(|V| \cdot |E|^2)$, де $|V|$ – кількість вершин, а $|E|$ – кількість ребер у мережі, що робить його ефективним для задач середнього масштабу.

2.2. Моделювання міського транспорту

Моделювання транспортної мережі є ключовим етапом у процесі розв'язання задачі руху транспортних потоків. Воно дозволяє представити реальну транспортну систему у вигляді математичної моделі, яка може бути аналізована та оптимізована з використанням різних алгоритмів.

При моделюванні транспортної мережі підхід аналогічний підходу в задачі про максимальний потік, а мережа зазвичай також представляється у вигляді графа, де вузли відповідають пунктам відправлення, пунктам призначення або перехрестям, а ребра – транспортним шляхам між цими пунктами. Кожне ребро має певну пропускну здатність та вартість перевезення.

Формально, нехай $G = (V, E)$ представляє граф транспортної мережі, де V – множина вузлів, а E – множина ребер. Кожне ребро $(i, j) \in E$ має пропускну здатність c_{ij}.

На рис. 1 наведений приклад графу транспортної мережі, отриманої з карти міста.

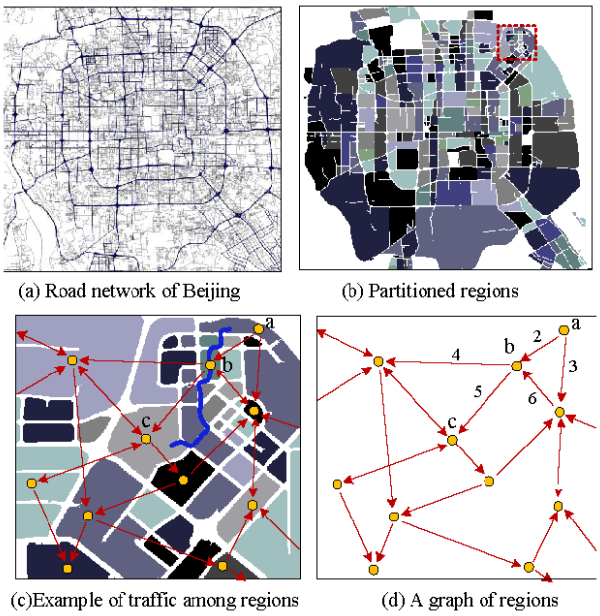


Рисунок 1. Приклад створення графу транспортної мережі на основі реальної карти [4]

Для моделювання транспортної мережі був розроблений власний алгоритм, вхідними даними якого є детальна карта міста з вказанням характеристик і типів доріг і який змінює топологію початкового графа. Після виконання алгоритма мережа відображає допустимі маневри автомобілів на перехрестях чи поворотах, відповідно до правил дорожнього руху.

Алгоритм працює на орієнтованому мультиграфі, де вершини мають географічні координати, а кожне ребро відповідає смузі руху. Алгоритм замінює вершину, яка відповідає за вузол перехрестя, на множину вхідних і вихідних вершин для даного перехрестя. Вхідні вершини для даного перехрестя позначають точки в'їзду на перехрестя чи початку маневру (повороту), а вихідні – точки виїзду або завершення маневру (повороту). Це дозволяє моделювати траєкторії руху з урахуванням геометрії перехресть і напрямків смуг. Розглянемо перехрестя вулиць Ризької, Дорогожицької та Парково-Сирецької, м. Києва (50.4702674, 30.4436834), зображеному на рис. 2 зліва. До перехрестя прилягають три дороги: вул. Ризька (3 смуги в обидва напрямки), вул. Дорогожицька (2 смуги в обидва напрямки) та вул. Парково-Сирецька (2 смуги в кожному напрямку). Справа на рис. 2 можна побачити готову модель цього перехрестя з врахованими маневрами для кожної смуги.

Рисунок 2. Модель перехрестя

Алгоритм перевіряє кожну пару вхідних і вихідних вершин для всіх перехресть. Для різних доріг аналізуються вектори напрямків, кути поворотів і відповідність смуг. Для однієї дороги перевіряється перетин траєкторій. На основі цих перевірок вершини з'єднуються, динамічно визначаючи можливі маневри.

В алгоритмі перевіряється можливість з'єднання кожної пари вхідної (enter_node) та вихідної (exit_node) вершин на перехресті, враховуючи напрямки доріг і відповідність смуг.

Якщо вершини належать різним дорогам, обчислюються вектори напрямків (ent_dir, ext_dir), кут θ і напрямок повороту turn. При $|\theta| > \pi/6$ перевіряється, чи відповідають смуги, після чого вершини з'єднуються залежно від напрямку повороту наліво або направо (left/right) та параметрів доріг (L, TL). Якщо вершини належать одній дорозі, враховується збіг траєкторій і відповідність смуг.

Ключові елементи:

1) enter_node, exit_node – вхідні/вихідні вершини;
2) L – номер смуги (першою вважається крайня права смуга, номер зростає із зміщенням наліво);
3) TL – загальна кількість смуг дороги;
4) ent_dir, ext_dir – напрямки доріг;
5) angle, turn – кут і напрямок повороту.

Обчислення кута та напрямку між векторами (рис. 3) включає розрахунок кута θ через функцію atan2:

$$\theta = \text{atan2}\left(\overrightarrow{TF}_x \times \overrightarrow{TG}_y - \overrightarrow{TF}_y \times \overrightarrow{TG}_x, \overrightarrow{TF}_x \times \overrightarrow{TG}_x - \overrightarrow{TF}_y \times \overrightarrow{TG}_y\right).$$

Кут нормалізується до діапазону $[-\pi, \pi]$, додаючи або віднімаючи 2π, якщо $\theta > \pi$ або $\theta < -\pi$. Після нормалізації, визначається напрямок повороту наступним чином: якщо $\theta > \pi$, то поворот «наліво», інакше – «направо».

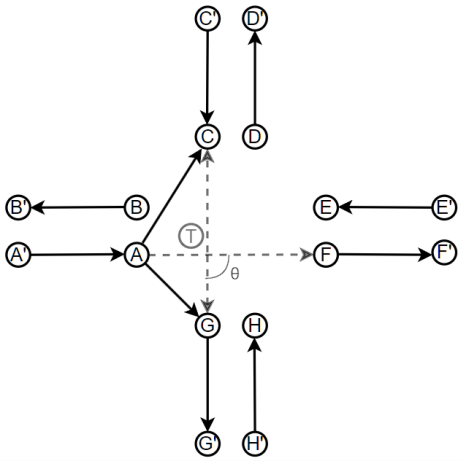


Рисунок 3. Кут повороту

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для програмної реалізації моделювання були використані геопросторові дані, надані проектом OpenStreetMap (OSM), та мова програмування Python з додатковими модулями, такими як NetworkX (для роботи з графами) та OSMnx (для завантаження та обробки даних).

У роботі було розроблено графову модель міської транспортної мережі на основі даних OpenStreetMap. В процесі моделювання дані завантажуються та оброблюються із використанням бібліотеки OSMnx. Основну увагу приділено аналізу ключових атрибутів

дорожньої мережі, таких як довжина сегмента дороги, кількість смуг, обмеження швидкості, тип дорожнього покриття та інші.

Для побудови графа було видалено сервісні дороги, спрощено топологію шляхом об’єднання ділянок з однаковими атрибутами, а також усунено зворотні ребра. Виконано декомпозицію доріг на окремі смуги, кожна з яких представлена окремим ребром у графі, що дозволяє деталізувати транспортні потоки.

Розроблено алгоритм зміни топології перехресть із додаванням вхідних та вихідних вершин для кожної смуги руху (детальніше див. п. 2.2). Це забезпечило реалістичне врахування маневрів на перехрестях відповідно до правил дорожнього руху: зокрема, розрізняються повороти наліво і повороти направо, розрізняються повороти з головної дороги на другорядну та повороти з другорядної дороги на головну. Пропускна здатність ребра обчислюється з урахуванням максимальної швидкості та відповідних коефіцієнтів.

На основі побудованої моделі виконано моделювання транспортних потоків, що дозволило ідентифікувати вузькі місця (bottlenecks) у дорожній інфраструктурі міста. Виявлено проблемні ділянки та вулиці, які обмежують пропускну здатність транспортної системи, що є основою для оптимізації транспортних потоків у міських умовах.

Етапи моделювання можна побачити на рис. 4: 1) початковий неспрощений мультиграф (лівий верхній кут); 2) граф без сервісних доріг (правий верхній кут); 3) спрощений граф (лівий нижній кут); 4) граф після декомпозиції доріг на окремі смуги та зміни топології перехресть (правий нижній кут).

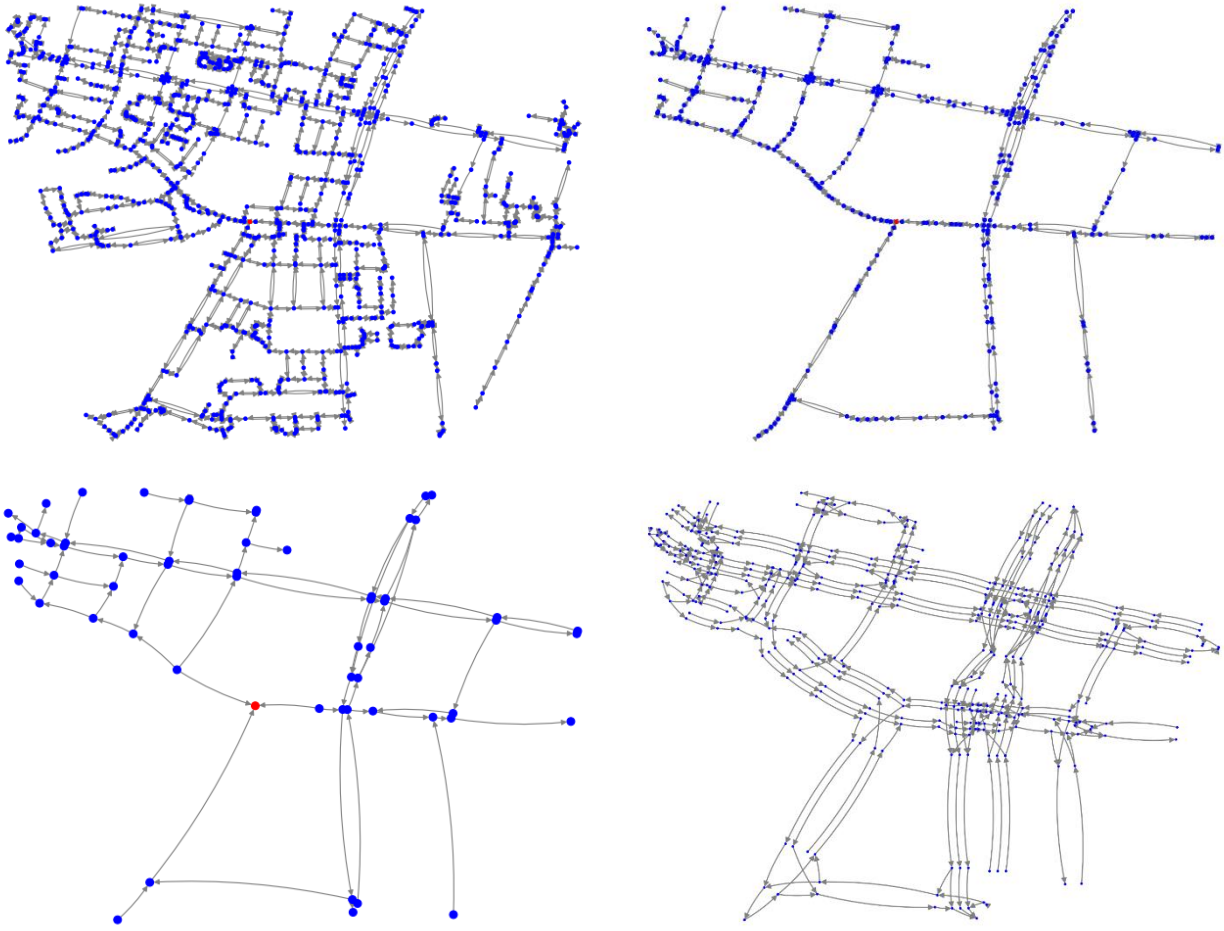


Рисунок 4. Етапи створення мережі

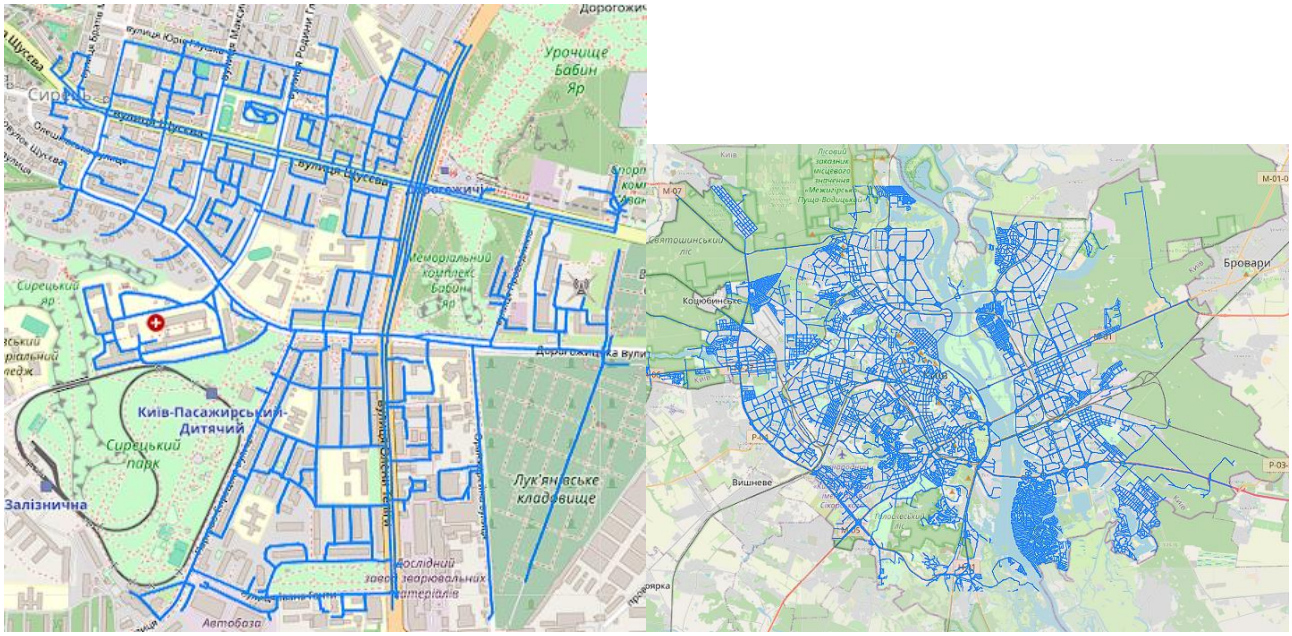


Рисунок 5. Мережі доріг Києва

На рис. 5 можна побачити початкову частину мережі доріг, що була змодельована (зліва) та мережу доріг Києва (справа).

В запровадженій моделі також наявні пропускні здатності кожного з ребер, що дозволяє програмно знайти максимальний потік. Для знаходження максимального потоку в цій мережі необхідно вибрати джерело та стік. Джерелом можуть бути ключові транспортні вузли, які відповідають за великий обсяг дорожнього трафіку, наприклад, спальні райони Києва, де зосереджено велику кількість житлових комплексів і житлових масивів. Стоком можуть виступати вузли в ділових районах або в центрі міста, де зосереджена основна економічна діяльність та вищий рівень інтенсивності транспорту. Також таку систему можна використовувати для збільшення товарообігу між містами або на підприємствах.

Для знаходження вузьких (або проблемних) місць мережі можна використати теорему про «максимальний потік-мінімальний розріз» (max-flow min-cut theorem) [5]. За допомогою цієї теореми було вирішено проблеми заторів у місті Кота Кінабалу (Сабах), що продемонстрували у своїй роботі Abdullah N. та Hua T. K. (2017) [6]. Вони виявили максимальний потік та вузькі місця в мережі, що дозволило інженерам з транспорту визначити, які дорожні об’єкти потребують поліпшення для мінімізації проблем з трафіком.

Зазначимо, що окрім вказаного підходу до показників потоку і ефективності транспортної системи в цілому, який можна назвати макроскопічним, розглядають також і інші підходи, зокрема, мікроскопічний, коли в якості об’єктів вибираються, наприклад, окремі автомобілі і вихідною точкою підходу є фізичні правила руху конкретного автомобіля з урахуванням правил дорожнього руху, швидкості, можливості зміни смуги тощо. Тоді усереднене значення потоку вже розраховується на базі сукупності великої кількості окремих автомобілів із заданими правилами руху. Такий мікроскопічний підхід може базуватись, зокрема, на концепції клітинних автоматів. Ця концепція клітинних автоматів є перспективною і багато авторів приділяють їй велику увагу. Один з прикладів – моделювання руху автомобілів. Інший приклад – моделювання руху пішоходів в серії робіт О.С. Макаренко із співавторами (див., наприклад, [7]).

4. ВИСНОВКИ

Проведене моделювання транспортних потоків у міській дорожній мережі дозволяє сформулювати низку практичних рекомендацій щодо оптимізації роботи транспортної системи. Зокрема, для підвищення пропускної здатності можуть бути запропоновані зміни максимальних швидкостей на окремих ділянках доріг, розширення вузьких ділянок, перерозподіл смуг руху, а також створення спеціальних смуг для громадського транспорту або велосипедистів. Крім того, важливо оптимізувати роботу світлофорів для зменшення часу очікування на перехрестях, встановити пріоритетні напрямки руху на ключових транспортних вузлах, розглянути можливість додавання нових розв'язок у місцях з високою концентрацією потоків та модернізувати пішохідні переходи для мінімізації їхнього впливу на транспортний рух. Виявлені вузькі місця можуть стати основою для стратегічних змін в інфраструктурі міста, що знизить затори та підвищить ефективність транспортної системи.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Harris T. E., Ross F. S. Fundamentals of a method for evaluating rail net capacities. Santa Monica, CA : RAND Corporation, 1955. 63 p. URL: https://apps.dtic.mil/sti/citations/AD0093458
2. Ford L. R., Fulkerson D. R. Maximal flow through a network. Canadian Journal of Mathematics. 1956. Vol. 8. P. 399–404. URL: https://doi.org/10.4153/cjm-1956-045-5
3. Edmonds J., Karp R. M. Theoretical improvements in algorithmic efficiency for network flow problems. Journal of the ACM. 1972. Vol. 19, no. 2. P. 248–264. URL: https://doi.org/10.1145/321694.321699
4. Liu W., Zheng Y., Chawla S., Yuan J., Xing X. Discovering spatio-temporal causal interactions in traffic data streams. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August, 2011, P. 1010–1018. URL: https://dl.acm.org/doi/abs/10.1145/2020408.2020571
5. Introduction to algorithms / T. H. Cormen et al. 3rd ed. Cambridge, MA, USA : The MIT Press, 2009. 1312 p.
6. Abdullah N., Hua T. K. Using Ford-Fulkerson algorithm and max flow-min cut theorem to minimize traffic congestion in Kota Kinabalu, Sabah. Journal of Information System and Technology Management, 2017, 2(4), P. 18–34. URL: http://www.jistm.com/PDF/JISTM-2017-04-06-02.pdf
7. Макаренко О.С., Крушинський Д.А. Моделювання руху пішоходів на основі клітинних автоматів. Системні дослідження та інформаційні технології, 2010, № 1, С. 100–109.

СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ДЛЯ КОМПЛЕКСНОГО АНАЛІЗУ ФІНАНСОВОГО РИНКУ

Болдарев Є.А.[1], Кузнєцова Н.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] boldarev.yehor@lll.kpi.ua, [2] n.kuznietsova@kpi.ua

Дана робота на меті має розробку системи підтримки прийняття рішень для комплексного аналізу фінансового ринку. Основна мета цієї системи — забезпечити точне прогнозування динаміки фінансових секторів та їх індустрій, а також прогнозування зміни цін активів на основі аналізу історичних даних, використовуючи сучасні моделі машинного навчання, таких як рекурентні нейронні мережі типу Long Short-Term Memory (LSTM). Особливістю запропонованої системи є урахування ентропії фінансового ринку при прогнозуванні та можливість агрегування результатів на рівні секторів або індустрій для виявлення загальних тенденцій, що дозволить фахівцям ухвалювати більш обґрунтовані рішення.

Ключові слова: фінансовий ринок, СППР, ентропія, LSTM.

1. ВСТУП

Сучасний фінансовий ринок є дуже складною і динамічною системою, що піддається впливу безлічі факторів — від макроекономічних показників до поведінки окремих компаній ринку. Прийняття ефективних рішень в таких умовах вимагає використання аналітичних інструментів, здатних не лише швидко обробляти великі обсяги інформації, але й прогнозувати зміни на ринку. У цьому контексті, системи підтримки прийняття рішень (СППР) стають корисним інструментом для аналізу фінансової галузі.

СППР є критично важливим інструментом для осіб, що приймають швидкі стратегічні рішення в полі фінансового ринку. Такими особами можуть бути фінансові аналітики, портфельні менеджери, інвестори, керівники фінансових установ та корпоративні лідери, які відповідають за управління активами та капіталом. СППР надає таким фахівцям змогу отримувати поглиблений аналіз та прогнози, що ґрунтуються на об'єктивних даних та потужних математичних моделях. Це дозволяє їм зменшити ризики та ухвалювати більш виважені рішення щодо інвестиційної стратегії, розподілу активів і реагування на ринкові коливання.

2. ВИБІР МОДЕЛІ ДЛЯ ПРОГНОЗУВАННЯ

На сьогоднішній день існує багато інструментів та моделей для прогнозування часових рядів на фінансових ринках. До найвідоміших можна віднести моделі ARIMA, метод опорних векторів, градієнтний бустинг та нейронні мережі різних типів (рекурентні, згорткові, глибокі або нечіткі). В цьому дослідженні використовується рекурентна нейронна мережа LSTM через ряд її основних властивостей та можливостей.

1. Обробка нерегулярних трендів і шумів. LSTM моделі, на відміну від класичних рекурентних нейронних мереж, здатні фільтрувати несуттєві коливання, фокусуючись на важливих шаблонах у даних. Це робить їх більш стійкими до шумів, які часто зустрічаються у фінансових даних [1].

2. Ефективна обробка великих обсягів даних. У багатьох випадках, прогнозування цін акцій вимагає обробки великих обсягів даних, особливо при роботі з кількома акціями або секторами ринку. LSTM моделі добре масштабуються та можуть обробляти великі обсяги послідовних даних, що робить їх придатними для прогнозування тенденцій як на рівні окремих акцій, так і на рівні галузей [1].
3. Успішне застосування в фінансових задачах. LSTM отримала значну популярність в задачах прогнозування часових рядів у фінансовій сфері через їхню ефективність у порівнянні з іншими моделями. LSTM забезпечують кращу точність у порівнянні з традиційними методами, такими як ARIMA, SARIMA та GARCH, особливо при роботі з нелінійними та складними часовими рядами [2].

Побудована нейронна мережа для нашої задачі прогнозування має архітектуру «Input-LSTM-Dropout-LSTM-Dense» (рис. 1). Шар «Input» приймає послідовність даних як вхід. Шари «LSTM» відповідають за вилучення інформації з вхідної послідовності. Шар «Dropout» запобігає перенавчанню мережі. Останній шар «Dense» є звичайним повнозв'язним шаром, який приймає вихід від попереднього шару і повертає вже остаточний прогноз [3].

Layer (type)	Output Shape	Param #
keras_tensor_18 (InputLayer)	(None, 10, 2)	0
lstm_4 (LSTM)	(None, 10, 50)	10,600
dropout_2 (Dropout)	(None, 10, 50)	0
lstm_5 (LSTM)	(None, 50)	20,200
dense_2 (Dense)	(None, 1)	51

Рисунок 1. Архітектура нейронної мережі

3. ЕНТРОПІЯ ФІНАНСОВОГО РИНКУ

Ентропія є мірою невизначеності випадкових величин. У контексті фінансових ринків, ентропія дозволяє кількісно оцінити рівень невизначеності в часових рядах цін на акції. Чим вища ентропія, тим більш хаотичними є зміни цін, що може ускладнювати прогнозування. Натомість, низька ентропія свідчить про більшу передбачуваність, що сприяє більш точним прогнозам [4]. Тож, ентропію було інтегровано до моделей прогнозування задля покращення їх здатності враховувати ринкові коливання.

В даному дослідженні для обчислення ентропії використовується ентропія Шеннона. Це класичний підхід до її вимірювання, який визначається наступною формулою [4]:

$$H(X) = -\sum_{i=1}^{N} p(x_i) \log_b p(x_i),$$

де $p(x_i)$ – ймовірність значення x_i, N – кількість можливих значень, b – значення варіюється між 2, e та 10 залежно від бажаної одиниці виміру. В дослідженні використовується $b = e$.

4. РЕАЛІЗАЦІЯ СППР

В рамках дослідження було спроектовано та чітко визначено архітектуру цільової СППР. Структурна схема архітектури СППР наведена на рисунку 2.

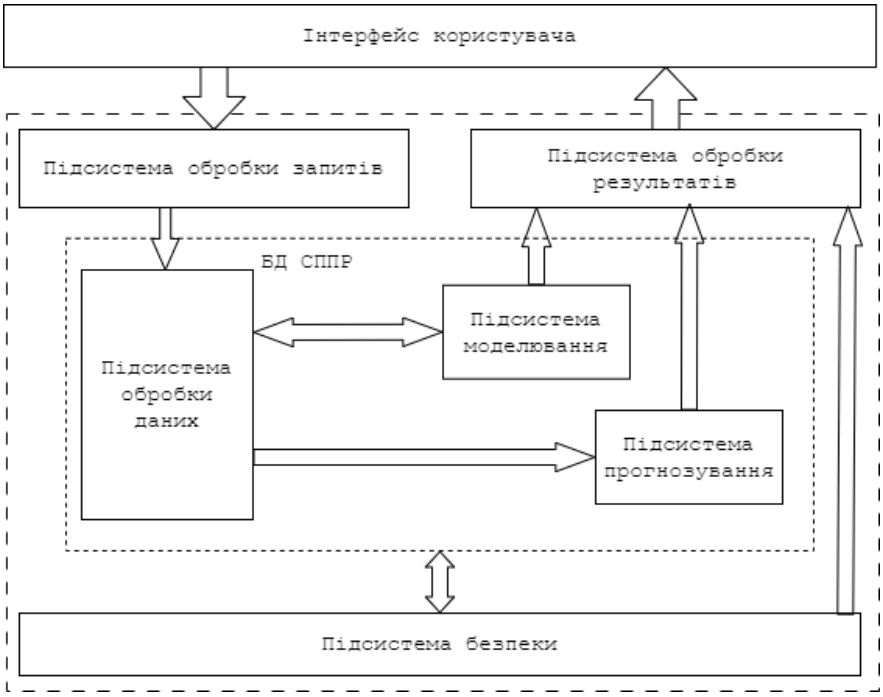


Рисунок 2. Структурна схема архітектури СППР

Система має 8 модулів, кожен з яких відповідає за виконання певних процесів.

1. Інтерфейс користувача. Підсистема, яка забезпечує взаємодію користувача з СППР на всіх етапах роботи. Має забезпечувати шляхи отримання даних та всіх можливих запитів, відображати вікна для виведення інформації користувачеві.
2. Модуль обробки запитів. Обробляє запити, отримані за допомогою інтерфейсу користувача. Ініціює роботу пов’язаних з запитом підсистем.
3. Модуль обробки даних. Залежачи від запиту, модуль або завантажує дані з сервісу Yahoo Finance та трансформує їх у вигляді часових рядів, зберігаючи результати в базі даних (БД) системи, або завантажує дані з бази даних СППР. Відповідає за попередню обробку даних та обчислення значень ентропії.
4. Підсистема моделювання. Отримує оброблені дані з БД. Модуль відповідає за побудову LSTM моделей та їх тренування на вхідних даних. По завершенню тренування найкращі моделі зберігаються до БД системи для їх подальшого використання.
5. Підсистема прогнозування. Завантажує моделі для відповідних даних з БД та проводить прогнозування на визначену користувачем кількість днів.
6. Підсистема безпеки. Збирає дані про роботу всіх процесів системи. Обробляє можливі помилки, що виникають під час виконання процесів.
7. Підсистема обробки результатів. Інтерпретує вихідну з підсистем інформацію та виводить її на інтерфейс. Агрегує результати прогнозування часових рядів для демонстрації зміни динаміки цін на рівні фінансової індустрії або сектору.
8. База даних. Зберігає вхідні дані та навчені моделі, оновлює та надає ці дані на вимогу інших підсистем.

Проектування інтерфейсу користувача було виконано з дотриманням вимог адаптованості, достатності, дружності та гнучкості [5].

5. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

5.1. Оцінка якості прогнозів моделей

Було проведено експерименти із застосуванням побудованих моделей до реальних фінансових даних. В якості вихідних даних було використано часові ряди змін цін закриття акцій компаній Amazon.com, Inc. (AMZN), Mondelez International, Inc. (MDLZ), NIKE, Inc. (NKE), The New York Times Company (NYT), NVIDIA Corporation (NVDA). Для компаній прогнозували ціни на їх акції на 10 днів вперед з урахуванням фактору ентропії. В якості метрики якості виступає середньоквадратична помилка (RMSE), середня абсолютна похибка (MAE), середня абсолютна відсоткова похибка (MAPE) [6]. Результати експериментів наведено у таблиці 1.

Таблиця 1. Результати експериментів прогнозування

Компанія	RMSE	MAE	MAPE	Час
AMZN	2.79077	2.46208	0.01311	0:00:09.985
MDLZ	1.18925	0.95694	0.01348	0:00:09.997
NKE	1.13506	0.95184	0.01152	0:00:09.565
NYT	1.33996	1.27757	0.02311	0:00:10.175
NVDA	3.88073	3.06548	0.02249	0:00:10.783

5.2. Приклад роботи СППР

На головному екрані спроектованої системи користувач має три типи роботи з системою: Сектор, Індустрія та Акції. Відповідно, для типу роботи «Сектор» результатом роботи СППР буде прогнозування зміни тенденції фінансового сектору. Для типу «Індустрія» — прогнозування зміни тенденції індустрії відповідного фінансового сектору. Для типу «Акції» — прогнозування зміни цін акції компанії. Результати процесів роботи СППР для кожного типу зображено на рисунках 3, 4 та 5 відповідно.

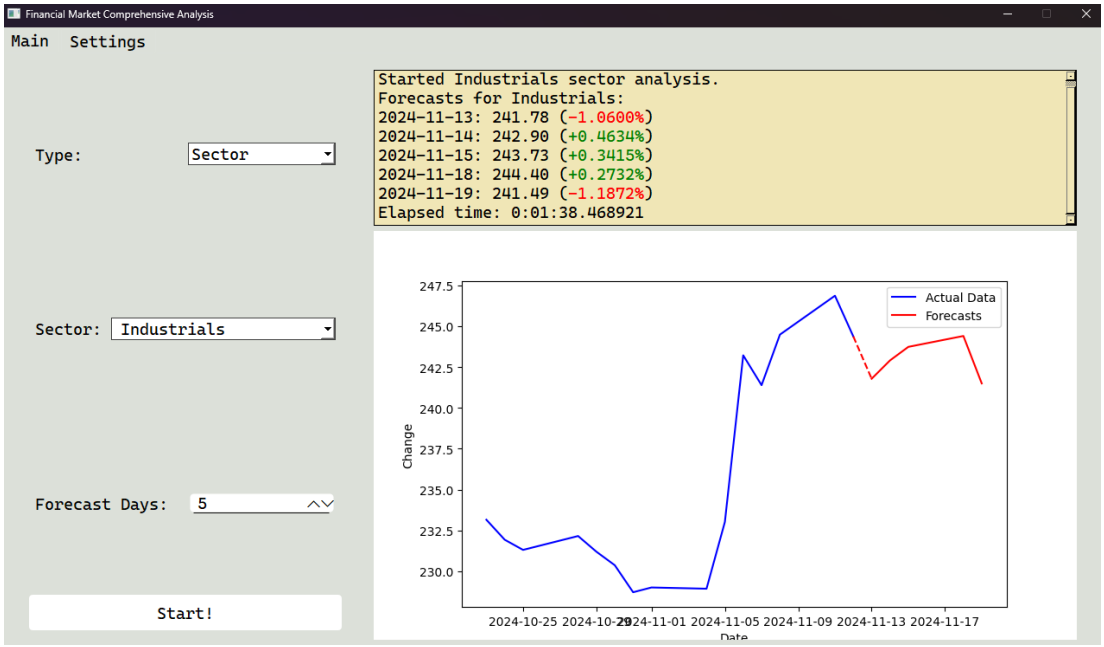


Рисунок 3. Результат процесів для типу роботи «Сектор»

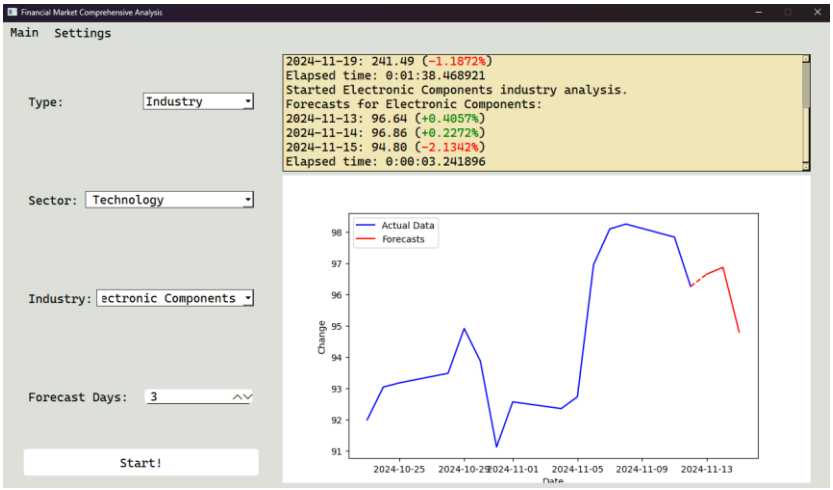


Рисунок 4. Результат процесів для типу роботи «Індустрія»

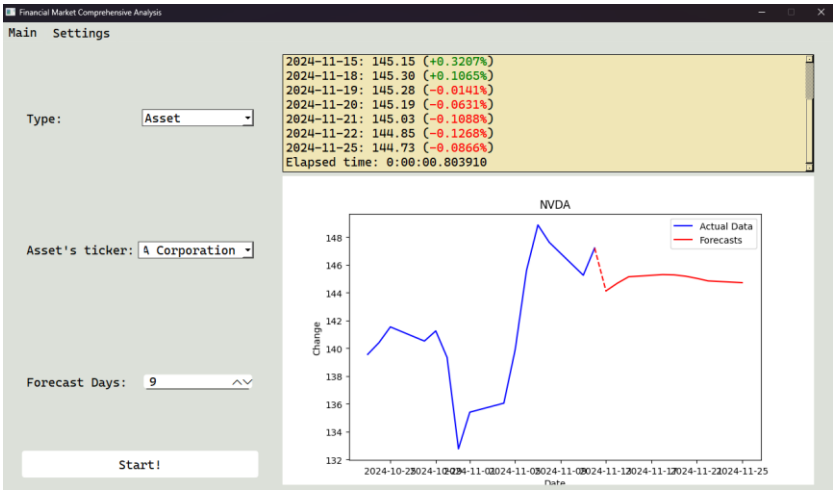


Рисунок 5. Результат процесів для типу роботи «Акції»

Крім основного функціоналу, користувач може оновити дані та моделі в базі даних. Ці операції ініціюються з вкладки «Налаштування». Результат виконання оновлення моделей зображено на рисунку 6.

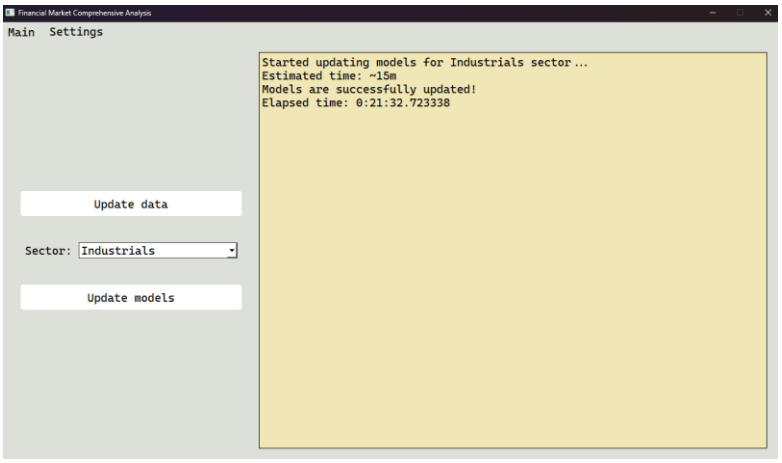


Рисунок 6. Результат процесів оновлення моделей в БД

6. ВИСНОВКИ

На сьогоднішній день фінансові ринки отримують все більшої популярності і зацікавленості з боку населення, вони стають більш доступними для користувачів. Однак складність та нестаціонарність даних, що фігурують на фінансовому ринку, потребують додаткових засобів для їх обробки. Дуже важливо використовувати якісні інструменти для аналізу даних з метою прийняття правильних стратегічних рішень на фінансовому ринку. Саме з цієї точки зору в нашому дослідженні було реалізовано систему підтримки прийняття рішень для комплексного аналізу і моніторингу фінансового ринку.

Було досліджено та інтегровано фактор ентропії в моделі прогнозування з метою врахування оцінки невизначеності для відповідного розподілу цін. На реальних даних цін на акції компаній було побудовано моделі LSTM і оцінено якість її прогнозування. Результати прогнозування на 10 днів цін акцій п’яти компаній з різних фінансових секторів показали, що модель має достатньо точні результати для того, щоб спиратись на її прогнози і використовувати для інвестування.

Реалізовану СППР було протестовано шляхом прогнозування зміни динаміки промислового сектору, індустрії електронних компонентів та цін акцій компанії NVIDIA. Тести показали хорошу результативність побудованої системи.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Zhenglin L., Hanyi Y., Jinxin X., Jihang L., Yuhong M. Stock Market Analysis and Prediction Using LSTM: A Case Study on Technology Stocks. Innovations in Applied Engineering and Technology, 2023, P.1–6.
2. Sonkavde G., Dharrao D.S., Bongale A.M., Deokate S.T., Doreswamy D., Bhat S.K. Forecasting Stock Market Prices Using Machine Learning and Deep Learning Models: A Systematic Review, Performance Analysis and Discussion of Implications. International Journal of Financial Studies, 2023.
3. Keras layers API. URL: https://keras.io/api/layers/ (дата звернення: 12.11.2024).
4. Schwill S. Entropy Analysis of Financial Time Series. The University of Manchester, 2015, P.137.
5. Бідюк П.І., Тимощук О.Л., Коваленко А.Є., Коршевнюк Л.О. Системи і методи підтримки прийняття рішень. Підручник для здобувачів ступеня магістра. Київ: КПІ ім. Ігоря Сікорського, 2022. 610с.
6. Metrics Evaluation: MSE, RMSE, MAE and MAPE. URL: https://medium.com/@jonatasv/metrics-evaluation-mse-rmse-mae-and-mape-317cab85a26b (дата звернення: 12.11.2024)

СИСТЕМНИЙ ПІДХІД ДО КОМПЛЕКСНОГО УПРАВЛІННЯ КЛІЄНТСЬКИМ ПОРТФЕЛЕМ БАНКУ

Гаврилко Д.О.[1], Кузнєцова Н.В.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] dddashagavrylko@gmail.com, [2] n.kuznietsova@kpi.ua

У роботі проводилось дослідження клієнтів банку з метою виявлення схильних до відтоку груп клієнтів. Для прогнозування відтоку клієнтів було побудовано та налаштовано ряд моделей, зокрема, логістичну регресію, наївний байєсівський класифікатор, дерева рішень, а також ансамблеві методи – AdaBoost та XGBoost, а також додатково використано метод опорних векторів (SVM). Для розуміння поведінки клієнтів і їх відтоку було проведено аналіз виживання з використанням непараметричної моделі Каплана-Майєра та напівпараметричної моделі пропорційних ризиків Кокса. Було виконано сегментацію клієнтської бази для виявлення груп клієнтів зі схожими характеристиками. У роботі також було розроблено програмний продукт, що використовує методи машинного навчання та аналізу виживання для комплексного управління клієнтським портфелем банку.

Ключові слова: системний аналіз, інтелектуальний аналіз даних, клієнтський портфель, машинне навчання, моделі виживання.

1. ВСТУП

Управління клієнтським портфелем є ключовим аспектом діяльності банківських установ. В умовах стрімкої діджиталізації та глобалізації фінансового ринку, банківський сектор переживає фундаментальні зміни у підходах до взаємодії з клієнтами. Перехід від традиційної продуктоцентричної моделі до клієнтоорієнтованого підходу вимагає від банків впровадження інноваційних технологій аналізу даних та прийняття рішень. Це особливо актуально в контексті зростаючої конкуренції не лише з традиційними фінансовими установами, але й з фінтех-компаніями.

Сучасні методи інтелектуального аналізу даних, зокрема машинне навчання та аналіз виживання, відкривають принципово нові можливості для банків у сфері персоналізації послуг, оптимізації маркетингових стратегій, управління ризиками відтоку клієнтів тощо. Ці технології дозволяють не лише аналізувати історичні дані, але й прогнозувати майбутню поведінку клієнтів, виявляти приховані закономірності та формувати персоналізовані рекомендації для кожного сегмента клієнтської бази.

2. ЗАДАЧА УПРАВЛІННЯ КЛІЄНТСЬКИМ ПОРТФЕЛЕМ

Забезпечення ефективного управління клієнтським портфелем банку сьогодні є важливою задачею, продиктовано сучасними тенденціями та вектором розвитку банківського сектору. В умовах стрімкої цифровізації, застосування сучасних методів інтелектуального аналізу даних стає критично важливим для збереження конкурентоспроможності банків. Станом на 2024 рік банківська система України демонструє стабільне зростання - активи платоспроможних банків зросли на 6% до 3,13 трлн грн, при цьому кредитний портфель

збільшився на 7% до 1,2 трлн грн. Це створює сприятливі умови для впровадження інноваційних технологій в області управління клієнтським портфелем, а також виникає потреба в кращому розумінні своїх споживачів з метою побудови довготривалої співпраці з ними [1].

Еволюція банківського маркетингу в Україні пройшла значний шлях розвитку: від формування базових понять (1991–2002) через впровадження маркетингових інструментів (2003–2010) та CRM-систем (2011–2019) до сучасного етапу цифрової трансформації, що характеризується розвитком омніканального підходу та використанням інтелектуального аналізу даних для побудови ефективної взаємодії з клієнтами[2].

Спостерігається кардинальна зміна парадигми – перехід від традиційного продуктоорієнтованого підходу до клієнтоорієнтованого. На відміну від традиційного підходу, де інформація була розпорошена між різними системами, сучасні аналітичні платформи забезпечують цілісне уявлення про клієнта та можливість персоналізованого обслуговування. Клієнтський портфель банку тепер розглядається як стратегічний актив, що включає комплексну інформацію щодо клієнтів – від базових даних до поведінкових патернів та фінансових показників[3].

За даними НБУ, вже 79,47% фізичних осіб та 89,15% суб'єктів господарювання користуються послугами дистанційного (інтернет) банкінгу, що свідчить про високий рівень готовності ринку до впровадження передових аналітичних інструментів. Сучасні банки активно розвивають екосистеми та маркетплейси фінансових послуг, пропонуючи клієнтам не лише традиційні банківські продукти, але й широкий спектр супутніх послуг[4].

У контексті розвитку банківського сектору України особливого значення набуває здатність банків ефективно використовувати наявні дані для прийняття обґрунтованих рішень та побудови довгострокових відносин з клієнтами. Машинне навчання стає дієвим інструментом управління клієнтським портфелем, дозволяючи банкам здійснювати сегментацію клієнтської бази, прогнозувати відтік клієнтів, аналізувати життєвий цикл та оцінювати ефективність маркетингових кампаній. Впровадження нових аналітичних інструментів та систем інтелектуального аналізу даних стає не просто конкурентною перевагою, а необхідною умовою для забезпечення стійкого розвитку банків у сучасному цифровому світі[5, 6].

Для комплексного аналізу клієнтського портфеля банку в даному дослідженні були застосовані інструменти машинного навчання. Для прогнозування відтоку клієнтів було побудовано та налаштовано ряд моделей, що зазвичай показують хорошу прогностичну здатність в подібних задачах бінарної класифікації, а саме, використання логістичної регресії дозволяє оцінити ймовірність події відтоку на основі набору характеристик. У проведеному дослідженні було також використано наївний байєсівський класифікатор, що базується на теоремі Байєса та ефективно працює з категоріальними змінними. Дерева рішень обрано як метод, що дозволяє виявляти нелінійні залежності та забезпечує зрозумілу інтерпретованість результатів. Для подальшого підвищення якості прогнозування застосовано і більш складні ансамблеві методи – AdaBoost та XGBoost, які дозволяють поетапно покращувати точність моделі, комбінуючи результати простіших класифікаторів. Додатково використано метод опорних векторів (SVM), який ефективно працює з багатовимірними даними та дозволяє знаходити оптимальну розділяючу гіперплощину для класифікації клієнтів[7].

Для глибшого розуміння поведінки клієнтів проведено аналіз виживання з використанням непараметричної моделі Каплана-Майєра та напівпараметричної моделі пропорційних ризиків Кокса[8]. Ці методи дозволяють оцінити тривалість взаємодії клієнтів з банком та виявити фактори, що впливають на ризик відтоку. На завершальному етапі здійснено сегментацію клієнтської бази з використанням різних підходів до кластеризації

(методу k-середніх та ієрархічного агломеративного кластерного аналізу), що дозволило виявити природні групи клієнтів зі схожими характеристиками. У ході дослідження проведено порівняльний аналіз ефективності всіх використаних методів на реальних банківських даних. Результати роботи представлено далі.

3. ПОБУДОВА МОДЕЛЕЙ ПРОГНОЗУВАННЯ

В якості даних для аналізу був використаний набір анонімізованих даних міжнародного банку, який містить інформацію клієнтів, таку як: їх кредитний рейтинг, географічне розташування, вік, тривалість співпраці з банком, баланс на рахунку, кількість придбаних банківських продуктів, наявність кредитної картки тощо.

Для ефективного прогнозування відтоку клієнтів першим кроком було проведено глибокий розвідувальний аналіз та візуалізацію даних, що вже на даному етапі дозволило виявити важливі закономірності в поведінці клієнтів. Зокрема, було встановлено, що жінки демонструють вищу схильність до припинення співпраці з банком (25.07% проти 16.47% у чоловіків), найвищий рівень відтоку спостерігається у віковій групі 50-60 років (56.2%), а також клієнти з високими балансами частіше припиняють співпрацю з банком.

Важливим викликом при моделюванні стала значна незбалансованість класів у вихідних даних – кількість клієнтів, що залишили банк, становила лише 20.4% від загальної вибірки. Така незбалансованість могла призвести до упередженості моделей у бік мажоритарного класу та їх нездатності ефективно виявляти випадки відтоку. Для вирішення цієї проблеми було застосовано такі техніки балансування даних: SMOTE для збільшення представництва міноритарного класу та RandomUnderSampler для зменшення кількості прикладів мажоритарного класу.

У процесі моделювання було протестовано сім різних алгоритмів машинного навчання: логістична регресія, наївний байєсівський класифікатор, дерева рішень, AdaBoostClassifier, XGBClassifier, GaussianNB та метод опорних векторів. Для кожного алгоритму спочатку оцінювалась базова прогностична здатність на необроблених даних, після чого проводилось навчання на попередньо підготовлених та збалансованих наборах даних. Додатково для кожної моделі здійснювався підбір оптимальних гіперпараметрів з використанням методу GridSearchCV, що дозволяє значно покращити їх ефективність.

Для оцінки якості моделей використовувався комплекс метрик: Accuracy, Precision, Recall та F1-score. Додатково для кожної моделі було побудовано ROC-криву та матрицю помилок, що дозволило детально оцінити їх класифікаційні можливості. Результати оцінювання якості моделей наведено в таблиці 1 та на рисунку 1.

Таблиця 1. Результати прогнозування відтоку на тестовому наборі даних

Модель	Accuracy	Precision	Recall	F1 score
AdaBoostClassifier	0.81304	0.8851	0.8744	0.8758
Decision trees	0.7301	0.7671	0.7731	0.7701
GaussianNB	0.7300	0.8782	0.7591	0.8228
Logistic Regression	0.7669	0.8565	0.8395	0.8479
XGBClassifier	0.8126	0.9069	0.8490	0.8770
SVC	0.7636	0.8323	0.8524	0.8423
Naïve baes	0.7303	0.4004	0.7156	0.5167

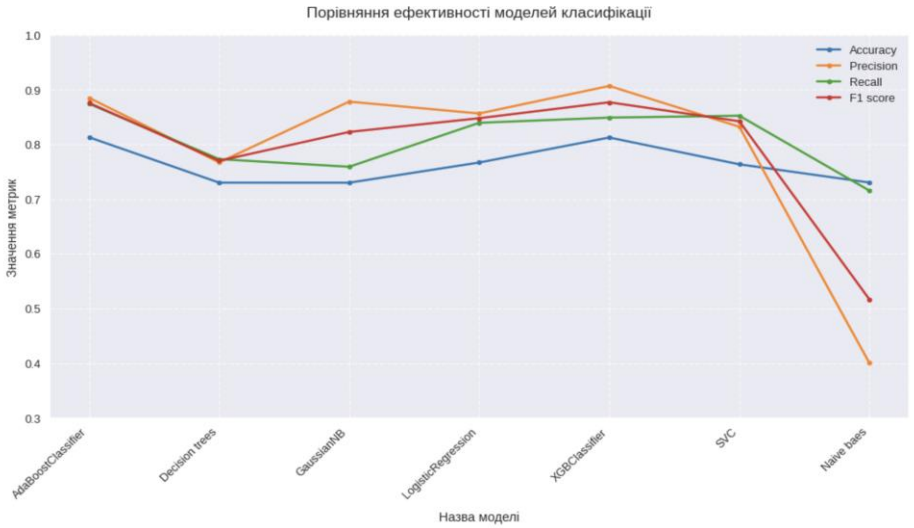


Рисунок 1. Порівняння ефективності моделей класифікації

За результатами порівняльного аналізу найвищу ефективність в задачі прогнозування відтоку клієнтів банку продемонстрували моделі AdaBoostClassifier та XGBClassifier з точністю 81.3% та 81.26% відповідно та значеннями F1-score на рівні 0.87–0.88. Логістична регресія та метод опорних векторів показали дещо нижчі, але прийнятні результати з точністю близько 76–77%. Наївний байєсівський класифікатор та дерева рішень виявилися менш ефективними в умовах даної задачі.

Для найбільш успішної моделі – AdaBoostClassifier – було також проведено аналіз важливості ознак, що дозволило виявити характеристики, які найбільше впливають на ймовірність відтоку клієнтів. Ця інформація може бути особливо корисною для розробки таргетованих стратегій утримання клієнтів.

Для глибшого розуміння процесу відтоку клієнтів було проведено аналіз виживання з використанням методу Каплана-Майєра, який є важливим інструментом в оцінці ризиків та аналізі часових характеристик відтоку, який дозволив визначити критичні часові періоди, коли ризик втрати клієнта є найвищим. Це особливо актуально для оптимізації ресурсів банку та проведення своєчасних превентивних заходів утримання клієнтів.

Спочатку було проведено аналіз та оцінку часу утримання середньостатистичного клієнта банку (рис. 2), що дозволило зрозуміти, як довго в середньому клієнти залишаються з банком перед тим, як приймають рішення про відмову від користування його послугами. Для більшої точності та деталізації оцінок було також проведено дослідження в межах різних клієнтських груп (рис. 3–6), сформованих за такими характеристиками як рівень лояльності, категорія цінності клієнта та рівень ризику.

Комплексний аналіз виживання було проведено з використанням розширеного набору характеристик, який включав як базові параметри (тривалість обслуговування, тип картки, географія, статус активності), так і створені показники (рівень фінансової активності, категорія цінності клієнта, рівень лояльності та ризику, інтенсивність використання продуктів).

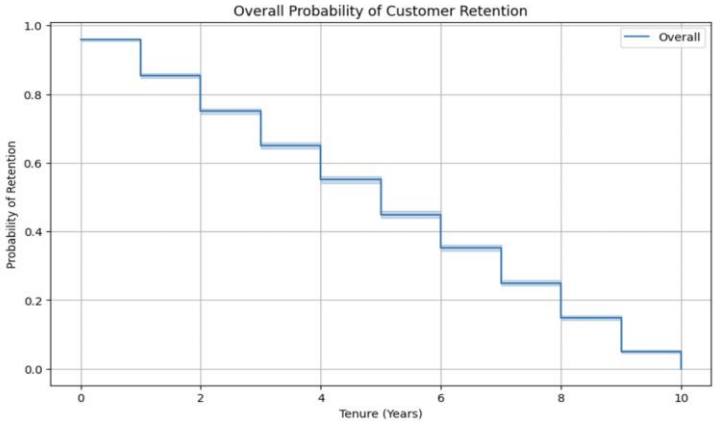


Рисунок 2. Ймовірність утримання клієнта в різні часові періоди

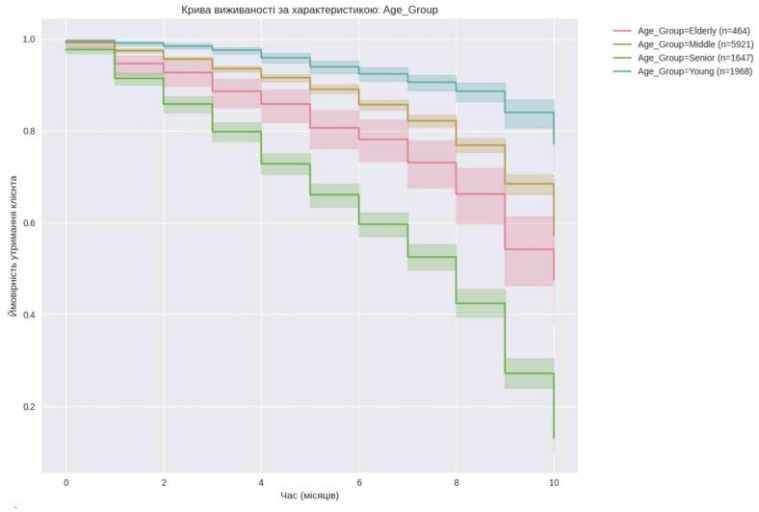


Рисунок 3. Ймовірність утримання в різні часові періоди за віковими групами

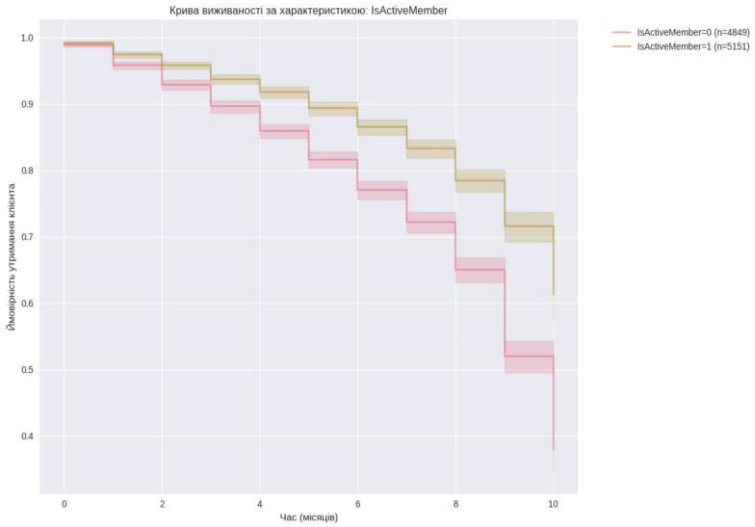


Рисунок 4. Ймовірність утримання в різні часові періоди за групами активності

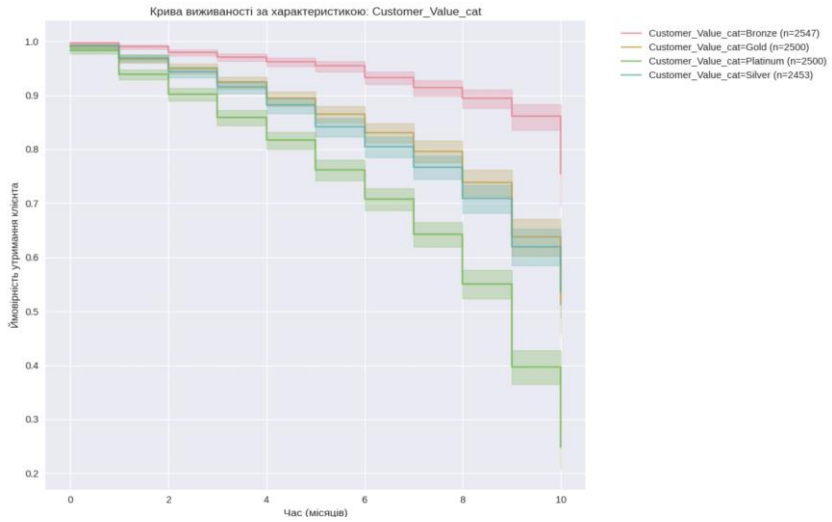


Рисунок 5. Ймовірність утримання в різні часові періоди за групами цінності клієнтів

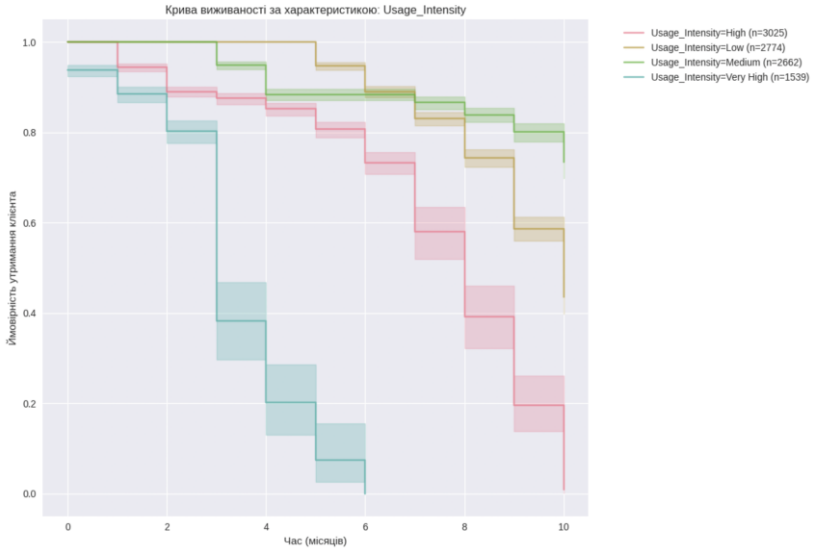


Рисунок 6. Ймовірність утримання в різні часові періоди за групами інтенсивності користування послугами банка

За результатами проведеного аналізу було виявлено, що ймовірність утримання клієнтів має чітку тенденцію до зниження з часом, при цьому найбільш критичними періодами є 2–4 роки та 6–8 років співпраці. Після 4 років обслуговування ймовірність утримання клієнта падає до 55%, а після 8 років – до 25%. Особливо критичним є період після 10 років співпраці, коли ймовірність утримання стає нижчою 10%.

Додатково було проведено аналіз виживання (лояльності клієнта до банку) з використанням моделі пропорційних ризиків Кокса, що дозволяє оцінити відносний вплив різних факторів на ризик відтоку та побудувати прогнозовані криві виживання для різних груп клієнтів. Ці моделі можуть бути використані для розробки диференційованих стратегій утримання клієнтів (рис. 7–10).

	coef	exp(coef)	se(coef)	coef lower 95%	coef upper 95%	exp(coef) lower 95%	exp(coef) upper 95%	cmp to	z	p	-log2(p)
CreditScore	-0.03	0.97	0.02	-0.07	0.00	0.93	1.00	0.00	-1.93	0.05	4.23
Gender	0.27	1.31	0.04	0.20	0.34	1.22	1.41	0.00	7.41	<0.005	42.82
Age	0.36	1.43	0.02	0.32	0.39	1.38	1.47	0.00	21.69	<0.005	344.09
Balance	0.10	1.10	0.02	0.06	0.14	1.06	1.15	0.00	4.86	<0.005	19.71
NumOfProducts	-0.04	0.96	0.02	-0.08	-0.00	0.93	1.00	0.00	-2.14	0.03	4.97
HasCrCard	-0.04	0.96	0.04	-0.12	0.04	0.89	1.04	0.00	-1.07	0.28	1.81
IsActiveMember	-0.45	0.64	0.04	-0.52	-0.38	0.59	0.69	0.00	-12.11	<0.005	109.69
EstimatedSalary	0.00	1.00	0.02	-0.03	0.04	0.97	1.04	0.00	0.27	0.79	0.34
Satisfaction Score	-0.01	0.99	0.02	-0.04	0.03	0.96	1.03	0.00	-0.51	0.61	0.71
Point Earned	-0.00	1.00	0.02	-0.04	0.03	0.96	1.03	0.00	-0.20	0.84	0.25
Geography_France	-0.14	0.87	0.05	-0.24	-0.05	0.79	0.95	0.00	-3.08	<0.005	8.92
Geography_Germany	0.29	1.34	0.05	0.19	0.39	1.21	1.48	0.00	5.85	<0.005	27.62
Geography_Spain	-0.10	0.90	0.05	-0.20	-0.00	0.82	1.00	0.00	-1.97	0.05	4.37
Card Type_DIAMOND	0.03	1.03	0.05	-0.06	0.13	0.94	1.14	0.00	0.70	0.48	1.06
Card Type_GOLD	-0.05	0.95	0.05	-0.15	0.04	0.86	1.04	0.00	-1.09	0.27	1.86
Card Type_PLATINUM	0.02	1.02	0.05	-0.07	0.12	0.93	1.13	0.00	0.47	0.64	0.65
Card Type_SILVER	-0.00	1.00	0.05	-0.10	0.09	0.90	1.10	0.00	-0.08	0.94	0.09

Рисунок 7. Оцінки впливу факторів на ризик відтоку клієнтів

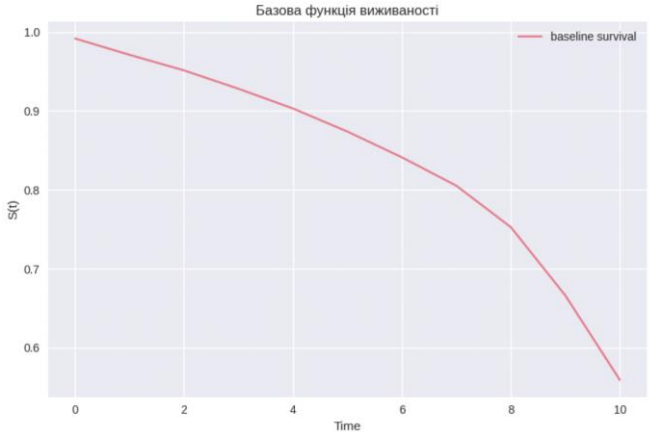


Рисунок 8. Базова функція виживання моделі пропорційних ризиків Кокса

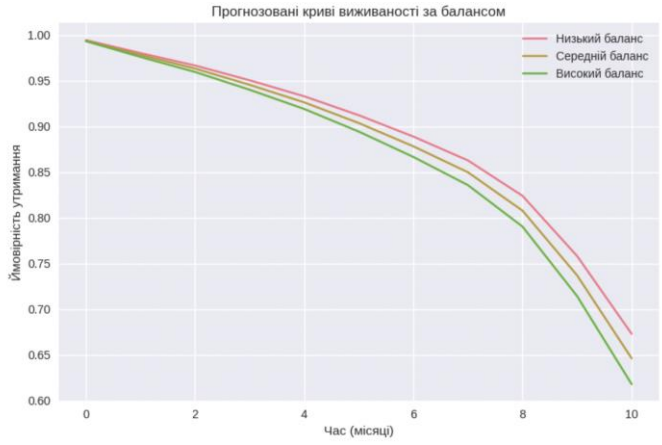


Рисунок 9. Криві виживання (утримання) клієнтів, диференційованих за балансом

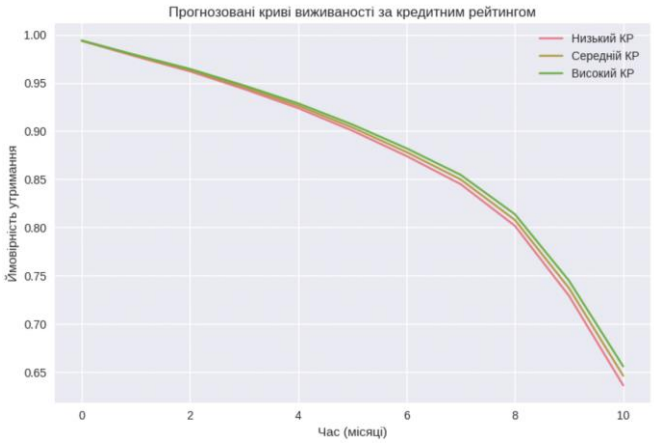


Рисунок 10. Прогнозовані криві виживання, диференційованих за кредитним рейтингом

Заключним етапом дослідження стала сегментація клієнтської бази банку з використанням методів машинного навчання без учителя – K-means та Agglomerative Clustering. За результатами методу ліктя та оцінки силуету було визначено оптимальну кількість кластерів k=3, що також підтвердилося аналізом дендрограми ієрархічної кластеризації (рис. 11).

Візуалізація результатів у просторі головних компонент (PCA) продемонструвала досить чітке розділення клієнтів на три групи (рис. 12–13).

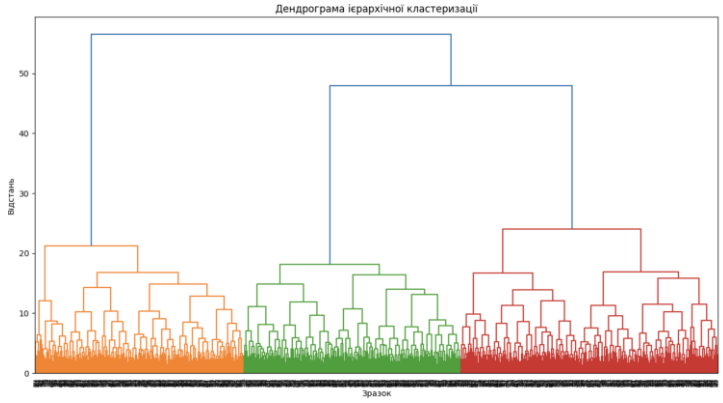


Рисунок 11. Дендрограма ієрархічної кластеризації

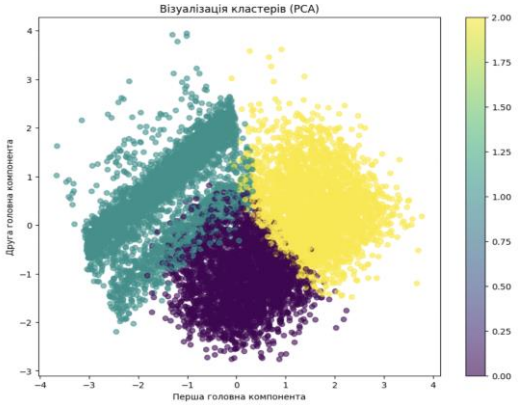


Рисунок 12. Результат клстеризації Kmeans

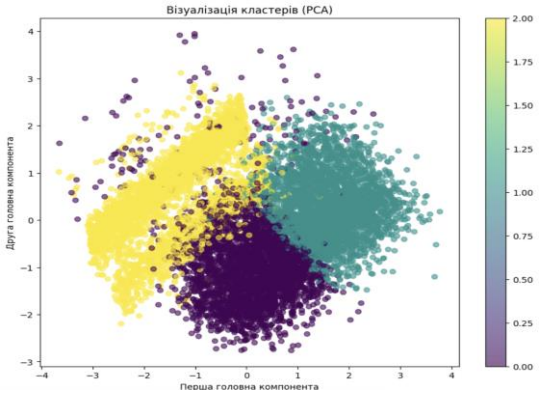


Рисунок 13. Результат кластеризаіції Agglomerative Clustering

Аналіз кластерів продемонстровано на рисунку 14.

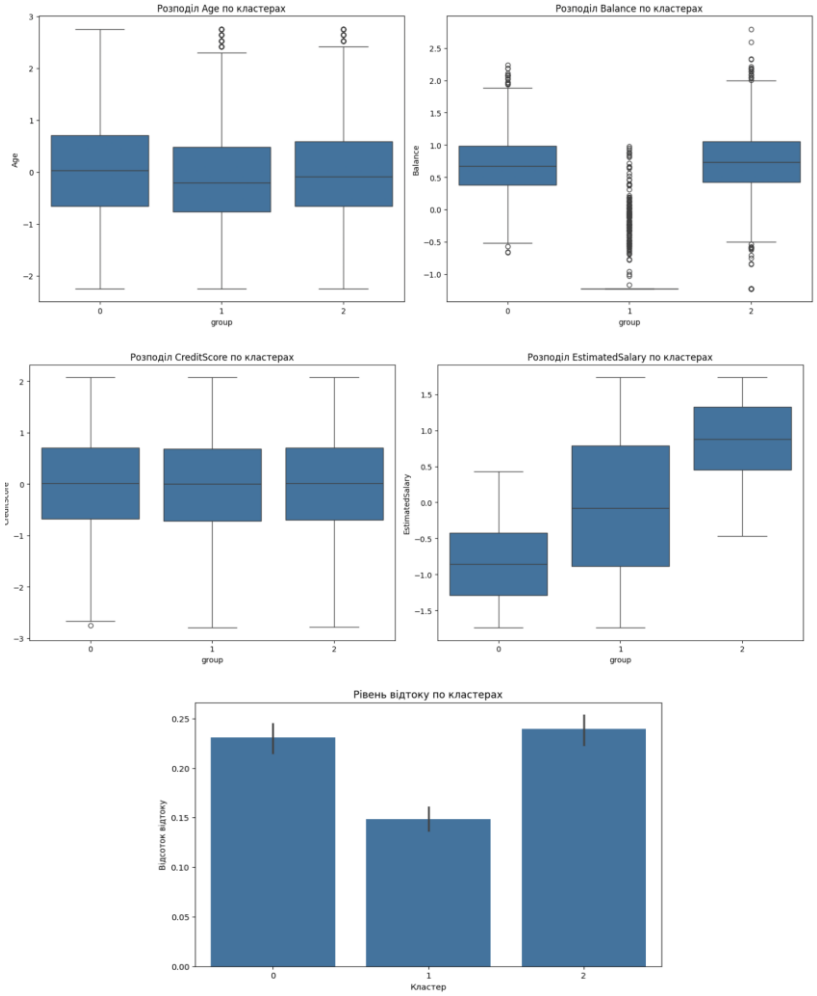


Рисунок 14. Аналіз кластерів

4. ВИСНОВКИ

У сучасному світі комплексне управління клієнтським портфелем банку стає все складнішим через зростаючий обсяг даних, різноманітність банківських продуктів та мінливість споживчої поведінки, що породжує потребу в розробці більш ефективних

інструментів для аналізу та прогнозування клієнтської активності. Особливої актуальності набуває системний підхід до аналізу даних, який розглядає клієнтську базу як цілісну систему взаємопов'язаних елементів та дозволяє виявляти приховані закономірності в поведінці клієнтів.

У ході дослідження було розроблено інтелектуальну систему аналізу даних, яка забезпечує можливість комплексного управління клієнтським портфелем банку через інтеграцію різних методів машинного навчання для прогнозування відтоку клієнтів, проведення ефективної сегментації та оцінки тривалості їх життєвого циклу моделями аналізу виживання. Результати дослідження показали, що найбільш ефективним підходом для прогнозування відтоку клієнтів є використання ансамблевих методів – AdaBoostClassifier та XGBClassifier, які продемонстрували точність понад 81%. При цьому застосування моделі Каплана-Майєра дозволило виявити критичні періоди у взаємодії з клієнтами та розробити стратегії для їх утримання. Доповнення системи методами кластерного аналізу забезпечило основу для розробки персоналізованих підходів до обслуговування різних сегментів клієнтів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Національний банк України, 2024. Огляд банківського сектору. [онлайн] Доступно: https://bank.gov.ua/admin_uploads/article/Banking_Sector_Review_2024-08.pdf?v=8 [Дата звернення 19 жовтня 2024]
2. Buttle, F., 2009. Customer Portfolio Management. In: Customer Relationship Management, Chapter 5. [online] Available at: https://nscpolteksby.ac.id/ebook/files/Ebook/Business%20Administration/Customer%20Relationship%20Management-Francis%20Buttle%20(2009)/11st%20Chapter%205%20-%20Customer%20Portfolio%20management.pdf [Accessed 5 November 2024]
3. Дзюблюк, О.В., 2023. Клієнтоорієнтований підхід у діяльності банків. Монографія / О. В. Дзюблюк. - Тернопіль: ТНЕУ.
4. Національний банк України, 2023. Кількість клієнтів та рахунків в банках – 2023 рік. [онлайн] Доступно: https://bank.gov.ua/ua/news/all/kilkist-kliyentiv-ta-rahunkiv-v-bankah-2023-rik [Дата звернення 12 листопада 2024]
5. Lara-Rubio, J., Villarejo-Ramos, Á.F. and Liébana-Cabanillas, F., 2021. Explanatory and predictive model of customer loyalty using machine learning techniques. European Research on Management and Business Economics, 27(1), pp.100140.
6. Koutanaei, F.N., Sajedi, H. and Khanbabaei, M., 2020. A hybrid data mining model of feature selection algorithms and ensemble learning classifiers for credit scoring. Journal of Retailing and Consumer Services, 27, pp.11-23.
7. Бідюк, П.І., Кузнєцова, Н.В. та Терентьєв, О.М., 2011. Система підтримки прийняття рішень для аналізу фінансових даних. Наукові вісті НТУУ "КПІ", 1, с.48-61.
8. Кузнєцова, Н.В., 2019. Методи і моделі аналізу, оцінювання та прогнозування ризиків у фінансових системах. Дис. д-ра техн. наук. Національний технічний університет України "Київський політехнічний інститут".

ПІДХІД ЩОДО МОДЕЛЮВАННЯ ВАРТОСТІ АКЦІЙ ПО ЗАКРИТТЮ БІРЖОВОГО РИНКУ ІЗ ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ

Гирила С.І.[1] , Савастьянов В.В.

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] hyryla.stepan@lll.kpi.ua

Прогнозування вартості акцій є важливим аспектом аналізу фінансових ринків, що дозволяє інвесторам та аналітикам ринків приймати більш обґрунтовані рішення. Використання нейронних мереж відкриває багато нових можливостей для створення моделей, що здатні враховувати складні нелінійні зв'язки у фінансових даних. Метою роботи є порівняння різних моделей нейронних мереж для моделювання вартості акцій по закриттю біржового ринку. Результатом роботи є можливість порівняти результати різних архітектур, таких як багатошарова нейронна мережа, мережа довгої короткочасної пам'яті та випадковий ліс, для вибору оптимальної моделі прогнозування. Основу в даному дослідженні становлять теоретичні та емпіричні методи дослідження.

Ключові слова: прогнозування вартості акцій, нейронна мережа, LSTM, Random Forest, багатошарова нейронна мережа, фондовий ринок.

1. ВСТУП

Сьогодні фондовий ринок є невід'ємним інструментом для фінансових аналітиків, інвесторів та трейдерів, що дозволяє інвестувати в цінні папери та оцінювати вартість активів. Проте, для більш ефективного прийняття рішень важливо мати точні прогнози цін на акції, що створює необхідність для появи та використання сучасних методів і моделей для аналізу динаміки цін на фондовому ринку. Розробка таких моделей дозволяє прогнозувати зміни у вартості акцій, що дозволяє приймати обґрунтовані рішення щодо інвестицій.

Вибір оптимальної моделі передбачає порівняння різних методів прогнозування для визначення найбільш відповідного до специфіки даних. У даній роботі застосовано підхід на основі використання багатошарової нейронної мережі (MLP), рекурентної нейронної мережі LSTM та алгоритму Random Forest. Моделі було обрано з огляду на їхню здатність обробляти як нелінійні зв'язки, так і часові залежності у нестаціонарних даних. Кожна з цих моделей має свої особливості, що дозволяє всебічно аналізувати ринкові дані та підвищувати точність прогнозування.

Для порівняння якості прогнозів було проведено аналіз продуктивності кожної моделі на історичних даних фондового ринку. Отримані результати демонструють ефективність нейронних мереж та алгоритмів машинного навчання для завдань моделювання і прогнозування в умовах фондового ринку.

2. МЕТОДИ ПРОГНОЗУВАННЯ

Багатошарова нейронна мережа (MLP) - це алгоритм навчання під контролем, який вивчає функцію $f: R^m \rightarrow R^o$ шляхом навчання на наборі даних, де m кількість вимірів на вході та o кількість вимірів на виході. Маючи набір ознак $X = x_1, x_2, \ldots, x_m$ та цільову функцію y мережа може навчитися апроксимувати нелінійну функцію для класифікації або регресії. Даний алгоритм відрізняється від логістичної регресії тим, що між вхідним і вихідним шарами може бути один або більше нелінійних шарів, які називаються прихованими шарами. На рисунку 1 показано MLP з одним прихованим шаром і скалярним виходом.

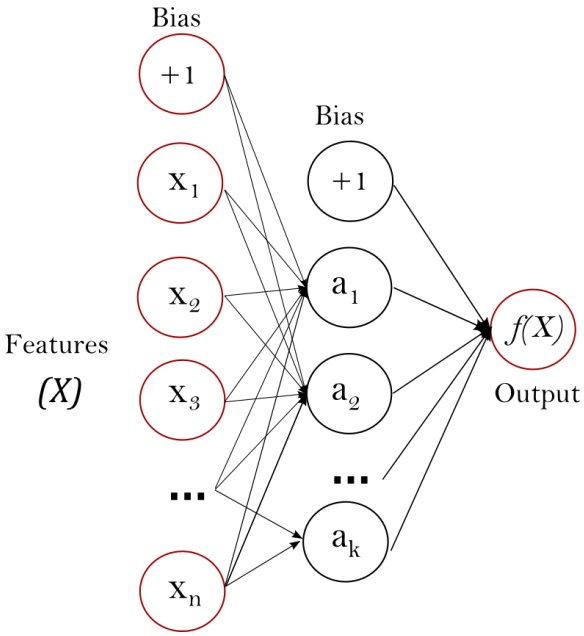


Рисунок 1. MLP з одним прихованим шаром

Крайній лівий шар, відомий як вхідний, складається з набору нейронів $\{x_i | x_1, x_2, \ldots, x_m\}$ що представляють вхідні характеристики. Кожен нейрон у прихованому шарі перетворює значення з попереднього шару за допомогою зваженого лінійного підсумовування $w_1 x_1 + w_2 x_2 + \ldots + w_m x_m$, за яким слідує нелінійна функція активації $g(\cdot): R \rightarrow R$- наприклад, гіперболічна функція тангенса. Вихідний шар отримує значення з останнього прихованого шару і перетворює їх у вихідні значення [1].

Мережа довгої короткочасної пам'яті – це покращена версія рекурентної нейронної мережі (RNN), розроблена Hochreiter & Schmidhuber.

Традиційна RNN має єдиний прихований стан, який передається в часі, що може ускладнити вивчення мережею довгострокових залежностей. LSTM-моделі вирішують цю проблему шляхом введення комірки пам'яті, яка є контейнером, що може зберігати інформацію протягом тривалого періоду.

Архітектура LSTM здатна вивчати довгострокові залежності в послідовних даних, що робить даний тип мереж корисними для таких завдань, як переклад мов, розпізнавання мови та прогнозування часових рядів.

Архітектура LSTM має ланцюгову структуру, яка містить чотири нейронні мережі та різні блоки пам'яті, які називаються комірками. Розглянемо на рисунку 2 спрощену діаграму (ваги та зсуви не відображені), щоб зрозуміти, як рекурентний блок LSTM обробляє інформацію [2].

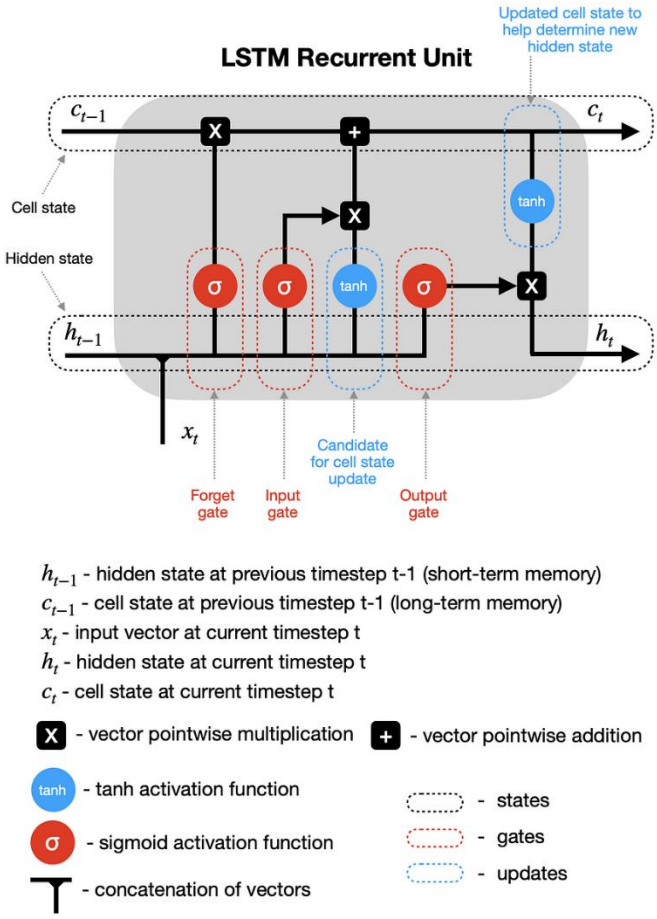


Рисунок 2. Архітектура мережі довгої короткочасної пам’яті (LSTM) [3]

Інформація зберігається в комірках, а маніпуляції з пам'яттю здійснюються за допомогою воріт. Існує троє воріт:

Ворота забуття (Forget gate)

Інформація, яка більше не є корисною у стані комірки, видаляється за допомогою воріт забуття. Два входи x_t (вхід в певний момент часу) та h_{t-1} (вихід попередньої комірки) подаються на ворота і перемножуються з ваговими матрицями з подальшим додаванням зсуву. Результат проходить через функцію активації, яка дає двійковий вихід. Якщо для певного стану комірки вихід дорівнює 0, то інформація забувається, а якщо вихід 1, то інформація зберігається для подальшого використання. Рівняння для воріт забуття має вигляд:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right),$$

де

- W_f – вагова матриця, пов'язана зі входом забування.
- $[h_{t-1}, x_t]$ позначає конкатенацію поточного входу та попереднього прихованого стану.
- b_f – зсув з входом забування.
- σ – сигмоїдна функція активації.

Вхідні ворота (Input gate)

Додавання корисної інформації до стану комірки здійснюється вхідними воротами. Спочатку інформація регулюється за допомогою сигмоїдної функції та фільтрує значення, які потрібно запам'ятати, подібно до воріт забуття, використовуючи входи h_{t-1}та x_t. Потім, за допомогою функції tanh створюється вектор, який має діапазон від –1 до +1, що містить всі

можливі значення з h_{t-1} і x_t Наприкінці, значення вектора і регульовані значення перемножуються для отримання корисної інформації. Рівняння для вхідних воріт має такий вигляд:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\hat{C}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

Ми множимо попередній стан на f_t, не враховуючи інформацію, яку ми раніше вирішили ігнорувати. Далі ми додаємо до нього $i_t * C_t$. Це оновлені значення кандидатів, скориговані на величину, на яку ми вирішили оновити значення кожного стану.

$$C_t = f_t \odot C_{t-1} + i_t \odot \hat{C}_t,$$

де

- $\odot$ позначає поелементне множення;
- tanh – функція активації tanh.

Ворота виводу(Output gate)

Задача вилучення корисної інформації з поточного стану комірки для представлення на виході виконується воротами виводу. Спочатку генерується вектор шляхом застосування функції tanh до комірки. Потім інформація регулюється за допомогою сигмоїдної функції та фільтрується за значеннями, які потрібно запам'ятати за допомогою входів h_{t-1} та x_t. Наостанок, значення вектора і регульованих значень перемножуються, щоб відправити їх як вихід і вхід до наступної комірки. Рівняння для воріт виводу має вигляд:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o).$$

Класифікація випадкових лісів (Random forest classification) – це техніка ансамблевого навчання, призначена для підвищення точності та робастності задач класифікації. Алгоритм будує безліч дерев рішень під час навчання і виводить клас, який є режимом класів класифікації (рис. 3).

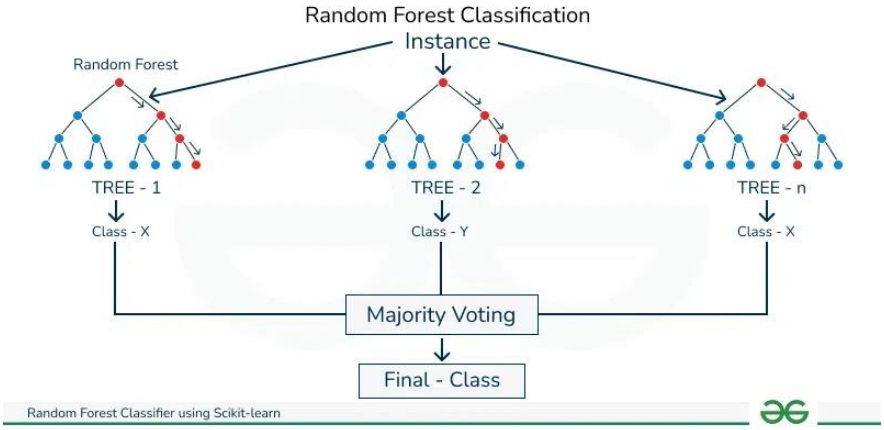


Рисунок 3. Архітектура Random Forest Classification

Кожне дерево рішень у випадковому лісі будується з використанням підмножини навчальних даних і випадкової підмножини ознак, що вносить різноманітність між деревами, роблячи модель більш надійною і менш схильною до перенавчання.

Алгоритм випадкового лісу використовує техніку, яка називається пакування (Bootstrap Aggregating) для створення цих різноманітних підмножин.

На етапі навчання кожне дерево будується шляхом рекурсивного розбиття даних на основі ознак. При кожному розбитті алгоритм вибирає найкращу ознаку з випадкової підмножини, оптимізуючи її за інформативністю або домішками Джині. Процес продовжується до тих пір, поки не буде досягнутий заздалегідь визначений критерій зупинки, наприклад, досягнення максимальної глибини або мінімальної кількості зразків у кожному листовому вузлі.

Після того, як випадковий ліс навчений, він може робити прогнози, використовуючи «голоси» кожного дерева за клас, і клас з найбільшою кількістю голосів стає передбачуваним класом для вхідних даних [4].

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Порівняльний аналіз моделей прогнозування буде здійснюватися на прикладі прогнозування вартості акцій компанії Alphabet Inc. (GOOG).

Першим кроком у прогнозуванні ціни акцій є обробка вхідних даних. Для підвищення якості навчання використовується кластеризація для сегментації даних на групи з подібними характеристиками волативності. Спочатку визначається оптимальна кількість кластерів, для цього використовуються два підходи:

- Метод ліктя (Elbow): в даному методі обчислюється SSE (сума квадратів відстаней від кожної точки до центроїди її кластера). Залежність між кількістю кластерів і SSE відображає вигин на графіку (точку «ліктя»), який вказує на оптимальну кількість кластерів.
- Silhouette Score: в даному методі оцінюється якість кластеризації шляхом порівняння відстаней між точками одного кластеру та іншими кластерами. Чим ближче значення до 1, тим якісніша кластеризація.

Результати обчислень аналізуються програмою та обирається кількість кластерів з найкращими результатами. Графічно результат обчислень можна побачити на рисунку 4.

Рисунок 4. Графіки підходів для визначення кількості кластерів

Алгоритм K-Means групує всі точки даних у відповідні кластери, що дозволяє сегментувати фінансові дані за характеристиками їхньої волатильності.

Після кластеризації дані кожного кластера розділяються на навчальні та тестові вибірки у співвідношенні 80% для навчання і 20% для тестування. Щоб забезпечити випадковість, перед розділенням дані в межах кожного кластера перемішуються для запобігання зміщення моделі, викликаного нерівномірним розподілом даних.

Протягом навчання зберігаються найкращі ваги моделі, це дозволяє уникнути

погіршення моделі через перенавчання або шум. Після завершення навчання для кожного кластера виконується прогнозування на тестовій вибірці, а результати використовуються для оцінки продуктивності моделі.

Для оцінок якості роботи мереж на кожному кластері використано кілька ключових метрик, таких як True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR), False Negative Rate (FNR), а також Root Mean Squared Error (RMSE).

Розглянемо результати роботи багатошарової нейронної мережі на рисунку 5. Дана мережа має функцію активації ELU, нормалізація виконується з-за допомогою BatchNormalization, має 150 епох та розмір партії 128.

```
Summary of results for all clusters:
Cluster 0:
True positive rate = 0.687
False positive rate = 0.313
True negative rate = 0.818
False negative rate = 0.182
RMSE: 0.4976133515281193

Cluster 1:
True positive rate = 0.857
False positive rate = 0.143
True negative rate = 0.689
False negative rate = 0.311
RMSE: 0.48322100334222434
```

Рисунок 5. Результат роботи багатошарової нейронної мережі.

Згідно отриманих результатів можна дійти висновку, що модель cluster 0 має точність у 81% при передбаченні негативних класів, в той час як модель cluster 1 має точність 85% при передбаченні позитивних класів. Для обох моделей значення RMSE менше 0,5, що свідчить про невисокий рівень помилок.

Далі розглянемо результати роботи мережі довгої короткочасної пам’яті які зображені на рисунку 6.

```
Summary of results for all clusters:
Cluster 0:
True positive rate = 0.73
False positive rate = 0.27
True negative rate = 0.794
False negative rate = 0.206
RMSE: 0.4879500364742666

Cluster 1:
True positive rate = 0.615
False positive rate = 0.385
True negative rate = 0.802
False negative rate = 0.198
RMSE: 0.5331641017462287
```

Рисунок 6. Результат роботи LSTM

Згідно отриманих результатів можна зробити висновок, що модель Cluster 0 демонструє хороші результаті, 73% при передбаченні позитивних результатів, та 79% при передбаченні негативних. RMSE знаходиться також у межах <0.5 що свідчить про невисокий рівень помилок.

Наступним розглянемо результати класифікації випадкових лісів (Random Forest) на рисунку 7.

```
Summary of results for all clusters:
Cluster 0:
True positive rate = 0.701
False positive rate = 0.299
True negative rate = 0.813
False negative rate = 0.187
RMSE: 0.49280538030458115

Cluster 1:
True positive rate = 0.791
False positive rate = 0.209
True negative rate = 0.717
False negative rate = 0.283
RMSE: 0.4987293499153675
```

Рисунок 7. Результат роботи Random Forest

Згідно отриманих результатів RMSE для обох кластерів менше 0,5, а точність передбачень становить більше 70%.

4. ВИСНОВОК

Сьогодні фінансові ринки стають все доступнішими для звичайних користувачів, тому виникає необхідність знайти більш ефективний інструмент аналізу та прогнозування для подальшого інвестування.

У даній роботі було проведено порівняльний аналіз методів прогнозування для нестаціонарних і нелінійних процесів фондових ринків та зроблено огляд моделей прогнозування, досліджено їх результати.

Під час аналізу були оцінені результати трьох методів моделювання: багатошарової нейронної мережі (MLP), мережі довгої короткочасної пам’яті (LSTM) та алгоритму Random Forest. Дані були поділені на два кластери в залежності від волатильності акцій, що дало змогу оцінити ефективність кожного методу в різних умовах.

Згідно отриманих результатів MLP показує найкращі результати в кластері 1, де дані характеризуються нижчою волатильністю. Цей метод забезпечує високий рівень точності у виявленні позитивних класів, що відображається у найвищому показнику True Positive Rate, а також має найменшу середньоквадратичну похибку (RMSE). Це робить MLP найбільш ефективним методом для прогнозування цін закриття в стабільних ринкових умовах.

У свою чергу LSTM, навпаки, є найкращим вибором для кластеру 0, що характеризується високою волатильністю. Ця модель показує найкращі результати за показником True Positive Rate у кластері 0, а також найменшу похибку RMSE. Однак у кластері 1 її ефективність знижується через вищий рівень хибнопозитивних передбачень, що обмежує її універсальність.

Random Forest показує стабільні, але середні результати в обох кластерах. Цей метод є універсальним, однак він поступається нейронним мережам у точності розпізнавання складних закономірностей. Він демонструє збалансовані показники у кластері 0, проте програє

MLP у кластері 1 за точністю та узгодженістю передбачень.

Таким чином, для моделювання вартості акцій у стабільних умовах рекомендується використовувати багатошарову нейронну мережу, тоді як для більш волатильних ринків оптимальним є підхід із використанням LSTM. Random Forest можна застосовувати як базову модель, однак для досягнення найкращих результатів доцільно віддавати перевагу нейронним мережам.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Scikit learn. 1.17. Neural network models (supervised). URL: https://scikit-learn.org/1.5/modules/neural_networks_supervised.html

2. GeeksForGeeks. What is LSTM – Long Short Term Memory? URL: https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/

3. Towards Data Science. LSTM Recurrent Neural Networks — How to Teach a Network to Remember the Past URL: https://towardsdatascience.com/lstm-recurrent-neural-networks-how-to-teach-a-network-to-remember-the-past-55e54c2ff22e

4. GeeksForGeeks. Random Forest Classifier using Scikit-learn, URL: https://www.geeksforgeeks.org/random-forest-classifier-using-scikit-learn/

ПЕРСОНАЛЬНИЙ ТРЕВЕЛ-АСИСТЕНТ НА БАЗІ GenAI З ІНТЕГРАЦІЄЮ API ДЛЯ ТОЧНОГО ПОШУКУ АВІАРЕЙСІВ ТА ПОДІЙ

Гула Т.В.[1], Мілявський Ю.Л.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] tetiana.v.gula@gmail.com

Метою дослідження є розробка персонального тревел-асистента, що базується на технологіях генеративного штучного інтелекту (GenAI) з інтеграцією API для точного пошуку авіарейсів і подій. У роботі досліджуються методи інтеграції великих мовних моделей (LLM), оптимізація викликів API для зниження витрат, розробка інтерфейсу для користувачів та забезпечення захисту від небажаних запитів. Експериментальні результати підтверджують ефективність розробленої системи у контексті покращення користувацького досвіду.

Ключові слова: Generative AI, API інтеграція, GPT, великі мовні моделі, чат-бот, туристичні послуги, оптимізація витрат.

1. ВСТУП

У сучасному світі швидкий доступ до релевантної інформації стає дедалі важливішим. Розробка інтелектуальних систем для автоматизації пошуку авіарейсів, подій та інших туристичних послуг є актуальним завданням. Метою роботи є створення персонального тревел-асистента, що використовує технології Generative AI та API інтеграцію для покращення якості обслуговування користувачів.

2. ПОСТАНОВКА ЗАДАЧІ

Розробити систему, яка:

- Інтегрує великі мовні моделі (LLM) з API, такими як Amadeus Air Content (AQC) і Ama4Dev.
- Забезпечує персоналізовані рекомендації на основі запитів користувачів.
- Оптимізує виклики API та промпти для зниження витрат.
- Забезпечує безпеку від небажаних запитів (джейлбрейків).
- Дотримується поставленої ролі.

3. РОЗРОБКА АНАЛІТИЧНОЇ МОДЕЛІ

Для реалізації тревел-асистента було впроваджено клієнт-серверну архітектуру з інтеграцією великих мовних моделей, таких як GPT-3.5 і GPT-4. Система опрацьовує запити, аналізує їхній контекст та виконує API-запити для отримання актуальних даних. У процесі роботи система підраховує використані токени, визначає складність задачі й автоматично обирає оптимальну модель та API. Оптимізація досягається завдяки сумаризації діалогів, дотриманню встановлених лімітів токенів та адаптивному вибору моделі для кожного типу запитів.

4. МЕТОДИ ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ

Для оцінки ефективності розробленої системи використовувався комплексний підхід, що включав експериментальні, аналітичні та практичні методи. Основними етапами дослідження були:

Аналіз продуктивності мовних моделей:

- Порівняння ефективності GPT-3.5 та GPT-4 для різних типів запитів (простих та складних).
- Вимірювання часу обробки запитів та визначення оптимальної моделі для кожного сценарію.
- Тестування здатності моделей надавати точні відповіді, зокрема для запитів, що потребують інтеграції з API.

Інтеграційні випробування:

- Перевірка злагодженої роботи з API Amadeus для пошуку авіарейсів і Ama4Dev для рекомендацій подій.
- Аналіз точності даних, отриманих від API, та їх відповідність потребам користувачів.
- Застосування механізмів кешування для зменшення повторних запитів.

Оцінка користувацького досвіду:

- Проведення опитувань серед тестувальників для збору відгуків про зручність інтерфейсу та точність рекомендацій.
- Аналіз найпоширеніших проблем та вдосконалення системи на основі зворотного зв’язку.
- Автоматизоване тестування релевантності:
- Використання спеціальних метрик для оцінки точності відповідей, уникнення «галюцинацій» моделей.
- Перевірка стійкості системи до небажаних запитів (наприклад, джейлбрейків) та аналіз захисних механізмів.

5. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

У цьому розділі наведено результати дослідження, отримані під час тестування персонального тревел-асистента. Метою було визначити ефективність моделей GPT-3.5 і GPT-4 для обробки різних типів запитів та їх стійкість до джейлбрейків, а також оптимізувати витрати на використання API (табл. 1).

Таблиця 1. Порівняння продуктивності GPT-3.5 і GPT-4

Тип тексту	GPT-3.5	GPT-4	Примітки
Час обробки простих запитів	0.75 с/запит	1.2 с/запит	GPT-3.5 швидший для базових завдань
Точність простих запитів	88%	96%	GPT-4 забезпечує більш релевантні відповіді
Час обробки складних запитів	1.8 с/запит	2.4 с/запит	GPT-4 краще обробляє складні сценарії
Точність складних запитів	70%	92%	GPT-3.5 вразливий до помилок
Джейлбрейки (уразливість)	Успіх у 15% випадків	Успіх у 4% випадків	GPT-4 краще захищає від небажаних маніпуляцій

Результати демонструють, що GPT-4 має вищу точність відповідей та стійкість до джейлбрейків, проте він трохи повільніший і дорожчий у використанні. Для простих завдань GPT-3.5 залишається прийнятним вибором, але для складних сценаріїв GPT-4 є незамінним.

Таблиця 2. Порівняння витрат для різних підходів

Метод	Середня вартість (євро)	Час відповіді (сек)
GPT-4 без узагальнення	2.256	13.57
GPT-4 з узагальненням	1.913	14.27
Гібридний підхід	2.175	16.63

Витрати для GPT-4 значно перевищують показники GPT-3.5 через ціну обробки токенів. Впровадження гібридного підходу дозволило знизити загальні витрати, використовуючи GPT-3.5 для узагальнення діалогів, а GPT-4 – для виконання складних запитів (табл. 2).

Пояснення даних таблиць

1. **Час та точність обробки:** Показано, що GPT-4 обробляє складні запити із точністю 92%, що на 22% вище за GPT-3.5. Однак для простих запитів GPT-3.5 обробляє дані швидше.
2. **Джейлбрейки:** GPT-4 демонструє значно кращу стійкість до небажаних атак завдяки вдосконаленим механізмам безпеки.
3. **Оптимізація витрат:** Узагальнення при використанні GPT-3.5 дозволило скоротити витрати, особливо при довгих діалогах.

Ці дані дозволяють рекомендувати GPT-4 для основного сценарію використання чат-бота, у той час як GPT-3.5 оптимальний для допоміжних функцій.

6. ВИСНОВКИ

У ході виконання роботи було розроблено персональний тревел-асистент, який інтегрує сучасні великі мовні моделі (GPT-3.5 та GPT-4) із зовнішніми API, такими як Amadeus Air Content (AQC) та Ama4Dev. Система забезпечує точний пошук авіарейсів, рекомендації подій та надає персоналізовані відповіді користувачам у режимі реального часу.

Основними досягненнями є:

1. **Підвищення точності відповідей:** GPT-4 показав високу релевантність для складних запитів (92% точності), що значно покращує користувацький досвід.
2. **Оптимізація витрат:** Впровадження гібридного підходу дозволило скоротити витрати на 15-20%, використовуючи GPT-3.5 для узагальнення діалогів та менш складних завдань.
3. **Стійкість до небажаних маніпуляцій:** Завдяки використанню GPT-4 вдалося досягти 95% успішного блокування джейлбрейків.
4. **Ефективність у багатокрокових запитах:** Інтеграція механізму узагальнення діалогів дозволила зберегти контекст розмов при зниженні обсягу даних, які обробляються.

Разом із цим, дослідження показало такі перспективи для подальшого розвитку:

- **Інтеграція додаткових функцій:** Наприклад, реалізація функції бронювання квитків з перенаправленням на зовнішні сервіси.
- **Застосування RAG (retrieval-augmented generation):** Це дозволить ефективно використовувати статичні дані для генерації відповідей.
- **Динамічний профіль користувача:** Система зможе враховувати попередні поїздки, мови та вподобання для персоналізації відповідей.

Результати дослідження підтверджують ефективність запропонованих підходів для створення сучасного тревел-асистента, який відповідає вимогам точності, безпеки та оптимізації витрат.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Brown, T., Mann, B., Ryder, N., et al. "Language Models are Few-Shot Learners." *NeurIPS*, 2020.
2. Rajpurkar, P., Zhang, J., et al. "Evaluation Metrics for AI Models: Beyond Accuracy." *AI Review Journal*, 2023.
3. OpenAI Safety Research. "AI Safety Best Practices." https://openai.com/research
4. Microsoft AI Team. "Using Summarization and Hybrid Techniques for Cost Reduction in AI Systems." *Microsoft Research Blog*, 2024.
5. Amadeus IT Group. "Amadeus Air Content Services API Documentation."
6. Amodei, D., Hernandez, D. "AI and Compute: Scaling Trends in Generative Models." *OpenAI Blog*, 2022.

ЗАСТОСУВАННЯ МЕТОДІВ ЯКІСНОГО АНАЛІЗУ ДЛЯ ДОСЛІДЖЕННЯ ТА ПЕРЕДБАЧЕННЯ СТАНУ ВИРОБНИЧИХ ПІДПРИЄМСТВ

Гулкевич Б.Ю.[1], Савченко І.О.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] boryagulkev@gmail.com, [2] savil.ua@gmail.com

Питання прийняття рішень щодо виробничих підприємств не завжди може покладатися тільки на кількісні параметри, або може не мати потрібних для цього даних. Тому у дослідженні було розглянуто використання методів якісного аналізу в даному напрямку і розроблено програмне забезпечення для цієї задачі. Для стану Крюківського вагонобудівного заводу було зібрано експертні оцінки і проведено двоетапний морфологічний аналіз, фінальні результати якого далі було використано у методі перехресного аналізу. Отримане дозволяє розглянути стан підприємства, його найбільш ймовірні конфігурації параметрів і розглянути як від зміни параметрів зміниться конфігурація.

Ключові слова: якісні методи, морфологічний аналіз, перехресний аналіз, аналіз заводу, стан підприємства.

1. ВСТУП

Питання аналізу стану виробничих підприємств завжди залишається актуальним у економічних відносинах та державному управлінні. Звичною справою для цього є використання методів кількісного аналізу, які дозволяють як і визначити стан, так і прогнозувати як він може змінитися. Однак подібний підхід не завжди може працювати, чи не завжди може бути застосованим.

При роботі з підприємством може не вистачати даних для проведення детальнішого аналізу. Мова йде не лише про неповноту, а також про недостатню кількість даних і їхнє викривлення. Подібні випадки є особливо актуальними в Україні в силу економічної та політичної ситуації, а також історичних обставин [1, 2]. Відповідно для проведення аналізу стану підприємства необхідно використати додаткові параметри та характеристики, які не оцінюються просто сухими кількісними даними. Саме тут є зручним застосування методів якісного аналізу [3].

Методи якісного аналізу покладаються на експертну оцінку, що дозволяє розглянути стан підприємства з інших боків, оцінити додаткові параметри та характеристики. Також подібний підхід дозволяє працювати в умовах неповноти даних та невизначеності, розширюючи можливості. Іншою перевагою якісних методів є спонукання менеджерів до розгляду різних сценаріїв і підходів, що сприяє розвитку творчого мислення та впровадженню інновацій. Це дозволяє знаходити нестандартні рішення для досягнення стратегічних цілей і покращення конкурентної позиції компанії.

Вищезгадані переваги дають можливість краще приймати рішення, особливо у ситуаціях які потребують комплексного підходу, як от підтримка промислових галузей державою, прийняття рішень щодо інвестування у компанію, передбачення можливих рішень конкурентів, або розробка стратегій дії.

2. ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

Метою проведення даної роботи є розгляд використання методів якісного аналізу для аналізу та передбачення стану виробничого підприємства. Виконання включає збір оцінки експертів за допомогою розроблених опитувальних форм, проведення аналізу та стану підприємства. Після цього, за допомогою отриманих результатів має бути розглянуто можливі рішення його керівництва, або держави щодо нього, а також зміну ймовірності цих рішень відповідно до прийняття сторонами рішень.

3. ТЕОРЕТИЧНИЙ ОПИС

Для опису стану підприємства дуже зручним підходом буде визначення важливих характеристик, для чого було обрано морфологічний аналіз [4]. Цей метод спочатку потребує індентифікації характеристичних параметрів $F_i, i \in \overline{1,N}$, для кожного з яких потім визначаються можливі альтернативи (множина альтернатив) $a_j^{(i)} \in \overline{1,n_i}$, де n_i – кількість альтернатив у i-го параметра, j – номер альтернативи. Для виконання цього залучаються експерти. Потім з отриманих результатів складається морфологічна таблиця – множина характеристичних параметрів системи чи об'єкта, кожен з яких характеризується через множину альтернатив, ймовірності настання яких оцінюються експертами. Після формування таблиці проводиться оцінка зв'язків між альтернативами. Так, для кожної пари альтернатив $a_{j_1}^{(i_1)}$ та $a_{j_2}^{(i_2)}$ різних параметрів F_{i_1} та F_{i_2} присвоюється оцінка $c_{i_1 j_1 i_2 j_2} \in [-1,1]$.

Кожен можливий стан підприємства буде описуватися конфігурацією морфологічної таблиці $s = \left\{a_{j_1}^{(1)}, a_{j_2}^{(2)}, a_{j_3}^{(3)}, \dots, a_{j_N}^{(N)}\right\}$, яка є набором по одній альтернативі з кожної характеристики. Для отримання ймовірностей настання конфігурацій використовуються наступні формули:

$$C_{\mathrm{j_1 j_2 \dots j_N}} = \prod_{m=1}^{N-1} \prod_{l=m+1}^{N} \left(c_{m j_m, l j_l} + 1\right)$$

$$\mathrm{p}_{\mathrm{j_k}}^{(i_k)} = \frac{\mathrm{p}_{\mathrm{j_k}}^{\prime(i_k)} \sum_{\mathrm{j_1}=1}^{\mathrm{n_1}} \dots \sum_{\mathrm{j_{k-1}}=1}^{\mathrm{n_{k-1}}} \sum_{\mathrm{j_{k+1}}=1}^{\mathrm{n_{k+1}}} \dots \sum_{\mathrm{j_N}=1}^{\mathrm{n_N}} C_{\mathrm{j_1 j_2 \dots j_N}} p_{j_1}^{\prime(i_1)} \dots p_{j_{k-1}}^{\prime(i_N)} p_{j_{k+1}}^{\prime(i_N)} \dots p_{j_N}^{\prime(i_N)}}{\sum_{\mathrm{j_1}=1}^{\mathrm{n_1}} \dots \sum_{\mathrm{j_1}=1}^{\mathrm{n_1}} C_{\mathrm{j_1 j_2 \dots j_N}} p_{j_1}^{\prime(i_1)} \dots p_{j_N}^{\prime(i_N)}},$$

$$P(s) = \frac{C_{\mathrm{j_1 j_2 \dots j_N}} p_{j_1}^{\prime(i_1)} \dots p_{j_N}^{\prime(i_N)}}{\sum_{\mathrm{j_1}=1}^{\mathrm{n_1}} \dots \sum_{\mathrm{j_1}=1}^{\mathrm{n_1}} C_{\mathrm{j_1 j_2 \dots j_N}} p_{j_1}^{\prime(i_1)} \dots p_{j_N}^{\prime(i_N)}}.$$

Отриманий результат надає оцінку настання кожної можливої конфігурації, тобто кожного можливого варіанту стану підприємства, кожного можливого набору альтернатив параметрів. Для знаходження кожної ймовірностей настання альтернатив $a_j^{(i)}$ треба знайти суму ймовірностей конфігурацій, які включають дану альтернативу. Отриманий результат можна записати у складену морфологічну таблицю.

Визначення можливих рішень держави та керівництва підприємства можливе за допомогою проведення другого етапу морфологічного аналізу. Спершу складається таблиця з можливими альтернативами рішень для кожної з сторін, яка пов'язується з попередніми результатами матрицею узгодженості, де кожній парі альтернатив $a_{j1}^{(i1)}, a_{j2}^{(i2)}$ параметрів $F1_{i1}, F2_{i2}$ таблиць двох етапів морфологічного аналізу присвоюється оцінка $c_{i1 j1 i2 j2} \in [-1,1]$. Для розрахунку очікуваних результатів для рішень використаються ці формули:

$$R\left(a_j^{(i)} \left| \left\{a_{j_1}^{(1)}, a_{j_2}^{(2)}, a_{j_3}^{(3)}, \dots, a_{j_N}^{(N)}\right\}\right.\right) = \frac{p_j^{\prime(i)} \cdot \prod_{m=1}^{N}\left(c_{mj_m,ij}+1\right)}{\sum_{k=1}^{n_i}\left(p_j^{\prime(i)} \cdot \prod_{m=1}^{N}\left(c_{mj_m,ij}+1\right)\right)},$$

$$R_j^{(i)} = \sum_{j_1=1}^{n_1}\sum_{j_2=1}^{n_2}\dots\sum_{j_N=1}^{n_N} R\left(a_j^{(i)} \left| \left\{a_{j_1}^{(1)}, a_{j_2}^{(2)}, a_{j_3}^{(3)}, \dots, a_{j_N}^{(N)}\right\}\right.\right) P\left(\left\{a_{j_1}^{(1)}, a_{j_2}^{(2)}, a_{j_3}^{(3)}, \dots, a_{j_N}^{(N)}\right\}\right)$$

Результатом другого етапу є оцінка можливих рішень зі сторони держави та керівництва підприємства. Для розгляду того, як ці результати зміняться від прийняття одною зі сторін якоїсь альтернативи пропонується звернутися до методу перехресного аналізу. Зазвичай для цього методу експертно складається список можливих подій, ймовірності настання яких оцінюються експертами. У рамках даної роботи для цього буде взято отримані у другому етапі морфологічного аналізу результати.

Якщо l_1 і l_2 є відповідно кількістю альтернатив для рішень керівництва підприємства та держави, тоді оцінки альтернатив можна записати як:

$$p(d_p) = \begin{cases} R_p^{(1)}, \text{якщо } p \le l_1 \\ R_{p-l_1}^{(2)}, \text{якщо } l_1 < p \le l_1 + l_2 \end{cases},$$

де $p \in \overline{1, l_1 + l_2}$, а d_p – можливі події, які складаються з можливих альтернатив для рішень керівництва підприємства і держави.

Надалі необхідно побудувати матрицю перехресного впливу [5]. Після розробки початкової подальше заповнення відбувається за допомогою випадкового вибору подій для тестування, де порівнюється ймовірність їхнього настання з випадковим числом, щоб визначити, чи подія відбудеться, чи ні. Після цього обчислюється вплив кожної події на інші події, залежно від того, чи вона настала, чи не настала.

Далі, розраховуються шанси настання події за формулами:

$$Odd\left(p(d_p)\right) = \frac{p(d_p)}{1-p(d_p)},$$

$$Odd\left(p(d_p/d_t)\right) = \frac{p(d_p/d_t)}{1-p(d_p/d_t)}.$$

Після цього розраховується матриця відношення шансів за формулою:

$$D_{pt} = \frac{Odd\left(p(d_p/d_t)\right)}{Odd(p(d_t))}.$$

Шанси не настання події розраховуються за формулою:

$$NOdd\left(p(d_p)\right) = Odd\left(p(d_p)\right) * D_{tp}.$$

Далі використовується метод Монте-Карло [6] для розрахунку ймовірностей для подій з урахуванням перехресного впливу. Так, із множини подій випадково обирається одна, для якої генерується випадкове число від 0 до 1. Якщо це число менше за ймовірність обраної події, вважається, що вона відбудеться, інакше – не відбудеться. В обох випадках обчислюється вплив цієї події на інші події. Алгоритм повторюється для всіх подій, а потім для всіх подій одночасно — багаторазово, з метою підвищення точності. За результатами багатьох ітерацій визначається відношення кількості настання кожної події до загальної кількості запусків, що дозволяє оновити ймовірності подій.

Отримані результати можна використати для дослідження чутливості, що дозволить побачити вплив рішень держави або підприємства на іншого. Аналіз передбачає виділення окремих рішень, щодо яких існує невизначеність. Ймовірності цих рішень змінюються, і модель запускається повторно. За цим можна дослідити, як рішення може вплинути на інші рішення.

4. ПРАКТИЧНЕ ДОСЛІДЖЕННЯ

Для практичної частини дослідження було обрано українське виробниче підприємство машинобудівної галузі ПАТ "Крюківський вагонобудівний завод", діяльність якого впливає на роботу вітчизняних залізниць, що у свою чергу вимагає додаткової уваги при прийнятті рішень.

При консультації з експертами було розроблено морфологічну таблицю, яка характеризує стан підприємства і представлена у таблиці 1. Також сформовано таблицю можливих рішень керівництва підприємства та держави у таблиці 2.

Таблиця 1. Морфологічна таблиця

Характеристичні параметри					
Виготовлені вагони за рік	Фінансовий стан	Тип контрактів	Конкуренто-здатність	Наявність держпідтримки	Можливості для розвитку
1	2	3	4	5	6
1.1Понад 1000	2.1 Стабільно	3.1 Державні	4.1 Так	5.1 Так	6.1 Достатньо
1.2 800-1000	2.2 Невеликий дефіцит	3.2 Приватні	4.2 Ні	5.2 Ні	6.2 Потреба у інвестиціях
1.3 400-800	2.3 Значний борг	3.3 Баланс			6.3 Проблема з людьми
1.4 До 400	2.4 Нестабільно				6.4 Висока конкуренція

Далі виконується розробка та використання опитувальних форм, через які відбувається збір експертних оцінок для методу. Роботу було виконано з використанням інструментарію Google Forms, який забезпечує зручність та універсальність при опитуванні. Після збору даних вони обробляються та зберігаються у необхідному форматі для подальшої роботи.

Для проведення аналізу і передбачення стану підприємства було розроблено програмний продукт, який завантажує зібрані дані та проводить необхідні для обраних методів дослідження. Так, на головному вікні, зображеному на рис. 1, можна побачити можливість швидкого виведення найбільш ймовірної конфігурації стану підприємства, а також кнопки для відкриття загального списку конфігурацій і їхніх ймовірностей, а також результати першого та другого етапів морфологічного аналізу. Окремо наявна кнопка для проведення подальшого перехресного аналізу.

Таблиця 2. Початкова морфологічна таблиця другого етапу

Реакції керівництва	Реакції держави
7.1 Оптимізація витрат	8.1 Фінансова підтримка
7.2 Інвестиції в модернізацію	8.2 Збереження рівня співпраці
7.3 Розширення ринків збуту	8.3 Інвестиції у проєкти оновлення залізниць
	8.4 Регуляторна підтримка
	8.5 Зменшення пріоритетності вітчизняного виробника

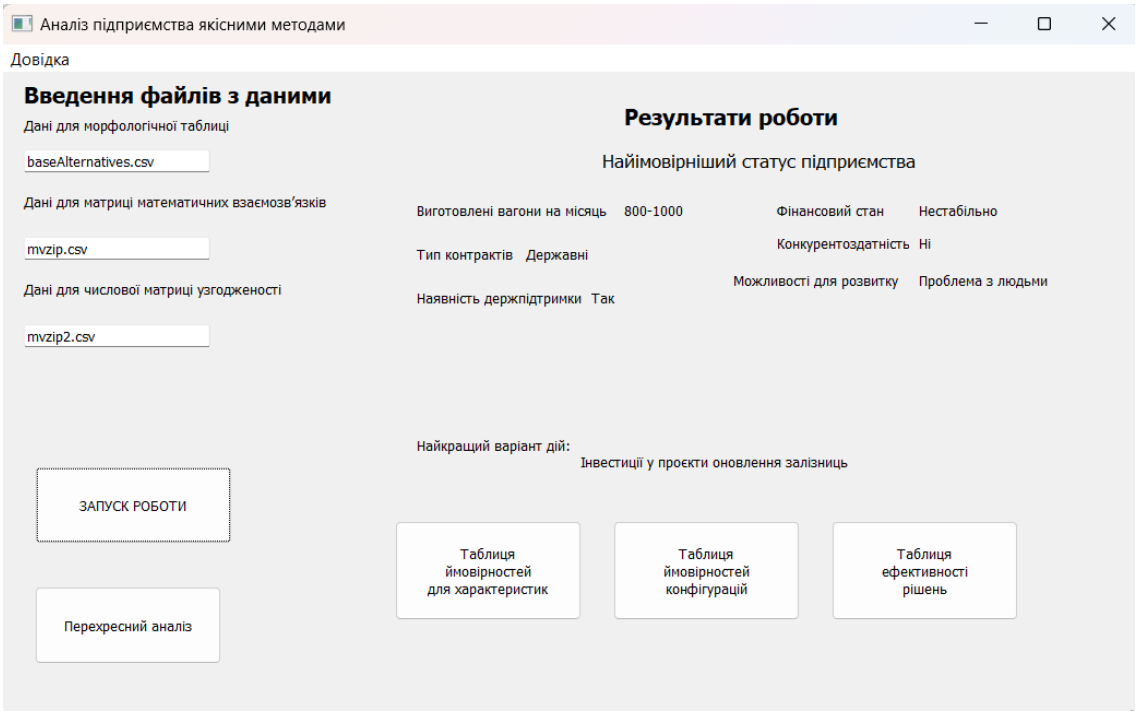
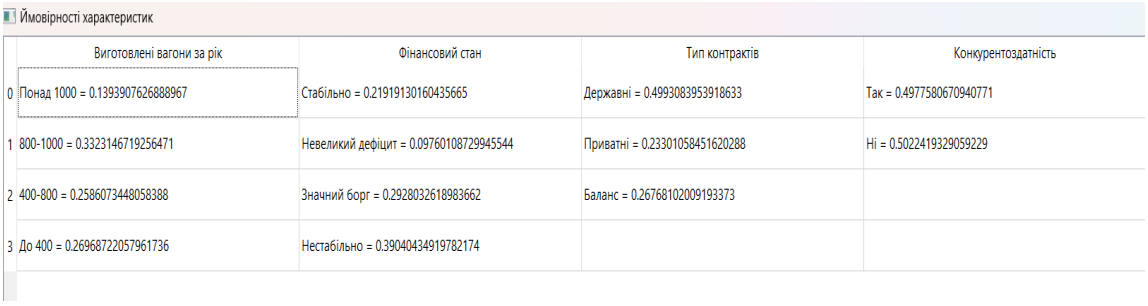


Рисунок 1. Головне вікно програмного продукту

Можна побачити найбільш ймовірну конфігурацію у 800-1000 вагонів виготовлених, переважно державні контракти, держпідтримку, нестабільний фінансовий стан, проблеми з персоналом і конкурентоздатністю. Це приблизно відповідає наявним про підприємство даним. Повну таблицю результату першого етапу можна побачити на рис. 2. Також можна побачити, що найбільш ймовірним варіантом дій для держави визначено інвестиції у оновлення залізниць, що зараз так само виконується [7]. Після отримання результатів другого етапу морфологічного аналізу можна записати як події. Далі починається робота з перехресним аналізом. Вводиться також матриця перехресного впливу. Оскільки використовуються дані з другого етапу морфологічного аналізу як події, то вводиться додаткове обмеження на перехресний вплив: події, які були альтернативами в одній колонці не мають впливу одне на одного. Ця запропонована особливість означає, що значення між ними будуть 0. Матриця перехресного впливу зображена у таблиці 3.


Ймовірності характеристик

	Виготовлені вагони за рік	Фінансовий стан	Тип контрактів	Конкурентоздатність
0	Понад 1000 = 0.1393907626888967	Стабільно = 0.21919130160435665	Державні = 0.4993083953918633	Так = 0.4977580670940771
1	800-1000 = 0.3323146719256471	Невеликий дефіцит = 0.09760108729945544	Приватні = 0.23301058451620288	Ні = 0.5022419329059229
2	400-800 = 0.2586073448058388	Значний борг = 0.2928032618983662	Баланс = 0.26768102009193373	
3	До 400 = 0.26968722057961736	Нестабільно = 0.39040434919782174		

Рисунок 2. Результуюча таблиця першого етапу морфологічного аналізу

Таблиця 3. Матриця перехресного впливу

	$p(e_i)$	7.1	7.2	7.3	8.1	8.2	8.3	8.4	8.5
7.1	0.304		0	0	0.1	0.4	0.2	0.4	0.2
7.2	0.24	0		0	0.3	0.2	0.14	0.5	0.1
7.3	0.46	0	0		0.1	0.6	0.6	0.12	0.31
8.1	0.027	0.1	0.7	0.5		0	0	0	0
8.2	0.148	0.5	0.3	0.3	0		0	0	0
8.3	0.355	0.3	0.6	0.2	0	0		0	0
8.4	0.248	0.2	0.1	0.4	0	0	0		0
8.5	0.223	0.46	0.1	0.5	0	0	0	0	

Матриця завантажується до програмного продукту і після обрахунку ймовірностей подій з урахуванням взаємозв'язків проводиться дослідження чутливості рішень. В рамках даної роботи буде взято, що кожна подія яка задається буде задаватися ймовірністю 1.0. Спочатку розглядається, якщо держава інвестує кошти у розвиток залізниці, результати на Рисунку 3.

Pandas Table in PyQt5

	old probability	new probability	difference
7.1	0.3042082498837502	0.4201627445765236	0.1159544946927734
7.2	0.2406356141993649	0.4003944488377772	0.1597588346384122
7.3	0.4551561359168848	0.5983151123159524	0.1431589763990675
8.1	0.0269155398279154	0.0269155398279154	0.0
8.2	0.1477376758494342	0.1477376758494342	0.0
8.3	0.355007711838695	0.355007711838695	0.0
8.4	0.2476643689407521	0.2476643689407521	0.0
8.5	0.2226747035432031	0.2226747035432031	0.0

Рисунок 3. Зміни у ймовірностях за умови інвестицій у залізниці

За результатами встановлено, що найбільше зростуть інвестиції у модернізацію.

Далі розглядається випадок коли керівництво підприємства вкладеться у модернізацію виробництва, результат на рисунку 4.

Pandas Table in PyQt5

	old probability	new probability	difference
7.1	0.3042082498837502	0.3042082498837502	0.0
7.2	0.2406356141993649	0.2406356141993649	0.0
7.3	0.4551561359168848	0.4551561359168848	0.0
8.1	0.0269155398279154	0.0351452568700644	0.008229717042149
8.2	0.1477376758494342	0.1812455062026319	0.0335078303531977
8.3	0.355007711838695	0.4265516216553013	0.0715439098166063
8.4	0.2476643689407521	0.3889982347544855	0.1413338658137333
8.5	0.2226747035432031	0.2505232175350518	0.0278485139918487

Рисунок 4. Зміни, якщо завод вкладеться у модернізацію потужностей

За результатом можна побачити, що у цьому випадку держава вкладеться у регуляторні зміни, що буде дуже корисним при ввезенні обладнання чи швидшому отриманні сертифікації

для експлуатації. Також існує цікаве спостереження щодо того, що рішення зі сторони держави сильніше впливають на прийняття рішень на підприємстві ніж коли навпаки.

5. ВИСНОВКИ

Під час виконання роботи було розглянуто використання методів якісного аналізу для аналізу та передбачення стану виробничого підприємства ПАТ “Крюківський вагонобудівний завод”. Експертні дані збиралися за допомогою опитування за допомогою форм, а для дослідження було реалізовано програмний продукт.

Було запропоновано і використано комбінацію двох методів. Спочатку за допомогою першого етапу морфологічного аналізу сформовано і оцінено стан підприємства і можливі його конфігурації, а потім у рамках другого етапу визначено та оцінено можливі рішення керівництва підприємства та держави. Далі, отримані на другому етапі дані було використано у якості базових вхідних ймовірностей для методу перехресного аналізу, за результатами якого було проаналізовано чутливість рішень держави до керівництва підприємства і навпаки.

Через об’єднання методів було введено деякі обмеження. Так, через особливості альтернатив характеристичних параметрів у перехресному аналізі не розглядався вплив рішень держави на державу, і керівництва підприємства на дії керівництва підприємства. Також було формально записано переведення фінальної таблиці з рішеннями з морфологічного аналізу у таблицю подій для перехресного аналізу.

Робота показала, що використання подібної комбінації може покращити результати аналізу підприємств якісними методами в загальному, чи у якості етапу у більших дослідженнях. Також розроблений програмний продукт показав потенціал до застосовування у подальших практичних задачах, також може виступити базою для подальшого допрацювання до повноцінної системи підтримки прийняття рішень.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Касич А. О., Яковенко Я. Ю. Проблеми впровадження МСФЗ в Україні та шляхи їх вирішення. Облік і фінанси. 2013. № 1(59). С. 128–132.

2. Жукова, Т.А. Сучасні проблеми аудиторських послуг в Україні [Текст] / Т.А. Жукова, К.О. Єрмоленко, І.Й. Плікус // Вісник Сумського державного університету. Серія Економіка. – 2020. – № 1. – С. 86-92.

3. Згуровський М.З. Основи системного аналізу / М.З. Згуровський, Н.Д. Панкра- това. – К.: BHV, 2007. — 544 с

4. Панкратова Н.Д., Савченко І.О. Застосування методу морфологічного аналізу до задач технологічного передбачення // Наукові праці / Миколаївський держ. гуманітарний ун-т ім. Петра Могили комплексу НаУКМА. Сер. Комп’ютерні технології, системний аналіз, моделювання. 2008. 90, вип. 77. – С. 6–13.

5. Оцінка достовірності результатів аналізу перехресного впливу при розв’язанні задач технологічного передбачення / Д.Є. Пилипенко // Систем. дослідж. та інформ. технології. — 2008. – № 3. – С. 129-140.

6. Roponen, J., & Salo, A. (2024). A probabilistic cross-impact methodology for explorative scenario analysis. *Futures & Foresight Science*, 6, e165. – Режим доступу: URL: https://doi.org/10.1002/ffo2.165

7. План для Ukraine Facility [Електронний ресурс] // UKRAINE FACILITY. – Режим доступу: URL: https://www.ukrainefacility.me.gov.ua/wp-content/uploads/2024/03/plan-ukraine-facility.pdf (дата звернення 10.11.2024).

СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ТУРИЗМУ З ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ ГЛИБОКОГО НАВЧАННЯ

Данилов В.Я., Джалаганія Б.І.[1]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] boriadzhalaganiya@gmail.com

Туризм є важливою складовою економіки, яка сприяє культурному обміну, міжнародному співробітництву та економічному зростанню. У сучасних умовах цифровізації виникає потреба в інноваційних підходах до прогнозування туристичних потоків. Для цього доцільно створити систему підтримки прийняття рішень на основі глибокого навчання. Мета роботи — розробка такої системи з використанням нейронних мереж, з акцентом на аналіз методів обробки даних, порівняння моделей прогнозування та впливу фрактальної інтерполяції на точність. Результатом є система, що прогнозує туристичні показники за допомогою сучасних нейронних мереж і фрактальної обробки даних, використовуючи теоретичні й практичні підходи для аналізу результатів.

Ключові слова: GRU, LSTM, фрактальна інтерполяція, прогнозування

1. ВСТУП

Туризм відіграє ключову роль у сучасній економіці та суспільстві, сприяючи культурному обміну, міжнародній співпраці та створенню нових робочих місць. Завдяки стрімкому розвитку цифрових технологій і глобалізації, туристична галузь зіштовхується з новими викликами, які вимагають інноваційних підходів до управління туристичними потоками та прогнозування попиту. Прогнозування туристичних показників є важливим інструментом для розробки ефективних стратегій розвитку галузі, дозволяючи забезпечити збалансоване використання ресурсів та адаптацію до змін у поведінці споживачів.

Сучасна динаміка туризму характеризується нерівномірним розподілом потоків, що зумовлено сезонністю, змінами економічних умов і різкими зовнішніми факторами, такими як пандемії чи економічні кризи. Використання систем підтримки прийняття рішень у туризмі дозволяє не лише адаптуватися до цих умов, а й виявляти потенційні можливості для розвитку регіонів і бізнесу.

Застосування методів глибокого навчання відкриває нові перспективи в аналізі великих обсягів туристичних даних і створенні високоточного прогнозування. Завдяки цим методам можливо моделювати складні взаємозв’язки між змінними, враховуючи як прямі залежності, так і приховані закономірності. Інтеграція фрактальної обробки даних, зокрема методів інтерполяції, дозволяє покращити якість прогнозів, особливо за умов нерівномірного розподілу даних.

Зміни в поведінці туристів і зовнішні фактори, такі як пандемії чи політична нестабільність, суттєво впливають на попит у туристичній галузі, що підкреслює необхідність вдосконалення підходів до прогнозування. Саме тому використання сучасних інструментів,

таких як штучний інтелект і машинне навчання, стає важливою складовою для розвитку туризму як в Україні, так і в усьому світі.

2. ПОСТАНОВКА ЗАДАЧІ

У сучасних умовах стрімкого розвитку туризму, зростання потоків туристів і збільшення обсягів даних виникає потреба в автоматизації процесів аналізу та прогнозування. Туристична галузь стикається з низкою викликів, серед яких виділяються нерівномірний розподіл туристичних потоків у часі й просторі, залежність від сезонності, зміни в поведінці споживачів і вплив зовнішніх чинників, таких як пандемії, економічні кризи чи природні катаклізми. Для ефективного управління цими викликами потрібні інноваційні інструменти, які дозволять вчасно реагувати на зміни та забезпечувати стабільний розвиток галузі.

Метою цієї роботи є розробка системи підтримки прийняття рішень у сфері туризму, яка дозволяє здійснювати прогнозування туристичних показників із використанням нейронних мереж глибокого навчання. Основними завданнями роботи є:

1. Аналіз і підготовка даних про туристичну діяльність.
2. Розробка моделей прогнозування на основі рекурентних нейронних мереж (LSTM та GRU).
3. Вивчення впливу різних методів фрактальної інтерполяції на точність моделей, зокрема методів Хігучі та сплайн-інтерполяції.
4. Порівняння результатів прогнозування моделей, навчених на звичайних даних та фрактально інтерпольованих наборах.
5. Розробка рекомендацій щодо використання підходів до прогнозування туристичних показників.

У процесі роботи здійснюється тестування моделей на реальних даних туристичної галузі, що дозволяє перевірити їхню практичну ефективність. Інтеграція фрактальної обробки даних спрямована на підвищення точності прогнозів у випадках нерівномірного розподілу або недостатньої кількості даних. Завдяки цьому система забезпечує гнучкість у роботі з різними сценаріями, що є важливим для стратегічного планування та ухвалення рішень у туризмі.

Значущість цієї роботи полягає у впровадженні новітніх технологій машинного навчання та обробки даних, що дозволяють підвищити конкурентоспроможність туристичної галузі, сприяти раціональному використанню ресурсів і поліпшити планування туристичної діяльності як на локальному, так і на глобальному рівні.

3. ОПИС НАБОРУ ДАНИХ

Дані, що використовуються у цьому дослідженні, представляють собою набір, який містить інформацію про туристичні локації з різних країн світу. Набір даних включає наступні атрибути:

1. Location – унікальний ідентифікатор туристичної локації.
2. Country – країна, де знаходиться локація.
3. Category – категорія туристичного об’єкта (наприклад, природа, історія, пляжі тощо).
4. Visitors – кількість відвідувачів за певний період часу. Цей параметр використовується як цільова змінна для моделювання.
5. Rating – середня оцінка туристичного об'єкта за відгуками відвідувачів.
6. Revenue – дохід, отриманий від туристичної локації.
7. Accommodation_Available – наявність доступного проживання поблизу (Yes/No).

Набір даних містить кількісні змінні (Visitors, Rating, Revenue) та якісні змінні (Country, Category, Accommodation_Available), що забезпечує різноманітність для аналізу та моделювання.

Для виконання задач прогнозування використовується параметр Visitors, який моделюється за допомогою методів аналізу часових рядів. Для забезпечення точності прогнозів застосовуються підходи нормалізації, стандартизації та методи фрактальної інтерполяції, що дозволяє працювати навіть із нерівномірними даними. Застосування фрактального аналізу, зокрема методу Хігучі, дозволяє підвищити якість роботи з даними.

Цей набір даних є репрезентативним і включає інформацію з різних регіонів, що робить його придатним для створення моделей прогнозування, які можуть бути використані для ухвалення рішень у сфері туризму.

4. ОПИС АЛГОРИТМУ

У цьому дослідженні використовуються сучасні алгоритми для моделювання та аналізу даних, зокрема методи нейронних мереж та фрактальної інтерполяції. Вони забезпечують високу точність прогнозування та дозволяють гнучко адаптуватися до різноманітних структур даних.

Нейронні мережі

Нейронні мережі є основним інструментом для аналізу даних у цій роботі. Вони дозволяють виявляти складні залежності між вхідними та вихідними параметрами, що є критичним для задач прогнозування часових рядів. У роботі використовуються такі типи нейронних мереж:

1. Feedforward Neural Networks (FNN)

Це базова архітектура, у якій інформація проходить через нейрони лише в одному напрямку – від вхідного шару до вихідного. FNN використовуються для аналізу даних, що не мають часової залежності, але можуть бути корисними для попередньої оцінки важливих параметрів.

2. Gated Recurrent Unit (GRU)

GRU є різновидом рекурентних нейронних мереж (RNN), спеціалізованих на обробці послідовних даних. Вони мають механізм запам’ятовування довгострокових залежностей у часових рядах, що дозволяє ефективно аналізувати тенденції у туристичних потоках. GRU обирається через його ефективність у навчанні та меншу кількість параметрів порівняно з іншими RNN.

Для навчання нейронних мереж використовуються методи оптимізації, такі як градієнтний спуск, що мінімізує функцію втрат, наприклад, середньоквадратичну помилку (MSE). Крім того, застосовуються регуляризація та нормалізація для покращення узагальнювальної здатності моделей.

Методи фрактальної інтерполяції

Фрактальна інтерполяція є потужним методом для роботи з нерівномірними даними або часовими рядами, які мають складну структуру. Вона дозволяє моделювати дані з урахуванням їхньої фрактальної природи, що особливо актуально для туристичних потоків, які можуть демонструвати хаотичні коливання.

1. Метод Хігучі

Алгоритм Хігучі застосовується для обчислення фрактальної розмірності часових рядів. Він допомагає оцінити складність даних та визначити, наскільки сильно вони відхиляються від

регулярних тенденцій. Цей показник використовується як додаткова характеристика для моделювання.

2. Фрактальна інтерполяція даних

Цей метод створює штучні точки даних між наявними шляхом використання фрактального параметра (D). Він дозволяє зберігати природну варіативність ряду, додаючи деталізацію, яка відображає реальні коливання туристичних потоків. Інтерполяція базується на використанні випадкових змін, зважених відповідно до обчисленої фрактальної розмірності, що забезпечує правдоподібність створених даних.

Застосування фрактальної інтерполяції дозволяє значно покращити результати навчання нейронних мереж, забезпечуючи їм більш інформативні та структуровані вхідні дані. Цей підхід є новаторським і демонструє високу ефективність у задачах прогнозування туристичних потоків.

У поєднанні нейронні мережі та методи фрактальної інтерполяції утворюють потужний інструментарій, що дозволяє досягти високої точності моделювання та аналізу, враховуючи як локальні, так і глобальні особливості даних.

5. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

У процесі дослідження було проведено оцінку моделей GRU та LSTM, навчання яких здійснювалося на трьох типах даних: звичайних, фрактально інтерпольованих за методом Хігучі та фрактально інтерпольованих із використанням сплайн-інтерполяції.

Результати оцінювання показали, що моделі, навчені на фрактально інтерпольованих даних, демонструють значно вищу точність у порівнянні зі звичайними даними. Зокрема, найкращі результати були отримані на фрактально інтерпольованих даних із використанням сплайн-інтерполяції (табл. 1).

Таблиця 1. Точність моделей в залежності від даних

	GRU(MSE)	LSTM(MSE)	GRU(R^2)	LSTM(R^2)
Звичайні дані	0.0866	0.0854	–0.0173	–0.0032
Хігучі	0.0246	0.0307	0.6061	0.5082
Сплайн-інтерполяція	0.0006	0.0066	0.9916	0.9082

На графіках чітко видно покращення збіжності передбачених значень із реальними даними при використанні моделей, навчених на фрактально інтерпольованих даних (рис. 1–3).

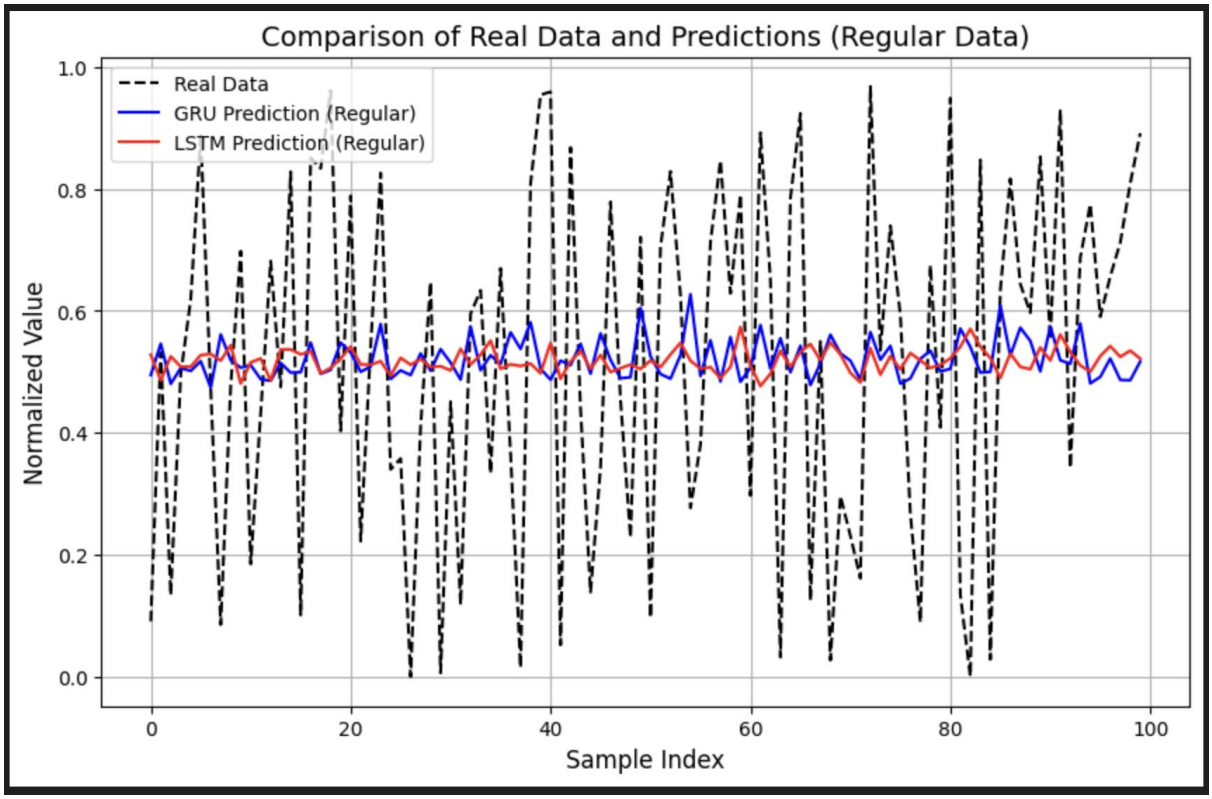


Рисунок 1. Порівняння результатів моделей з реальними даними, навченими на звичайних даних

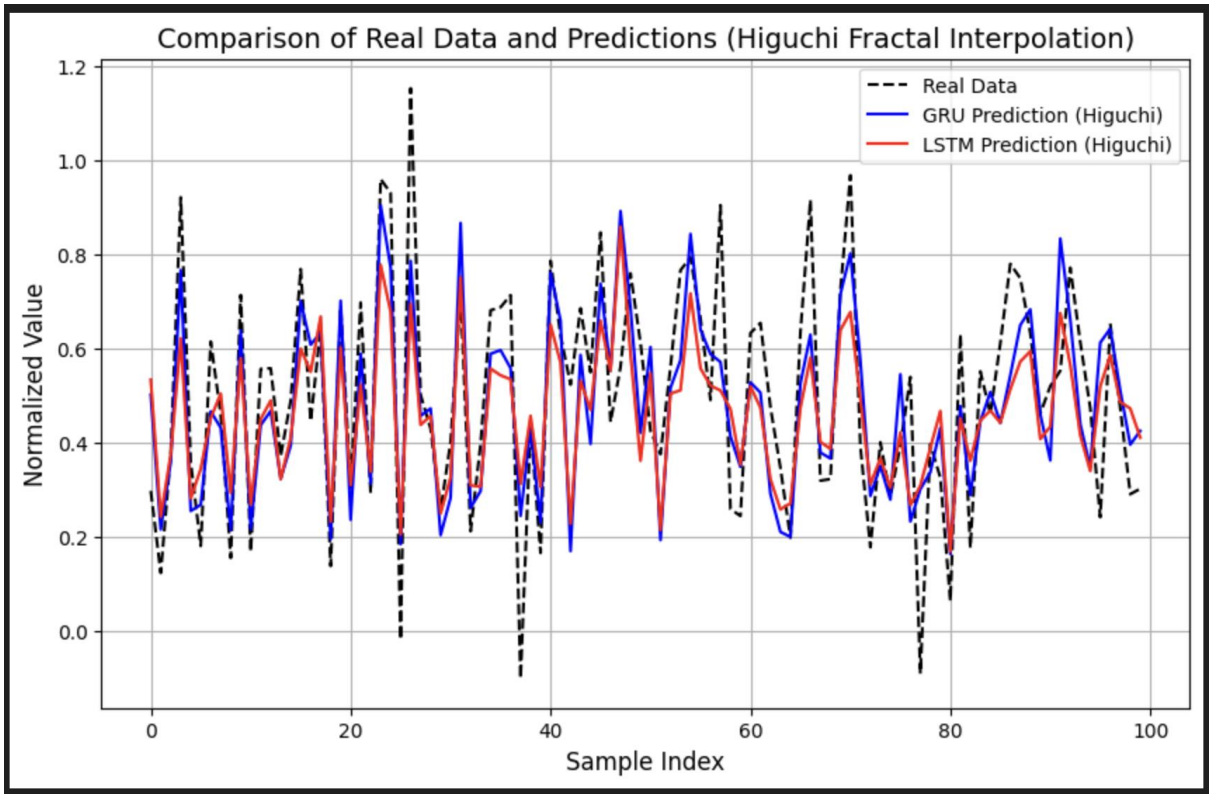


Рисунок 2. Порівняння результатів моделей з реальними даним, навченими на фрактально інтерпольованих даних з хігучі

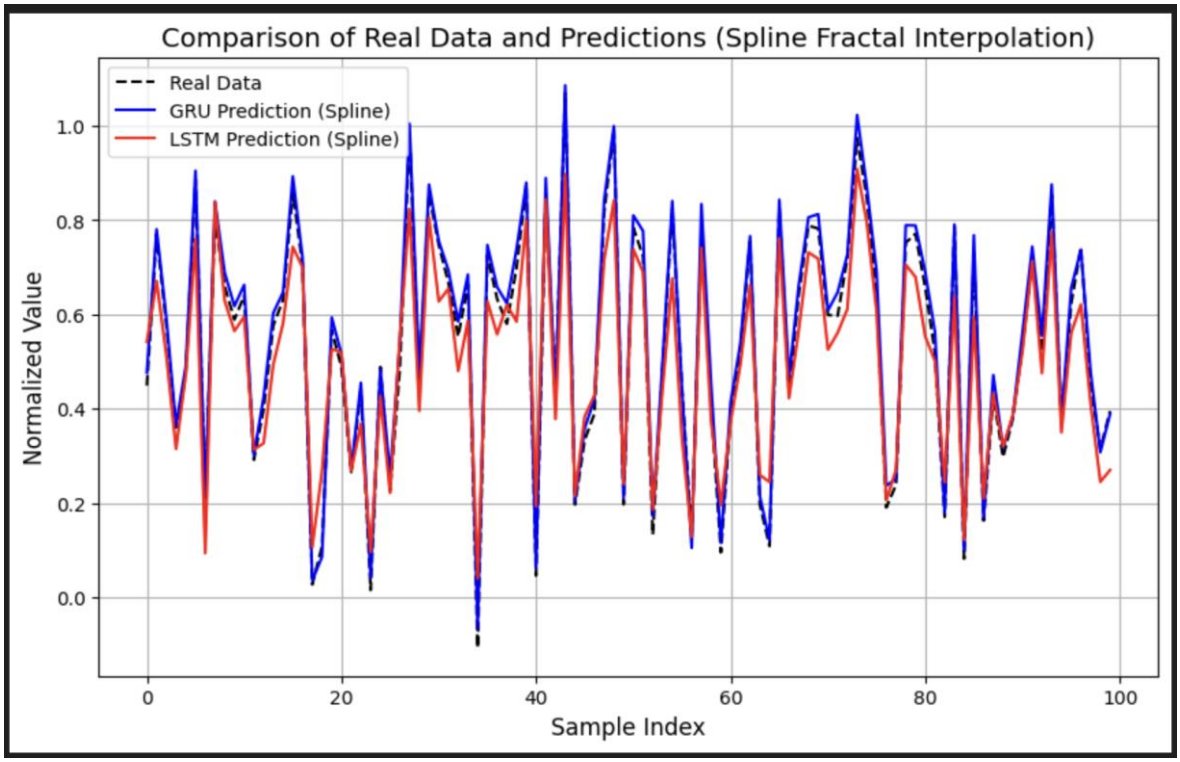


Рисунок 3. Порівняння результатів моделей з реальними даними, навчених на графічно інтерпольованих даних (сплайн-інтерполяції)

6. ВИСНОВОК

У проведеному дослідженні було розроблено та оцінено інноваційний підхід до прогнозування туристичних показників за допомогою моделей глибокого навчання (GRU, LSTM), які навчалися на різних типах даних: звичайних та фрактально інтерпольованих. Запропоновані методи фрактальної інтерполяції, зокрема із використанням методів Хігучі та сплайн-інтерполяції, внесли новизну в обробку даних, що дозволило суттєво підвищити точність прогнозів, про що свідчать отримані значення метрик MSE та R^2. Найкращі результати були досягнуті при застосуванні сплайн-інтерполяції, де моделі продемонстрували високу збіжність передбачень із реальними даними.

Новаторський підхід із використанням фрактальної інтерполяції значно знижує похибки моделі та підвищує її здатність до узагальнення. Це підтверджує ефективність обраної стратегії, яка поєднує глибокі нейронні мережі та передові методи обробки даних. Зокрема, цей підхід має важливе значення для задач прогнозування в туристичній галузі, де точність передбачень є критично важливою для прийняття ефективних управлінських рішень.

Дослідження відображає новизну застосованого методу та відкриває перспективи для подальшого вдосконалення алгоритмів прогнозування. Результати підтверджують доцільність впровадження інноваційних технік фрактальної обробки даних для моделювання в різних галузях, створюючи основу для нових відкриттів і розробок.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Fractal Methods in Data Analysis: Applications in Time Series and Forecasting, First Online: 10 March 2024, pp57–77 https://link.springer.com/

2. Time series analysis via network science: Concepts and algorithms, First published: 01 March 2021, https://wires.onlinelibrary.wiley.com/doi/10.1002/widm.1404

СИСТЕМА ВИЯВЛЕННЯ АКТИВНОСТІ ГОЛОСУ ТА РОЗПІЗНАВАННЯ МОВИ НА ОСНОВІ ГЛИБОКИХ НЕЙРОННИХ МЕРЕЖ

Заяць В.А.[1], Данилов В.Я.

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] vzayats000@gmail.com

Системи виявлення голосової активності та розпізнавання мови є важливими компонентами сучасних технологій обробки мовлення. Їхня ефективність визначає успішність багатьох застосувань, таких як голосові асистенти, автоматичні перекладачі та медичні системи. Використання глибоких нейронних мереж забезпечує високі показники точності виявлення голосової активності (VAD) і розпізнавання мовлення (ASR), що робить їх надзвичайно актуальними для реальних застосувань.

У цій роботі представлено розробку системи, яка поєднує VAD, побудований на архітектурі ResNet50, та систему розпізнавання мовлення, що базується на моделі Whisper. Основною метою дослідження було створення комплексного рішення для перетворення аудіопотоку в текст у режимі реального часу. Результатом є інтегрована система, що дозволяє виявляти голосові відрізки у потоці аудіо та конвертувати їх у текст із високою точністю.

Ключові слова: голосова активність, розпізнавання мовлення, нейронні мережі, машинне навчання.

1. ВСТУП

Дослідження процесів розпізнавання мовлення та виявлення голосової активності є надзвичайно актуальним, оскільки ці технології знаходять широке застосування у різних сферах, таких як голосові асистенти, автоматичні переклади, системи безпеки та медичні платформи. Вони виступають ключовими елементами для інтерактивної взаємодії між людиною та комп'ютером і сприяють розвитку сучасних цифрових рішень.

Зростаючий попит на системи, що забезпечують якісне розпізнавання мовлення в реальному часі, висуває нові вимоги до їх точності, швидкодії та адаптивності. Аналіз голосової активності (Voice Activity Detection, VAD) є важливим етапом попередньої обробки мовного сигналу, який дозволяє виділяти активні фрагменти мовлення для подальшого аналізу.

Для реалізації таких систем застосовуються сучасні підходи, засновані на використанні глибоких нейронних мереж (Deep Neural Networks, DNN), які забезпечують високу точність та ефективність. Впровадження цих технологій у практику дозволяє значно покращити якість обробки мовних даних і сприяє розробці інноваційних рішень у сфері штучного інтелекту.

2. РОЗРОБКА СИСТЕМИ ВИЯВЛЕННЯ ГОЛОСОВОЇ АКТИВНОСТІ ТА РОЗПІЗНАВАННЯ МОВИ

Створення ефективних систем розпізнавання мови сьогодні є найважливішим завданням для розвитку людино-машинної взаємодії. З цією метою провідними технологічними компаніями були розроблені різноманітні архітектури нейронних мереж, серед яких особливу увагу привертає ResNet50 через її ефективність у задачах класифікації. Ця архітектура успішно застосовується для виявлення голосової активності (VAD) завдяки здатності виділяти складні ієрархічні ознаки з аудіосигналу.

Розробка системи VAD базується на двоетапному підході:

1. Виявлення голосової активності (VAD) за допомогою ResNet50;
2. Розпізнавання мови за допомогою моделі Whisper.

Процес обробки аудіосигналу відбувається за наступною схемою:

- Отримання аудіопотоку в реальному часі;
- Сегментація аудіопотоку на фрагменти;
- Аналіз кожного фрагменту за допомогою VAD;
- Передача фрагментів з виявленою голосовою активністю на модель розпізнавання мови.

Для навчання моделі VAD використано датасет RealVAD, що містить різноманітні зразки мовлення в реальних умовах. Попередньо натренована модель ResNet50 була адаптована для задачі бінарної класифікації аудіофрагментів на наявність чи відсутність голосової активності.

Для оцінки ефективності системи використовуються такі метрики як точність (precision), повнота (recall) та F1-score. Модель демонструє високу точність у виявленні голосової активності, що є критичним для подальшого успішного розпізнавання мови. Застосування методів глибокого навчання дозволяє системі адаптуватися до різних умов запису та типів голосів.

У ході дослідження проведено тестування розробленої системи на реальних аудіоданих для аналізу її ефективності. Результати роботи алгоритмів та деталі експериментів представлено далі.

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Вихідними даними для розробки та тестування системи було використано датасет RealVAD, що містить аудіозаписи реального мовлення у різних умовах. RealVAD - це набір даних, спеціально створений для тренування та оцінки систем виявлення голосової активності, що включає розмітку часових відрізків з наявністю та відсутністю мовлення.

Оскільки класи в вибірці є незбалансованими (періоди тиші зазвичай переважають над періодами мовлення), для підвищення якості класифікації впроваджено балансування класів за допомогою методу аугментації даних. За основний критерій оцінки ефективності моделі було обрано F1-score. Також для моделі побудовано ROC-криву та розраховано основні метрики якості класифікації.

Таблиця 1. Результати тестування системи VAD

Метрика	Значення
Точність	0.94
Повнота	0.92
F1-score	0.93
ROC AUC	0.96

Модель ResNet50 продемонструвала високу ефективність у виявленні голосової активності, що підтверджується високими показниками всіх метрик (табл. 1). Особливо важливим є високе значення F1-score, що свідчить про збалансованість між точністю та повнотою класифікації. Для оцінки швидкодії системи проведено тестування в режимі реального часу. Результати показали, що система здатна обробляти аудіопотік з затримкою менше 100 мс, що є прийнятним для більшості практичних застосувань.

Для оцінки ефективності повної системи розпізнавання мови було проведено тестування на наборі тестових фраз різної складності. Результати показали, що попереднє виявлення голосової активності значно підвищує точність розпізнавання мови та зменшує обчислювальне навантаження на систему завдяки фільтрації неголосових сегментів (табл. 2).

Таблиця 2. Результати тестування повної системи розпізнавання мови

Метрика	З VAD	Без VAD
Word Error Rate (WER)	5.2%	8.7%
Real-time Factor (RTF)	0.8	1.4
Середній час обробки фрази (мс)	450	750

4. ВИСНОВКИ

У ході дослідження було розроблено та реалізовано ефективну систему виявлення голосової активності та розпізнавання мови на основі глибоких нейронних мереж. Запропонована архітектура, що базується на моделі ResNet50 для VAD та інтеграції з Whisper для розпізнавання мови, продемонструвала високу точність та надійність в умовах реального часу. Експериментальні результати показали, що система досягає F1-score 0.93 для задачі виявлення голосової активності та забезпечує зменшення word error rate на 3.5% порівняно з системами без попередньої фільтрації голосової активності. Важливим досягненням є забезпечення обробки аудіопотоку з затримкою менше 100 мс, що робить систему придатною для широкого спектру практичних застосувань. Використання попередньо навченої моделі ResNet50 з подальшим доналаштуванням на специфічному наборі даних RealVAD дозволило досягти оптимального балансу між обчислювальною складністю та якістю розпізнавання. Розроблена система демонструє стійкість до різних рівнів фонового шуму та варіацій в характеристиках голосу, що підтверджує її практичну цінність для реальних застосувань. Подальші дослідження можуть бути спрямовані на оптимізацію швидкодії системи та розширення її можливостей для роботи з різними мовами та акцентами.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., ... & Zhu, Z. (2016). Deep speech 2: End-to-end speech recognition in English and Mandarin. International Conference on Machine Learning (pp. 173-182).

2. Graves, A., Mohamed, A. r., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (pp. 6645-6649). IEEE.

3. Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., ... & Zhu, Z. (2014). DeepSpeech: Scaling up end-to-end speech recognition. arXiv preprint arXiv:1412.5567.

4. Zhang, Y., Chan, W., & Jaitly, N. (2017). Very deep convolutional networks for end-to-end speech recognition. 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4845-4849). IEEE.

5. Watanabe, S., Hori, T., Kim, S., Hershey, J. R., & Hayashi, T. (2017). Hybrid CTC/attention architecture for end-to-end speech recognition. IEEE Journal of Selected Topics in Signal Processing, 11(8), 1240-1253.

6. Li, J., Deng, L., Haeb-Umbach, R., & Gong, Y. (2015). Robust automatic speech recognition: A bridge to practical applications. Academic Press.

7. Li, J., Deng, L., Haeb-Umbach, R., & Gong, Y. (2015). Robust automatic speech recognition: A bridge to practical applications. Academic Press.

8. Chen, X., Wu, Y., Gao, F., & Wang, S. (2019). Voice activity detection using deep neural networks and boosted decision trees. Speech Communication, 107, 1-10.

9. Sainath, T. N., & Parada, C. (2015). Convolutional neural networks for small-footprint keyword spotting. Interspeech (pp. 1478-1482).

АНАЛІЗ РИНКУ КРИПТОВАЛЮТ З ВИКОРИСТАННЯМ МЕТОДІВ МАШИННОГО НАВЧАННЯ ТА ТЕОРІЇ ІГОР

Івашова А.П.[1] , Зайченко О.Ю.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] nastyaivashova2002@gmail.com, [2] zaichenko.helen@lll.kpi.ua [0000-0001-9662-3269]

Аналіз криптовалютних ринків є складним та непередбачуваним, оскільки необхідно включати багато факторів. Мета цього дослідження полягає у полегшені процесу прогнозування та підвищення точності оптимізації стратегії для кращого розуміння ринкових тенденцій. У ході робити було застосовано методи машинного навчання для оцінки цін купівлі і продажу та теорії ігор для вдосконалення рішень трейдера. Її наукова новизна полягає у поєднані обох підходів для аналізу ринка, що дозволяє підвищити ефективність аналіз ринку.

Ключові слова: аналіз криптовалютних ринків, прогнозування, оптимізація, методи машинного навчання, теорія ігор, ефективність.

1. ВСТУП

Ринки криптовалют є швидкозмінними та непередбачуваними, що створює ризики, так і можливості для трейдерів. Традиційні методи фінансового аналізу іноді є неефективними для широкого розуміння стану руху ринку. Відповідно до цього виникла потреба у застосуванні сучасних технологій, котрі можуть підвищити точність прогнозування ціни купівлі/продажу, оптимізувати стратегію та уникнути ризиків, що важко визначити під час торгівлі активами.

Криптовалюта – це форма цифрової валюти, котра існує в рамках децентралізованої бази даних для транзакцій. Децентралізованість полягаю у здійснені транзакції між користувачами без залучення посередників [1]. Зазвичай криптовалюта не визнається як платіжний засіб у більшості країн, однак в Україні є легалізованою та регулюється законом «Про віртуальні активи», а також в ЄС має нормативну базу для підтримки фінансової стабільність. До найпопулярніших криптовалют відносяться Bitcoin, Ethereum, Tether та BNB [1].

Головні фактори при торгівлі криптовалютами включають вибір правильної стратегії, тип позиції, метод аналізу ринку та ефективне управління ризиками. Торгівля активами охоплює різні підходи такі як денна торгівля, позиційна торгівля та інше. Також, важливим є тип позиції (довга та коротка) та методи аналізу (фундаментальний та технічний) [2]. Всі ці аспекти повинен враховувати сам трейдер, хоча він може помилитися та недооцінити ризик. Сучасні технології намагаються полегшити цей процес для користувача та дозволяють приймати швидкі рішення, що і робить дане дослідження актуальним як з наукової точки зору – через поєднання різних підходів, так і з практичної точки зору – через користь для трейдерів.

2. ЗАСТОСУВАННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ТА ТЕОРІЇ ІГОР

Як зазначалося вище у вступі, торгівля криптовалютами є достатньо складним процесом, що потребує автоматизації. Різноманітні методи машинного навчання у поєднанні з теорією ігор дозволяють адаптуватися до динамічного ринку та приймати ефективні рішення. Методи машинного навчання знаходять приховані закономірності та моделі, котрі є корисними для прогнозування цін криптовалют.

У даній роботи застосовувалося навчання з вчителем, оскільки дозволяє класифікувати чи передбачити цільові позначки при наявних даних. Для навчання моделі виділяється 80% даних, а для тренування – 20%, такий поділ дозволить уникнути перенавчання моделей. Для прогнозування сигналів купівлі/продажу застосовуються наступні класифікаційні моделі:

1. Random Forest застосовують велику кількість дерев рішень, де кожне дерево може надавати власний прогноз, а остаточне рішення приймається на основі голосування дерев [3].
2. AdaBoost – це метод ансамблевого навчання, що послідовно тренується декілька слабких моделей, кожна з яких зосереджена на прикладах, котрі були неправильно кваліфіковані іншими моделями [4].
3. Extra Trees є розширенням методу Random Forest, де ще додатково додається випадковість під час вибору розділень вузлів дерев [5].
4. Gradient Boosting додає нові дерева, щоб виправити помилки попередніх дерев, що підвищує точність прогнозування [6].
5. Ensemble Model основується на тому, що декілька не складних моделей можуть давати високу точність у порівнянні з одним складним. Кожна модель спеціалізується на своїх аспектах роботи, а кінцеве рішення відбувається шляхом узгодження моделей [7].

Ці методи були обрані тому, що вони можуть працювати з великими обсягами даних та враховують різні характеристики вибірки. Завдяки варіації вибору алгоритмів та їх ансамблю ці моделі знижують ризик перенавчання, підвищують стабільність та точність прогнозування, що є досить важливим для динамічного ринку криптовалют.

Після прогнозування цін криптовалют починається оптимізація стратегії. Застосування ігор в цій частині роботи дозволяє враховувати динамічну природу ринку криптовалют та його невизначеність. Це підвищує адаптивність стратегії, дозволяючи прогнозувати дії інших учасників та оптимізувати власні рішення. Тому було розроблено наступний алгоритм оптимізації з використанням теорії ігор:

1. Отримання сигналів купівлі/продажу після машинного навчання.
2. Застосування рівноваги Неша. Відбувається через порівняння між цінами купівлі та продажу, де вигода обраховується випадковою змінною, що моделюють економічні фактори.
3. Застосування Байєсівських ігор. Застосовують ймовірнісній підхід, при якому враховуються попередні знання про ринкові умові. Тобто рішення приймається за допомогою прогнозів та очікувань з урахуванням того, як ціни будуть поводитися у майбутньому.
4. Фільтрація сигналів. Прибирає ті сигнали, що не слідують чергуванню купівлі/продажу, з метою зменшення ризику та покращенню оптимізації стратегії.
5. Створення комбінованої стратегії. Тобто об’єднуємо в одну єдину персональну вигідну стратегію.

Цей підхід дозволяє збільшити точність прогнозів та покращити прийняття рішень на основі комплексного аналізу.

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

Результати роботи показують різну ефективність алгоритмів прогнозування сигналів купівлі/продажу та стабільну роботу оптимізації стратегії для Bitcoin та Ethereum. Для визначання якості результатів застосовується оцінки точності прогнозування:

1. Точність (Accuracy Score) – це оцінка, що основана на відношенні кількості правильних передбачень [8].

$$\text{Точність} = \frac{\text{Кількість правильних передбачень}}{\text{Загальна кількість передбачень}}$$

2. Класифікаційний звіт (Classification Report) включає в себе важливі метрики [8]:

а. Точність (Precision) – відсоток адекватних прогнозів серед усіх прогнозів.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

б. Повнота (Recall) – відсоток адекватних прикладів, котрі було прогнозовано.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

в. F1-міра (F1-score) – гармонічне середнє між точністю та повнотою, тобто надає збалансовану оцінку моделі.

$$F1 = \frac{Precision \cdot Recall}{Precision + Recall}$$

г. Support – кількість реальних прикладів для кожного класу в тестовій вибірці.

Для визначення ефективності стратегії застосовують BackTest [7]:

1. Sortino Ratio застосовуються для оцінки ризику та прибутковості криптовалюти. Значення більше 1 свідчить про відмінне вкладення та низькі ризики.

2. Beta вказує на скільки чутлива ціна фінансового активу на зміни ринку. Значення більше 1 свідчить про високу волатильність та ризик. У випадку, коли менше 1, це буде свідчити про стабільність.

3. Alpha показує перевищення чи недостачу прибутковості фінансового активу відносно очікувань. Позитивне значення вказує на додаткову прибутковість.

4. Maximum Drawdown – це метрика, котра визначає максимальну втрату фінансів від пікової вартості до найнижчої за певний проміжок часу. Чим вище її значення, тим більший ризик.

Розглянемо на рис. 1 та рис. 2 результати аналізу криптовалюти Bitcoin за період 2023–2024.

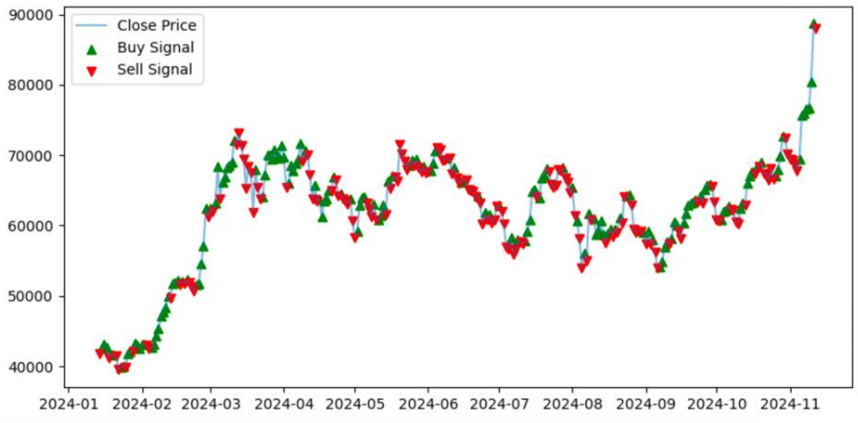


Рисунок 1. Графік прогнозування цін купівлі/продажу Bitcoin

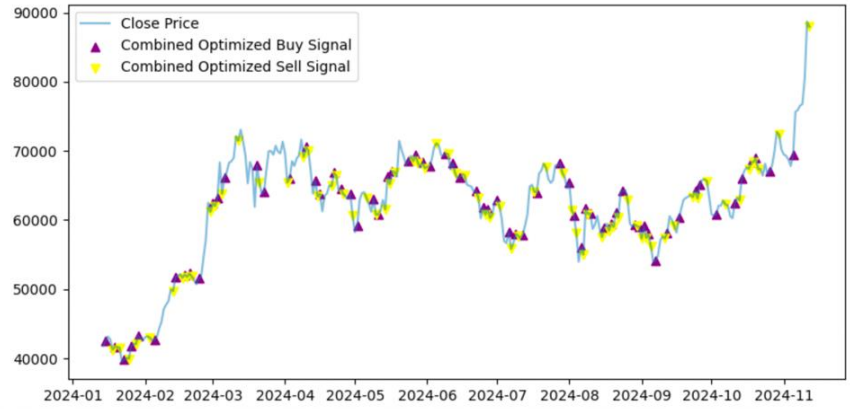


Рисунок 2. Графік оптимальних стратегій цін купівлі продажу Bitcoin

На рис. 3 зображено оцінки якості. Результати продуктивності декількох моделей показує, що найкращу точність надає ансамблевий метод 62%, що є достатньо хорошим результатом. Згідно з класифікаційним звітом для найкращої моделі, точність прогнозування сигналу Sell становить 70% при підтримці 35 прикладів, а для сигналу Buy – 57% прі підтримці 29 прикладів. Сигнал Buy показав кращу повноту порівняно з показником Sell, що вказує на здатність моделі краще передбачати сигнали купівлі та про можливе зростання цін. Макро середні показники вказують на збалансовану продуктивність моделі між класами. Зважене середнє теж стверджує на стабільність моделі.

```
Model Performance:
Optimized Random Forest Accuracy: 0.58
Optimized Gradient Boosting Accuracy: 0.58
Optimized AdaBoost Accuracy: 0.50
Optimized Extra Trees Accuracy: 0.61
Ensemble Model Accuracy: 0.62

Classification Report for Best Model:
              precision    recall  f1-score   supp
ort

        Sell       0.70      0.54      0.61
 35
         Buy       0.57      0.72      0.64
 29

    accuracy                           0.62
 64
   macro avg       0.64      0.63      0.62
 64
weighted avg       0.64      0.62      0.62
 64
```

Рисунок 3. Оцінки якості прогнозування Bitcoin

На рис. 4 BackTest виявив за допомогою фінансових метрик Sortino Ratio, Alpha та Beta високий прибуток стратегії з мінімальними ризиками, а Maximum Drawdown дорівнює 0%, що вказує на стабільність.

```
Sortino Ratio: 8.567
Beta: 0.007
Alpha: 5.775
Maximum Drawdown: 0.001
```

Рисунок 4. BackTest стратегії криптовалюти Bitcoin

Розглянемо на рис. 5 та рис. 6 результати аналізу криптовалюти Ethereum за період 2022–2023.

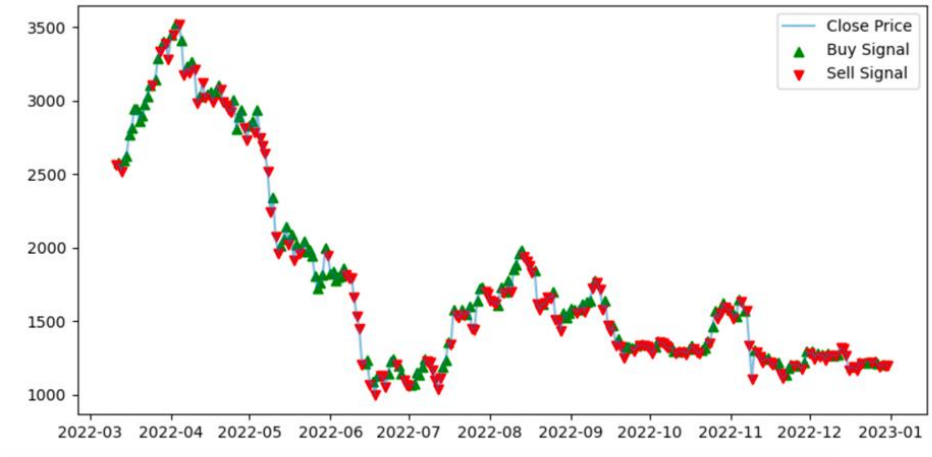


Рисунок 5. Графік прогнозування цін купівлі/продажу Ethereum

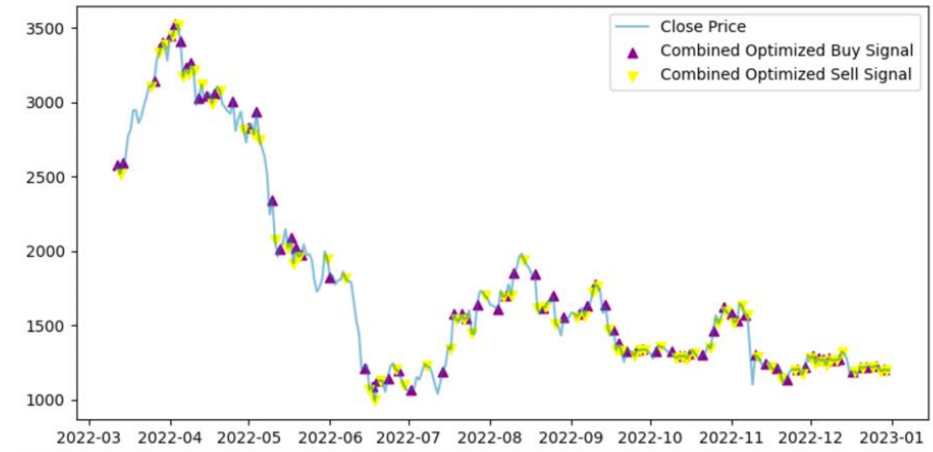


Рисунок 6. Графік оптимальних стратегій цін купівлі продажу Ethereum

На рисунку 7 зображено оцінки якості прогнозування. Результати точність прогнозування в цьому випадку демонструють те, що найвищу оцінку має AdaBoost – 62%. Класифікаційний звіт вказує на те, що для сигналу Sell точність становить 67% при підтримці 33 прикладів, а для сигналу Buy – 57% при підтримці 27 прикладів. Макро середні свідчать про стабільність продуктивності моделі між сигналами. Зважене середнє підтверджує узгодженість результатів моделі. Модель демонструє дещо кращу повноту для сигналів Buy (63%) та Sell (61%), що свідчить про здатність моделі передбачати сигнали

купівлі та продажу з досить рівномірною точністю. На рисунку 11 зображено оцінки якості прогнозування.

```
Model Performance:
Random Forest Accuracy: 0.52
Gradient Boosting Accuracy: 0.45
AdaBoost Accuracy: 0.62
Extra Trees Accuracy: 0.48
Ensemble Model Accuracy: 0.47

Classification Report for Best Model:
              precision    recall  f1-score   supp
ort

        Sell       0.67      0.61      0.63
 33
         Buy       0.57      0.63      0.60
 27

    accuracy                           0.62
 60
   macro avg       0.62      0.62      0.62
 60
weighted avg       0.62      0.62      0.62
 60

Analysis complete. See the graph for signals.
```

Рисунок 7. Оцінки якості прогнозування Ethereum

На рис. 8 BackTest виявив високу дохідність стратегії з незначним ризиком, а Maximum Drawdown становить 15%, що вказує на невеликі ризики, котрі у сукупності не значно випливають на загальний прибуток.

```
Sortino Ratio: 12.125
Beta: -0.091
Alpha: 25.100
Maximum Drawdown: 0.154
```

Рисунок 8. BackTest стратегії криптовалюти Ethereum

4. ВИСНОВКИ

Дослідження доводить, що поєднання методів машинного навчання та теорії ігор може значно підвищити ефективність аналізу ринку криптовалют. Застосування різних методів машинного навчання, зокрема таких як Random Forest, AdaBoost, Extra Trees та Gradient Boosting дозволило створити моделі для прогнозування купівлі/продажу криптовалют з різною точністю та стабільністю. У роботі було застосовано такі критерії оцінки якості, як точність (Accuracy) та класифікаційний звіт (Precision, Recall, F1-score) для визначення продуктивності моделей. Крім того, BackTest дозволяє оцінити ефективність стратегії за допомогою фінансових показників Sortino Ratio, Beta, Alpha та Maximum Drawdown, що підтверджують здатність моделей знижувати ризики та покращувати прибутковість. Отримані результати засвідчують, що моделі можуть забезпечувати високий рівень прогнозування за умов правильної оптимізації стратегії.

Це дослідження має практичну цінність, тому що результати можуть застосовуватися для покращення торгової стратегії та підвищення точності прогнозів на ринку криптовалют, що дозволить трейдерам приймати більш обґрунтовані результати. Подальші дослідження можуть бути спрямовані на розширення моделей для інших фінансових активів або їх вдосконалення під специфічні умови волатильності ринку. Незважаючи на позитивні результати, варто зазначити, що висока мінливість ринку може вплинути на прогностичні показники моделей, що вимагає ретельного контролю та налаштування стратегій.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. What Is Cryptocurrency and How Does It Work? Binance Academy. URL: https://academy.binance.com/en/articles/what-is-a-cryptocurrency (дата звернення: 02.11.2024).

2. A Complete Guide to Cryptocurrency Trading for Beginners. (2020, June 22). Binance Academy. URL: https://academy.binance.com/en/articles/a-complete-guide-to-cryptocurrency-trading-for-beginners (дата звернення: 02.11.2024)

3. Random Forest Algorithm in Machine Learning // GeeksforGeeks. URL: https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/ (дата звернення: 04.11.2024).

4. Zhu H., Chen R., Li R. An Efficient AdaBoost Algorithm with the Multiple Thresholds Classification // Appl. Sci. 2022. Vol. 12, No. 12. Article 5872. URL: https://www.mdpi.com/2076-3417/12/12/5872 (дата звернення: 04.11.2024).

5. Behzat A., Orhan A., Cemil A. An Extra Tree Regression Model for Discharge Coefficient Prediction: Novel, Practical Applications in the Hydraulic Sector and Future Research Directions // Mathematical Problems in Engineering. 2021. URL: https://www.researchgate.net/publication/354760259_An_Extra_Tree_Regression_Model_for_Discharge_Coefficient_Prediction_Novel_Practical_Applications_in_the_Hydraulic_Sector_and_Future_Research_Directions (дата звернення: 06.11.2024)

6. Natekin A., Knoll A. Gradient Boosting Machines, A Tutorial // Frontiers in Neurorobotics. 2013.
URL: https://www.researchgate.net/publication/259653472_Gradient_Boosting_Machines_A_Tutorial (дата звернення: 07.11.2024).

7. Python for Finance and Algorithmic Trading (2nd edition): Machine Learning, Deep Learning, Time Series Analysis, Risk and Portfolio Management for MetaTrader5 Live Trading. – Jupyter, 2023. (дата звернення: 10.11.2024).

8. Understanding the Accuracy Score Metric's Limitations in the Data Science Classification Problems/ Akshay W. // LinkedIn Pulse. 2023. URL: https://www.linkedin.com/pulse/understanding-accuracy-score-metrics-limitations-data-akshay-w

МЕТОДИ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧІ HUMAN ACTIVITY RECOGNITION

Каленіченко Д.О.[1], Данилов В.Я.[2]

Інститут прикладного системного аналізу, КПІ ім. Ігоря Сікорського, Київ, Україна

[1] kalenichenko.denys@lll.kpi.ua, [2] danilov1950@ukr.net [0000-0003-3389-3661]

Носимі пристрої, такі як розумні годинники, збирають величезні обсяги даних з різних датчиків, таких як акселерометри, гіроскопи та магнітометри. Ці дані мають значний потенціал для розпізнавання людської активності, що має вирішальне значення у сферах здоров'я, фітнесу та безпеки. Варіаційні автокодери (VAE) стали потужним інструментом у цьому контексті, вони дозволяють імовірнісно моделювати латентні простори, пропонуючи гнучкі та значущі представлення даних. У цьому документі досліджується застосування VAE для розпізнавання активності на основі немаркованих даних з носимих пристроїв. Результати демонструють потенціал VAE для вирішення проблем, пов'язаних з немаркованими даними датчиків у носимих технологіях.

Ключові слова: глибоке навчання, генеративні мережі, автокодувальник, варіаційний автокодувальник, напівкероване навчання.

У сучасному світі носимі пристрої, такі як розумні годинники, збирають і передають значні обсяги даних, отриманих від різноманітних сенсорів—акселерометрів, гіроскопів, магнетометрів тощо. Ці дані можуть бути використані для розпізнавання активності людини, що має велике значення в галузях здоров'я, фітнесу та безпеки. Проте в більшості випадків ці дані є нерозміченими, оскільки користувачі не здійснюють їх ручну анотацію, а залучення експертів для розмітки є складним та витратним процесом.

У зв'язку з цим виникає потреба в застосуванні методів навчання без нагляду, які можуть виявляти структуру та закономірності в нерозмічених даних. **Варіаційні автокодувальники** (Variational Autoencoders, VAE [1]) є потужним інструментом у цій сфері. Вони поєднують можливості автокодувальників з варіаційними методами байєсівської статистики, що дозволяє моделювати складні розподіли даних та витягувати приховані характеристики.

Класичні методи обробки даних, такі як стандартні автокодувальники або інші алгоритми навчання без нагляду, мають обмеження у здатності моделювати складні ймовірнісні розподіли та генерувати нові дані. Вони часто не враховують невизначеність у латентному просторі та можуть призводити до менш змістовного представлення даних. Варіаційні автокодувальники, навпаки, дозволяють моделювати дані з урахуванням ймовірнісних розподілів, що забезпечує більш гнучке та потужне представлення інформації. Це робить VAE більш придатними для задачі розпізнавання активності на основі нерозмічених даних.

Важливість варіаційних автокодувальників полягає в їх здатності навчатися компактному та змістовному представленню даних у латентному просторі. Це особливо корисно для аналізу даних з носимих пристроїв, де важливо виявити приховані патерни та особливості в сигналах сенсорів без попередньої розмітки. VAE дозволяють генерувати нові зразки даних, що схожі на вихідні, а також виявляти аномалії та кластеризувати дані за схожістю.

Актуальність теми: з огляду на зростання обсягів нерозмічених даних з носимих пристроїв, існує потреба в ефективних методах навчання без нагляду для розпізнавання активності людини. Варіаційні автокодувальники пропонують перспективне рішення цієї проблеми.

Мета дослідження: вивчити можливості варіаційних автокодувальників у розпізнаванні активності людини на основі нерозмічених даних, зібраних носимими пристроями, та оцінити ефективність запропонованої системи.

Принцип роботи варіаційних автокодувальників

VAE поєднує властивості класичного автокодувальника з принципами варіаційного байєсівського виведення. Структурно він складається з двох основних компонентів: енкодера та декодера.

- **Енкодер** $q_\phi(z \mid x)$: приймає вхідні дані x та відображає їх у параметри розподілу латентних змінних z, зазвичай це середнє μ та дисперсія σ^2 нормального розподілу.
- **Декодер** ($p_\theta(z \mid x)$): приймає латентні змінні z та відновлює початкові дані x.

Математичне обґрунтування

Мета VAE – максимізувати правдоподібність даних $p_\theta(x)$. Оскільки обчислити її безпосередньо складно, використовують нижню варіаційну межу (Evidence Lower BOund, ELBO):

$$\log p_\theta(x) \geq \mathbb{L}(\theta, \phi; x) = -D_{KL}(q_\phi(z|x)||p_\theta(z)) + \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)],$$

де:

- $D_{KL}(q_\phi(z|x)||p_\theta(z))$ – дивергенція Кульбака-Лейблера між апроксимацією постеріорного розподілу та апріорним розподілом латентних змінних.
- $\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$ – очікувана реконструктивна похибка, яка вимірює, наскільки добре модель відновлює вхідні дані.

Репараметризація

Для забезпечення диференційовності функції втрат по параметрах ϕ та θ, використовується трюк репараметризації. Замість вибірки $z \sim q_\phi(z \mid x)$, вводять випадкову змінну $\epsilon \sim N(0, I)$ та обчислюють:

$$z = \mu_\phi(x) + \sigma_\sigma(x) \odot \epsilon,$$

де:

- $\mu_\phi(x),\ \sigma_\sigma(x)$ – виходи енкодера, що залежать від вхідних даних x.
- $\odot$ – покомпонентне множення.
- ϵ – шум, згенерований з стандартного нормального розподілу.

Компоненти функції втрат

Функція втрат VAE складається з двох частин:

1. **Реконструктивна втрата**: вимірює різницю між вхідними даними x та їх реконструкцією $\hat{x}$

$$\mathbb{L}_{recon} = \mathbb{E}_{q_\phi(z|x)}[-\log p_\theta(x|z)].$$

2. **Втрата регуляризації**: спрямована на мінімізацію відмінності між апроксимацією постеріорного розподілу та апріорним розподілом:

$$\mathbb{L}_{KL} = D_{KL}(q_\phi(z|x)||p_\theta(z)).$$

Загальна функція втрат:

$$\mathbb{L} = \mathbb{L}_{recon} + \mathbb{L}_{KL}$$

Під час виконання роботи ми зіштовхнулися з труднощами при навчанні варіаційного автокодувальника, яка зупиняла навчання через дуже великі значення регуляризатора. Ця проблема була також зустрічалася в [2] і вони запропонували модифікувати функцію втрат:

$$\mathbb{L} = \mathbb{L}_{recon} + \alpha \times \mathbb{L}_{KL},$$

де α – змінна, яка збільшується протягом навчання від 0 в межах [0, 1].

Набір даних

Для цього експерименту був обраний датасет MotionSense [3]. Цей датасет включає у себе часові ряди даних, згенеровані датчиками акселерометра та гіроскопа (положення, гравітація, користувацьке прискорення та швидкість обертання). Інформація збиралася з використанням iPhone 6s, розташованого в передньому кишені учасника за допомогою SensingKit, який збирає дані з фреймворку Core Motion на пристроях iOS. Всі дані були зібрані з частотою 50 Гц. Всього 24 учасники різної статі, віку, ваги та зросту виконали 6 видів активностей у 15 випробуваннях в однаковому середовищі та умовах: спуск по сходах, підйом по сходах, ходьба, біг, сидіння та стояння. За допомогою цього датасету ми сподіваємося знайти особисті відмітки атрибутів у часових рядах даних з датчиків, тобто патерни, які є характерними для певних атрибутів та можуть бути використані для визначення статі чи особистості суб'єктів даних, крім їхніх активностей.

Дані, які зібрані в цьому датасеті, мають вигляд схожий на той, який зображений на рисунку 1.

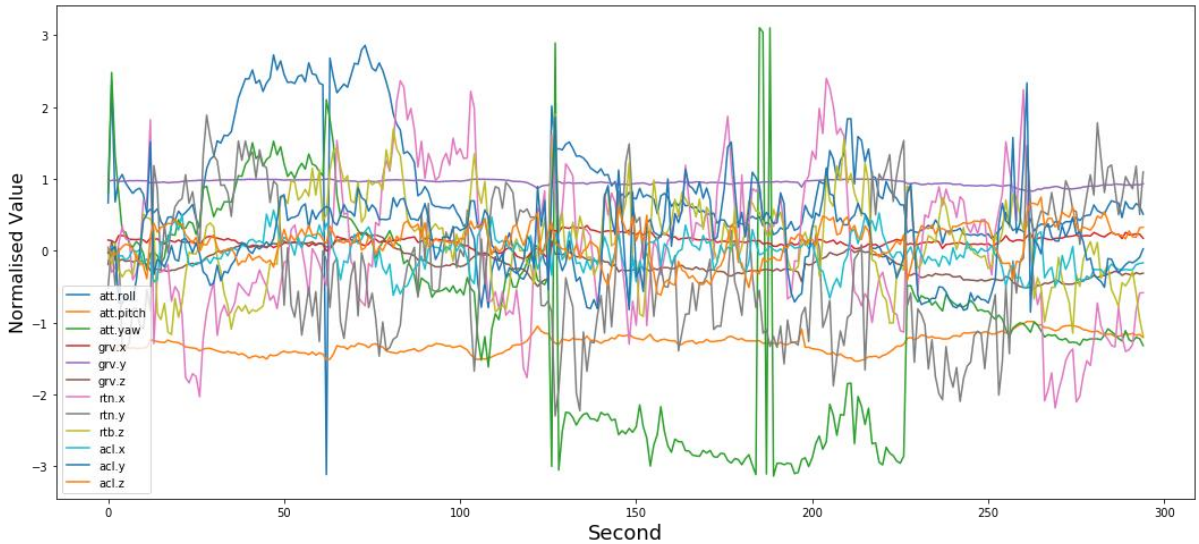


Рисунок 1. Часові ряди, які відповідають за «Ходьбу» об’єкта №3

Архітектура варіаційного автокодувальника буде складатися з рекурентних шарів, а саме LSTM [4]. Ми вибрали архітектуру, базуючись на дослідженні [5], в якому для задачі біометричної верифікації ця мережа показала низький рівень похибки (EER= 0,111).

Архітектура автокодувальника представлена на рисунках 2 і 3.

Layer (type)	Output Shape	Param #
lstm_12 (LSTM)	(32, 100, 128)	69,120
lstm_13 (LSTM)	(32, 64)	49,408
dense_9 (Dense)	(32, 32)	2,080
dense_10 (Dense)	(32, 32)	2,080

Total params: 122,688 (479.25 KB)
Trainable params: 122,688 (479.25 KB)
Non-trainable params: 0 (0.00 B)

Рисунок 2. Архітектура кодувальника

Layer (type)	Output Shape	Param #
repeat_vector_3 (RepeatVector)	(32, 100, 32)	0
lstm_14 (LSTM)	(32, 100, 64)	24,832
lstm_15 (LSTM)	(32, 100, 64)	33,024
time_distributed_3 (TimeDistributed)	(32, 100, 6)	390

Total params: 58,246 (227.52 KB)
Trainable params: 58,246 (227.52 KB)
Non-trainable params: 0 (0.00 B)

Рисунок 3. Архітектура декодувальника

Навчання ми проводили за допомогою алгоритма оптимізатора Adam, використовуючи навчання батчами по 32 одиниці даних. Навчалася модель протягом 100 епох.

Для використання подібної мережі в задачі класифікації ми побудували 4 мережі з цією архітектурою, кожна з яких відповідатиме за розпізнавання визначеної активності. Для цього експерименту ми виділили 4 активності: спуск сходами, підйом сходами ходьба і легкий біг.

Результати дослідження

1. **Обробка вхідних даних**: Кожна модель незалежно пропускає вхідні дані через свій власний автоенкодер.

2. **Розрахунок помилки реконструкції**: Для кожної моделі обчислюється середньоквадратична помилка (MSE) між початковими вхідними даними та їх відновленими версіями з автоенкодера. Ця помилка відображає точність, з якою модель відтворює дані.

3. **Вибір моделі з найменшою помилкою**: Зі всіх отриманих помилок вибирається найменша. Модель з найменшою помилкою реконструкції вважається тією, що найкраще відповідає вхідній активності, і, таким чином, ідентифікує її.

Таким чином, алгоритм використовує набір спеціалізованих автоенкодерів і аналізує їх помилки реконструкції для точного визначення типу активності на основі вхідних даних.

Після навчання отримуємо такі результати, які наведені в таблиці 1 та на рисунку 4:

Таблиця 1. Результати навчання

	precision	recall	f1-score	support
Downstairs	0.5	0.77	0.6	2242
Upstairs	0.48	0.79	0.6	2691
Walking	0.88	0.54	0.67	5556
Jogling	0.87	0.55	0.68	2195
accuracy			0.64	12684
macro avg	0.68	0.66	0.64	12684
weighted avg	0.73	0.64	0.64	12684

Висновки

У цьому дослідженні було застосовано варіаційні автокодувальники для розпізнавання активності людини на основі нерозмічених даних, зібраних носимими пристроями. Результати експериментів показали, що запропонований підхід має потенціал, проте потребує подальшого вдосконалення для досягнення більш високої точності та надійності.

Отримані результати класифікації свідчать про наступне:

- **Загальна точність моделі** становить 64%, що вказує на здатність моделі розпізнавати активності краще, ніж випадкове вгадування, але недостатньо для практичного застосування.
- **Класи «Downstairs» та «Upstairs»** продемонстрували помірну F1-міру в 60% з високим показником повноти (77% та 79% відповідно) та нижчою точністю (50% та 48%). Це означає, що модель часто правильно ідентифікує ці активності, але також помилково включає до них інші.

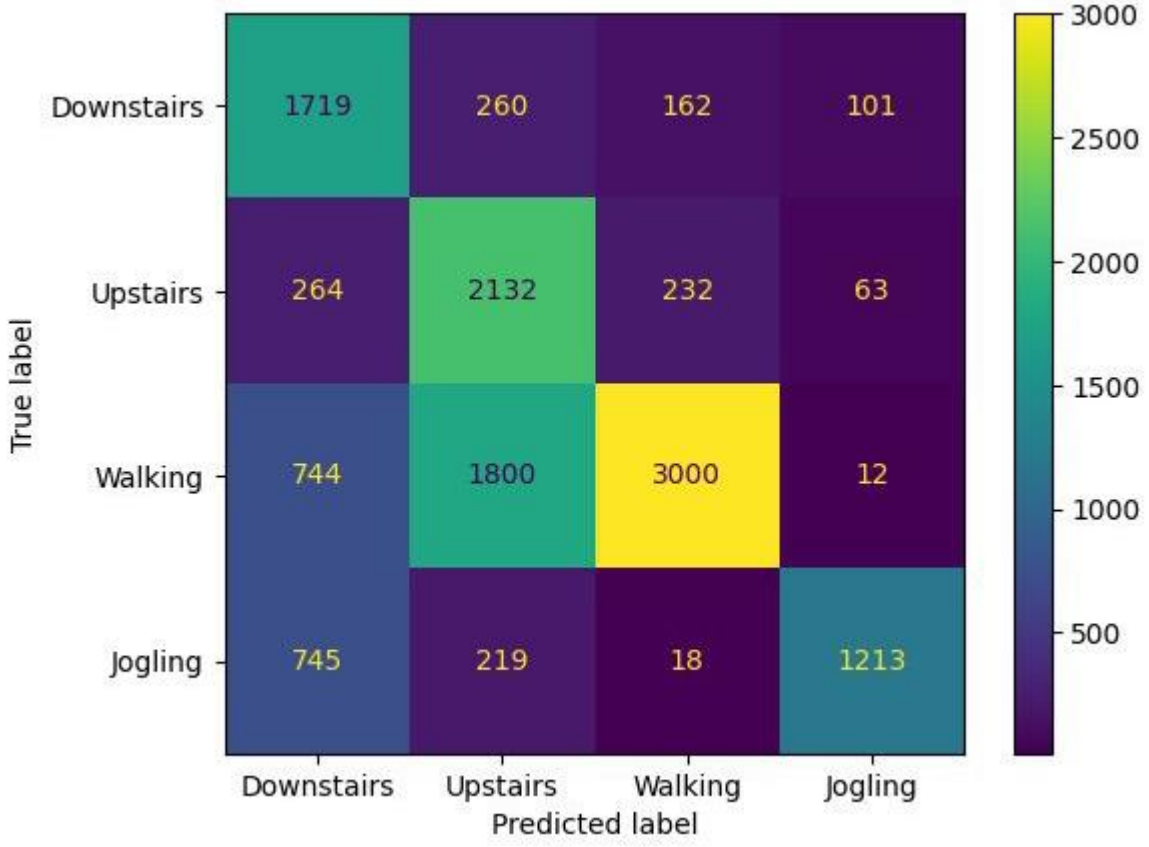


Рисунок 4. Матриця помилок для кожної з активностей

- **Класи «Walking» та «Jogging»** мають високу точність (88% та 87%), але нижчу повноту (54% та 55%), що свідчить про те, що модель впевнено класифікує ці активності, але пропускає значну кількість їх випадків.
- **F1-міра** для всіх класів коливається між 60% та 68%, що свідчить про необхідність покращення балансу між точністю та повнотою.

Перспективи подальших досліджень

Для підвищення ефективності та надійності моделі рекомендується:

1. **Розширення та збалансування датасету**: Зібрати більший обсяг даних з рівномірним представленням кожної активності, що допоможе моделі краще навчитися розпізнавати патерни.
2. **Оптимізація архітектури варіаційного автокодувальника**: Експериментувати з різними гіперпараметрами, розмірами латентного простору та структурами нейронних мереж для покращення здатності моделі вловлювати складні залежності в даних.
3. **Поєднання з іншими методами навчання**: Розглянути можливість використання напівконтрольованого навчання або кластеризації для покращення розпізнавання активностей.
4. **Аналіз помилок моделі**: Детально вивчити випадки неправильної класифікації для виявлення причин помилок та вдосконалення моделі.
5. **Тестування на різних наборах даних**: Перевірити модель на інших датасетах для оцінки її стійкості та узагальнюючої здатності.
6. **Персоналізація моделі**: Розробити методи адаптації моделі до індивідуальних особливостей користувачів, що може покращити точність для конкретних осіб.

Висновок

Запропонований підхід з використанням варіаційних автокодувальників демонструє перспективи в розпізнаванні активності людини на основі нерозмічених даних з носимих пристроїв. Отримані результати вказують на те, що модель здатна виявляти певні патерни, проте потребує подальшого вдосконалення. Реалізація запропонованих напрямків майбутніх досліджень може значно підвищити ефективність системи та сприяти її практичному застосуванню в галузях охорони здоров'я, фітнесу та безпеки.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Kingma, D. P., & Welling, M. (2019). An introduction to variational autoencoders. Foundations and Trends® in Machine Learning, 12(4), 307-392.

2. Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A. M., Jozefowicz, R., & Bengio, S. (2015). Generating sentences from a continuous space. arXiv preprint arXiv:1511.06349.

3. MotionSense dataset. Available at https://github.com/mmalekzadeh/motion-sense

4. Vennerød, C. B., Kjærran, A., & Bugge, E. S. (2021). Long short-term memory RNN. arXiv preprint arXiv:2105.06756.

5. Gozhyj, A. (2020). Comparative Analysis of using Recurrent Autoencoders for User Biometric Verification with Wearable Accelerometer.

ПРОГНОЗУВАННЯ ТА ОЦІНЮВАННЯ ТРАНСФЕРНИХ РИЗИКІВ

Кармазін В.А.[1], Левенчук Л.Б.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] vova.karmazin2015@gmail.com,
[2] levenchuk.liudmyla@lll.kpi.ua [0000-0002-8600-0890]

В умовах зростаючої нестабільності світових фінансових ринків особливої актуальності набуває проблема ефективного прогнозування трансферних ризиків. У роботі проведено комплексний аналіз макроекономічних показників НБУ та розроблено систему прогнозування на основі методів машинного навчання. Порівняльний аналіз ефективності моделей показав найкращі результати для Ridge Regression з $R^2 = 0.626$. Визначено ключові фактори впливу на трансферні ризики та запропоновано підходи до їх оцінювання.

Ключові слова: трансферні ризики, машинне навчання, прогнозування, макроекономічні показники

1. ВСТУП

Трансферні ризики в банківській системі набувають особливої актуальності в умовах глобалізації фінансових ринків та зростаючої волатильності міжнародних фінансових потоків. Здійснення міжнародних платежів та транскордонних операцій супроводжується невизначеністю щодо майбутніх змін валютних курсів, процентних ставок та інших макроекономічних показників. Це призводить до необхідності розробки ефективних інструментів прогнозування та оцінювання трансферних ризиків.

Своєчасна та точна оцінка потенційних ризиків дозволяє банкам оптимізувати свою діяльність, мінімізувати можливі втрати та забезпечити стабільність операцій. Для цього необхідно враховувати широкий спектр факторів впливу, проводити комплексний аналіз макроекономічних показників та застосовувати сучасні методи прогнозування [1, 2].

Використання методів машинного навчання для прогнозування трансферних ризиків дозволяє автоматизувати процес аналізу даних, виявляти приховані закономірності та підвищувати точність прогнозів. Разом з тим, кожен метод має свої особливості та обмеження, що зумовлює необхідність їх порівняльного аналізу та вибору найбільш ефективних підходів для конкретних умов застосування [3].

Дане дослідження спрямоване на розробку комплексної системи прогнозування трансферних ризиків, що поєднує аналіз макроекономічних показників НБУ з сучасними методами машинного навчання для забезпечення більш точного та надійного оцінювання потенційних ризиків у банківській системі України.

2. ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

Метою роботи є розробка та впровадження системи прогнозування трансферних ризиків на основі методів машинного навчання для підвищення ефективності управління ризиками в банківській системі України.

Дослідження передбачає проведення комплексного аналізу макроекономічних показників НБУ та підготовку даних для моделювання. На основі підготовлених даних необхідно розробити та порівняти ефективність різних моделей машинного навчання для прогнозування трансферних ризиків, а також визначити ключові фактори впливу на їх формування. Важливим етапом є створення системи оцінювання та прогнозування ризиків на основі найбільш ефективної моделі та розробка рекомендацій щодо її практичного застосування в банківському секторі.

Основними вимогами до розроблюваної системи є забезпечення точності прогнозування не нижче $R^2 > 0,6$, можливість врахування сезонних коливань та автоматизована обробка вхідних даних.

3. ОСНОВНІ МЕТОДИ

Трансферні ризики в банківській системі обумовлені складною взаємодією численних макроекономічних факторів, включаючи валютні курси, процентні ставки, платіжний баланс та зовнішній борг. Наявність великої кількості взаємопов'язаних показників робить задачу прогнозування складною та вимагає застосування комплексного підходу до аналізу даних [4].

Для прогнозування трансферних ризиків було використано кілька методів машинного навчання. Ridge Regression (гребенева регресія) застосовує L2-регуляризацію для зменшення впливу мультиколінеарності та запобігання перенавчанню моделі. Цей метод особливо ефективний при роботі з макроекономічними показниками, які часто мають високий рівень кореляції [5].

Gradient Boosting та Random Forest представляють ансамблеві методи, що дозволяють врахувати нелінійні залежності між показниками. Gradient Boosting послідовно будує прості моделі, кожна з яких коригує помилки попередніх, тоді як Random Forest використовує принцип беггінгу для побудови множини незалежних дерев рішень [6].

Для врахування часової структури даних було використано LSTM (Long Short-Term Memory) – різновид рекурентних нейронних мереж, спеціально розроблений для аналізу часових рядів. XGBoost, як оптимізована реалізація градієнтного бустингу, забезпечує високу швидкодію та точність при обробці великих наборів даних [7].

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

4.1. Набір даних

Першим етапом дослідження став аналіз та підготовка даних НБУ. Для забезпечення якості прогнозування було створено три основні набори даних з різними часовими періодами (рис. 1).

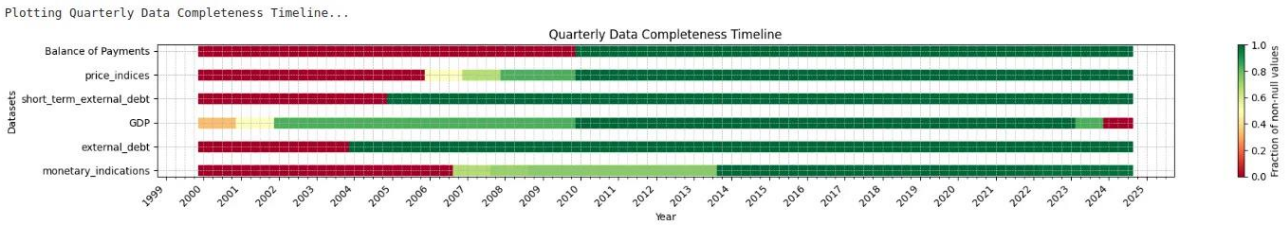


Рисунок 1. Часова діаграма повноти квартальних даних за наборами

На основі аналізу повноти даних було сформовано набори 2010–2023, 2014–2023 та повний датасет. Кореляційний аналіз виявив значні взаємозв'язки між показниками (рис. 2).

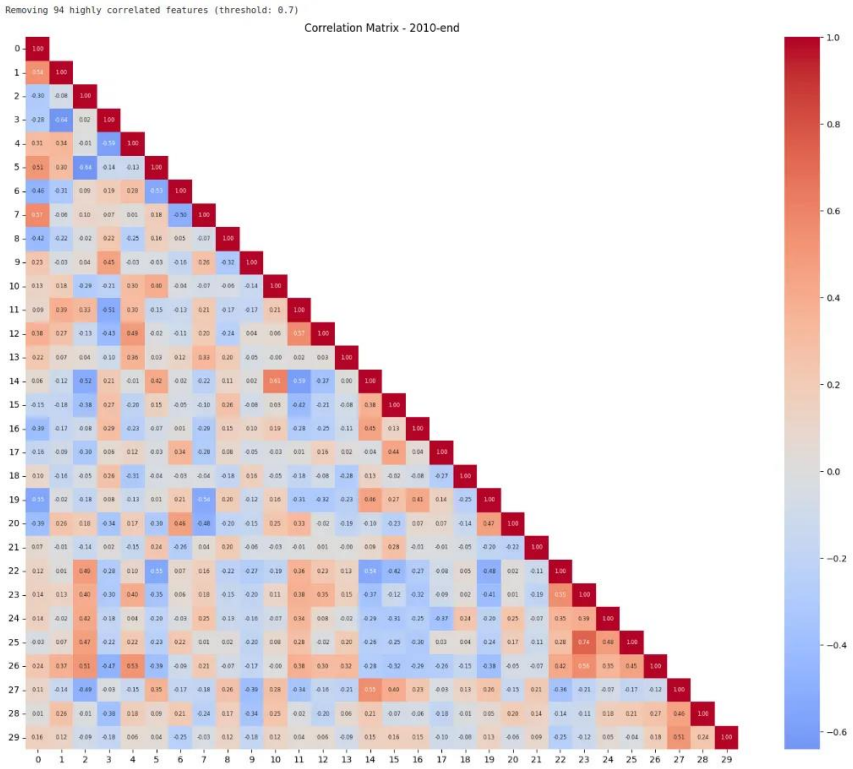


Рисунок 2. Кореляційна матриця для набору даних 2010-end

Для оптимізації наборів даних було проведено видалення сильно корельованих змінних (табл. 1).

Таблиця 1. Результати видалення корельованих змінних

Набір даних	Початкова кількість змінних	Видалено змінних	Залишилось змінних	% скорочення
2010-2023	124	95	29	76.6%
Повний датасет	124	83	41	66.9%
2014-2023	124	96	28	77.4%
2010-end	124	94	30	75.8%

Для подальшого зменшення розмірності даних було застосовано метод головних компонент (PCA). Аналіз власних значень та кумулятивної поясненої дисперсії показано на рис. 3.

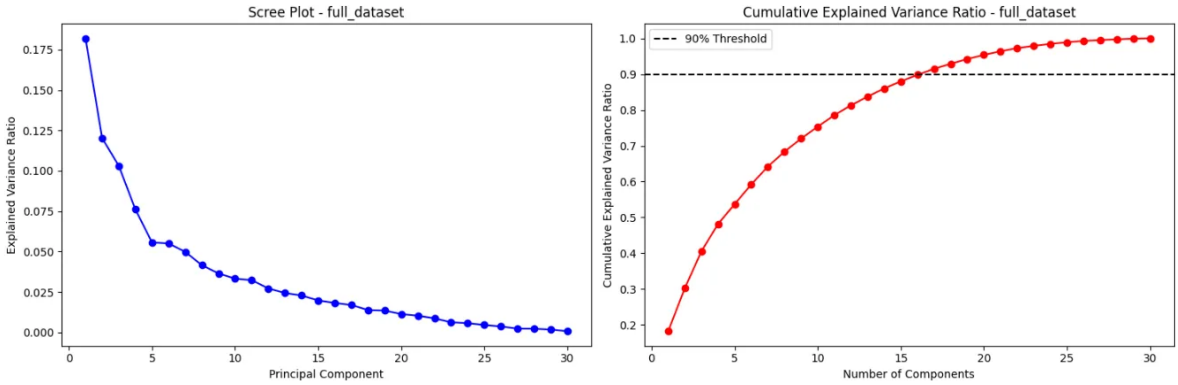


Рисунок 3. Аналіз головних компонент для повного набору даних

Застосування методу головних компонент дозволило значно зменшити розмірність даних, зберігаючи при цьому більше 90% інформації для кожного з наборів даних, що підтверджується результатами (табл. 2).

Таблиця 2. Результати відбору значущих компонент

Набір даних	Початкова кількість ознак	Кількість головних компонент	% скорочення розмірності	Кумулятивна дисперсія
2010-2023	29	16	44.8%	90.2%
2010-end	30	17	43.3%	90.5%
2014-2023	28	15	46.4%	90.3%
Повний датасет	41	18	56.1%	90.1%

Такий підхід до підготовки даних забезпечив оптимальну базу для подальшого навчання предиктивних моделей, зберігши при цьому найбільш значущу інформацію та усунувши надлишковість у даних.

4.2. Тренування моделей

Для оцінки ефективності розробленої системи прогнозування трансферних ризиків було проведено комплексний аналіз роботи різних моделей машинного навчання на даних НБУ за період 2010–2024 років. Основною метою було порівняння точності прогнозування та визначення найбільш ефективного методу для практичного застосування.

Порівняльний аналіз моделей показав суттєві відмінності в їх ефективності. На періоді 2019–2021 років Ridge Regression та Random Forest продемонстрували найбільш стабільні результати, тоді як LSTM показала значні коливання в прогнозах, що видно на рис. 4.

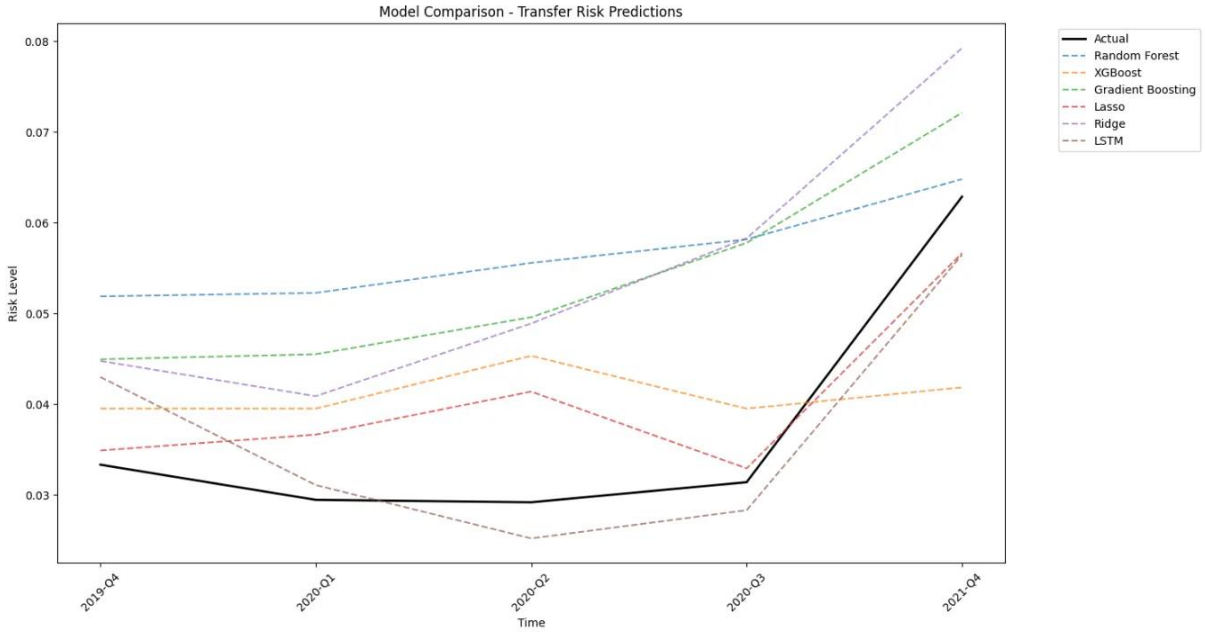


Рисунок 4. Порівняння точності прогнозування моделей за період 2019–2021

Кількісна оцінка ефективності моделей за основними метриками якості прогнозування показала наступні результати (табл. 3).

Таблиця 3. Порівняння метрик ефективності моделей

Модель	MSE	RMSE	MAE	R^2
Ridge Regression	7.60e-05	0.0087	0.0058	0.626
Gradient Boosting	1.27e-04	0.0113	0.0095	0.375
Random Forest	1.38e-04	0.0117	0.0113	0.321
XGBoost	1.49e-04	0.0122	0.0091	0.267
LSTM	4.66e-04	0.0216	0.0178	-1.294
Lasso	1.02e-03	0.0320	0.0191	-4.029

Ridge Regression показала найкращі результати за всіма метриками, зокрема найвищий коефіцієнт детермінації R^2 (0,626) та найнижчу середньоквадратичну помилку RMSE (0,0087). Ансамблеві методи (Gradient Boosting, Random Forest, XGBoost) продемонстрували помірну ефективність з R^2 в діапазоні 0,3–0,4.

5. ВИСНОВКИ

У результаті проведеного дослідження була розроблена та протестована система прогнозування трансферних ризиків на основі методів машинного навчання. В процесі роботи створено три оптимізовані набори даних (2010–2023, 2014–2023 та повний датасет), що забезпечують репрезентативну базу для аналізу трансферних ризиків. Застосування методу головних компонент дозволило зменшити розмірність даних на 44–56% при збереженні понад 90% інформативності.

Порівняльний аналіз шести моделей машинного навчання показав найвищу ефективність Ridge Regression з коефіцієнтом детермінації $R^2 = 0,626$ та найнижчою середньоквадратичною помилкою RMSE = 0,0087. Ансамблеві методи, такі як Gradient Boosting та Random Forest, продемонстрували помірну ефективність з R^2 в діапазоні 0,3–0,4, в той час як LSTM та Lasso виявились найменш ефективними для даної задачі.

За результатами аналізу важливості параметрів виявлено ключові фактори впливу на трансферні ризики. Найбільш значущими виявились монетарні показники з лаговими значеннями, що мають важливість близько 0,6 за оцінкою XGBoost. Показники платіжного балансу показали важливість близько 0,2, також суттєвий вплив мають короткострокові боргові зобов'язання та цінові індекси.

Створена система забезпечує автоматизовану оцінку та прогнозування трансферних ризиків з точністю, достатньою для практичного застосування в банківському секторі. Подальше вдосконалення системи може бути спрямоване на розробку гібридних моделей для підвищення точності прогнозування, впровадження механізмів раннього попередження про зростання ризиків та розширення часового горизонту прогнозування.

Отримані результати підтверджують ефективність використання методів машинного навчання для прогнозування трансферних ризиків та створюють основу для подальшого розвитку систем ризик-менеджменту в банківській сфері України. Практичне впровадження розробленої системи дозволить підвищити якість управління ризиками та забезпечити стабільність банківських операцій в умовах мінливого фінансового середовища.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Кузнєцова Н.В., Бідюк П.І. Системний підхід до менеджменту фінансових ризиків. Системні дослідження та інформаційні технології. 2018. № 2. С. 124–140.
2. Новоселецький О.М., Якубець О.В. Моделювання кредитних ризиків комерційного банку в умовах асиметрії інформації. Наукові записки НаУКМА. Економічні науки. 2020. № 5. С. 71-77.
3. Камінський А.Б. Моделювання фінансових ризиків : монографія. Київ : ВПЦ "Київський університет", 2019. 304 с.
4. Дробязко А.О., Любіч О.О. Банківська система України в умовах глобалізації фінансових ринків. Економіка України. 2021. №3. С. 17-28.
5. Friedman J., Hastie T., Tibshirani R. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, 2022. 745 p.
6. Кузнєцова Н.В., Бідюк П.І. Порівняльний аналіз характеристик моделей оцінювання ризиків кредитування. Наукові вісті НТУУ "КПІ". 2019. №5. С. 24-34.
7. Hastie T., Tibshirani R., Wainwright M. Statistical Learning with Sparsity: The Lasso and Generalizations. Chapman and Hall/CRC, 2023. 367 p.

ПІДХІД ДО ПРИЙНЯТТЯ ТОРГОВИХ РІШЕНЬ НА ОСНОВІ ОПТИМІЗАЦІЇ ІНВЕСТИЦІЙНОГО ПОРТФЕЛЯ

Кіріянов І.О.[1], Савастьянов В.В.

Національний технічний університет України
“Київський політехнічний інститут ім. Ігоря Сікорського”, Київ, Україна

[1] kiriyanov03@gmail.com

Аналіз та оптимізація портфеля є важливими аспектами сучасної теорії портфеля та інвестиційного менеджменту. Портфельний аналіз оцінює характеристики ризику та дохідності набору активів, тоді як оптимізація портфеля має на меті максимізувати дохідність при заданому рівні ризику або мінімізувати ризик. Сучасна теорія портфеля Гаррі Марковіца формує основу для оптимізації портфеля, наголошуючи на диверсифікації та поєднанні активів з різними характеристиками ризику та дохідності. Метою даної роботи є розробка програми для отримання інформації про аналіз та оптимізацію інвестиційного портфелю, а також проведення порівняльного аналізу різних методів портфельної оптимізації.

Ключові слова: портфельная оптимізація, матриця кореляції , Cvar, HRP, коефіцієнт Шарпа.

1. ВСТУП

У сучасному світі інвестиції є невід'ємною частиною економічного розвитку як окремих осіб, так і підприємств. Оптимізація інвестиційного портфеля є ключовим завданням для забезпечення максимальної віддачі при мінімальному ризику. Класична теорія портфельної оптимізації, розроблена Гаррі Марковітцем, стала основою для багатьох інвестиційних стратегій. Однак, у реальному світі інвестори часто стикаються з невизначеністю та нечіткістю в оцінках ризиків та доходів. Нечітка логіка та методи штучного інтелекту дозволяють обробляти невизначені дані і створювати більш гнучкі моделі.

Управління інвестиційним портфелем це наука про створення та управління диверсифікованим портфелем активів для досягнення конкретних фінансових цілей, забезпечуючи баланс між ризиком і прибутковістю. Вивчення моделі оптимального портфеля ринку криптовалют стає особливо актуальним в умовах соціальних та поведінкових змін [1].

2. ОГЛЯД ВХІДНИХ ДАНИХ ТА АНАЛІЗ ІНВЕСТИЦІЙНОГО ПОРТФЕЛЮ

В значенні параметрів використаємо дані з сайту binance.com [2]. Серед криптовалют для розляду було обрано наступні: Bitcoin (BTC/USDT), Ethereum (ETH/USDT), Binance coin (BNB/USDT), Cardano (ADA/USDT), Soland (SOL/USDT). За часовий проміжок буде взято останні 5 років. Тобто дані з 17.11.2019 року до 15.11.2024. У цьому датасеті основні атрибути наступні:

1) відомості про вартість криптовалюти на початку календарного дня(Open);
2) на кінець календарного дня (Close);
3) найвища вартість впродовж дня (High);
4) найнижча вартість впродовж дня (Low).

Для аналізу інвестиційного портфеля будемо використовувати наступні метрики :

- бета коефіцієнт;
- модель оцінки капітальних активів (CAPM);
- коефіцієнт Шарпа;
- коефіцієнт Трейнора;
- M2 (модель Модльяні-Міллєра);
- альфа Дженсена;
- показник співвідношення інформації;
- коефіцієнт Сортіно.

3. ОПТИМІЗАЦІЯ ІНВЕСТИЦІЙНОГО ПОРТФЕЛЮ

Для оптимізації інвестиційного портфеля будемо використовувати наступні методи оптимізації інвестиійного портфелю як коефіцієнта Шарпа, ієрархічного паритету ризиків (HPR), умовної вартості під ризиком(CVaR), мінімальної дисперсії (Min_Var) та рівних ваг (Equal_Weight).

Коефіцієнт Шарпа

Коефіцієнт Шарпа (Sharp Ratio) – показник, розроблений американським ученим-економістом, лауреатом Нобелівської премії з економіки Вільямом Ф. Шарпом. Він став важливим доповненням (деякою мірою – значущою частиною) портфельної теорії Марковіца. Коефіцієнт Шарпа показує, яку дохідність приносять інвестиції на кожну одиницю ризику. Таке розуміння цього показника дає змогу легко порівнювати різні варіанти інвестування, портфелі тощо.

Оптимізація за допомогою ієрарархічного паритету ризиків

Ієрархічний паритет ризиків (HRP) – це метод розподілу портфеля у фінансах, який має на меті оптимізувати співвідношення ризику та дохідності портфеля. Він базується на ідеї диверсифікації та паритету ризиків, що означає, що кожен актив у портфелі повинен вносити рівний внесок у загальний ризик портфеля. У HRP активи групуються у кластери на основі їхньої кореляції один з одним. Потім алгоритм обчислює внесок кожного кластера в ризик і розподіляє капітал між кластерами таким чином, щоб вирівняти їхні внески в ризик. Цей підхід спрямований на створення більш збалансованого та диверсифікованого портфеля, що потенційно знижує загальний ризик портфеля, зберігаючи або покращуючи його очікувану дохідність [3].

Оптимізація за допомогою умовної вартості під ризиком

Ефективна умовна вартість під ризиком (CVaR) – це концепція у фінансах, яка стосується вимірювання ризику в інвестиційному портфелі. CVaR – це міра ризику, яка кількісно оцінює потенційні втрати в хвості розподілу можливих прибутків. Він дає більш повне уявлення про ризик порівняно з традиційними показниками ризику, такими як стандартне відхилення або вартість під ризиком (VaR). Ефективний CVaR, також відомий як оптимізація середнього значення CVaR, має на меті побудувати інвестиційний портфель, який максимізує очікувану прибутковість при мінімізації умовної вартості під ризиком. Цей підхід враховує не тільки очікувану дохідність і волатильність портфеля, але й потенційну серйозність втрат, що перевищують певний поріг. Оптимізуючи Ефективний CVaR, інвестори прагнуть створити портфелі, які є більш стійкими до екстремальних ринкових подій та хвостових ризиків. Це може бути особливо важливо для інвесторів, які не схильні до ризику, або для тих, хто працює на нестабільних ринках, де потенціал великих збитків викликає значне занепокоєння.

Загалом, якщо інвестиція демонструє стабільність протягом тривалого часу, то вартість під ризиком може бути достатньою для управління ризиками в портфелі, що містить цю інвестицію. Однак, чим менш стабільною є інвестиція, тим більша ймовірність того, що VaR не дасть повної картини ризиків, оскільки вона байдужа до всього, що перевищує її власний поріг. Умовна вартість під ризиком (CVaR) намагається усунути недоліки моделі VaR, яка є статистичним методом, що використовується для вимірювання рівня фінансового ризику в рамках фірми або інвестиційного портфеля протягом певного періоду часу. У той час як VaR відображає найгірші втрати, пов'язані з ймовірністю та часовим горизонтом, CVaR - це очікувані втрати, якщо цей поріг найгіршого випадку буде перетнуто. Іншими словами, CVaR кількісно виражає очікувані збитки, які виникають за межами точки розриву VaR [4].

Оптимізація портфеля з мінімальною дисперсією

Портфель з мінімальною дисперсією (MVP) зосереджується на коваріації. Він фокусується виключно на мінімізації загальної дисперсії портфеля, покладаючись на коваріаційну матрицю дохідності активів. Такий підхід, як правило, демонструє кращі результати поза вибіркою порівняно з портфелями середньої дисперсії, які також враховують очікувану дохідність. Незважаючи на свої переваги, MVP все ще може бути чутливим до помилок оцінки в коваріаційній матриці, що призводить до нестабільності ваг портфеля, які коливаються з часом. Дисперсія портфеля обчислюється як сукупна вага індивідуальних дисперсій, пов'язаних з кожним базовим цінним папером. Кожна дисперсія додатково коригується за допомогою коваріації. Зрештою, інвестори прагнуть, щоб загальна дисперсія портфеля була нижчою за середньозважену індивідуальну дисперсію [5].

Оптимізація портфеля за допомогою рівних ваг. Рівна вага – це різновид пропорційного методу вимірювання, який надає однакову важливість кожній криптовалюті в портфелі. Індекс рівної ваги також відомий як незважений індекс [6].

4. РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Результат роботи програми аналізу інвестиційного портфелю

Далі зображені відповідні результати аналізу портфеля обраних криптовалют (рис. 1–3).

	Volatility	Beta	CAPM	Sharpe ratio	Treynor ratio	M2	Jensen's alpha	Sortino ratio	Information ratio
BTC/USDT	0.605082	0.105704	0.338406	3.218761	18.425238	0.000000	1.741745	5.007578	0.000000
ETH/USDT	0.820437	0.228917	0.709555	2.537099	9.092968	0.874755	1.635690	3.827278	0.012289
BNB/USDT	1.273217	0.341180	1.047721	2.814324	10.502513	5.592267	2.918758	5.683575	0.150092
ADA/USDT	1.125496	0.234617	0.726727	2.616134	12.550021	3.529272	2.487505	4.881178	0.091474
SOL/USDT	1.525126	1.000000	3.032256	1.975086	3.012256	5.644859	1.064641	3.826708	0.097696

Рисунок 1. Вихідні дані аналізу оцінки портфелю метриками

Рисунок 2. Вихідні дані аналізу оцінки портфелю метриками

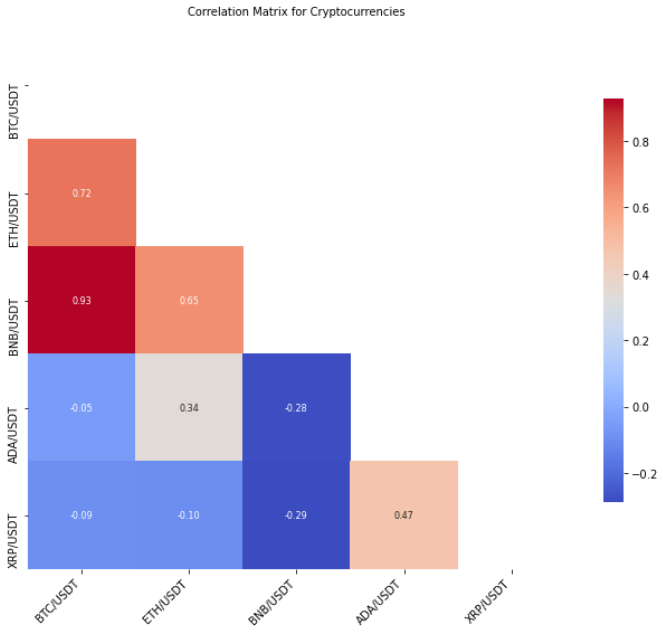


Рисунок 3. Матриця кореляції

Результати роботи програми оптимізації інвестиційного портфелю

Далі зображені відповідні результати оптимізації портфеля обраними методами та їх порівняння (рис. 4–9).

```
Expected annual return: 522.6%
Annual volatility: 81.1%
Sharpe Ratio: 6.42
Not allocated budget: 63.59 out of 5000.00
                Price  Optimal Weights  Optimal Volume   Total pay
BTC/USDT  58746.57000          0.00892            0.0     0.00000
ETH/USDT   1840.46000          0.00000            0.0     0.00000
BNB/USDT    311.29620          0.00000            0.0     0.00000
ADA/USDT      1.21189          0.38865         1603.0  1942.65967
SOL/USDT     19.06850          0.60244          157.0  2993.75450
```

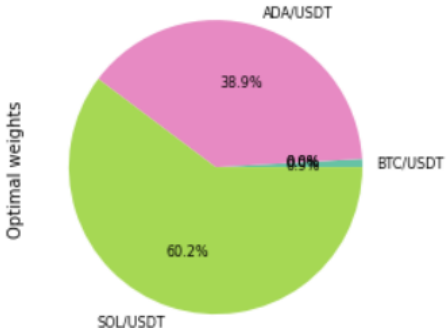


Рисунок 4. Результат оптимізації коефіцієнтом Шарпа

```
Expected annual return: 250.6%
Annual volatility: 69.1%
Sharpe Ratio: 3.60
Not allocated budget: 1153.74 out of 5000.00
                Price  Optimal Weights  Optimal Volume   Total pay
BTC/USDT  58746.57000         0.377760            0.0     0.00000
ETH/USDT   1840.46000         0.205472            1.0  1840.46000
BNB/USDT    311.29620         0.138714            2.0   622.59240
ADA/USDT      1.21189         0.181379          748.0   906.49372
SOL/USDT     19.06850         0.096675           25.0   476.71250
```

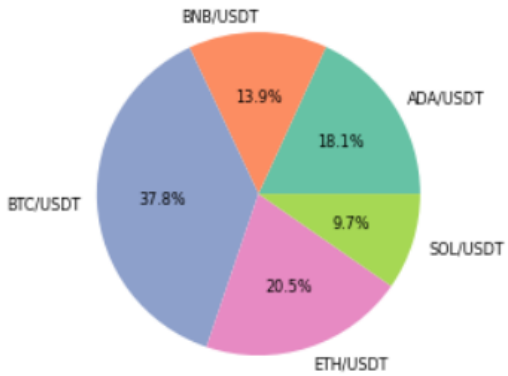


Рисунок 5. Результат оптимізації за допомогою ієрархічного паритету ризиків (HRP)

Not allocated budget: 139.87 *out of* 5000.00

	Price	Optimal Weights	Optimal Volume	Total pay
BTC/USDT	58746.57000	0.0000	0.0	0.000
ETH/USDT	1840.46000	0.0000	0.0	0.000
BNB/USDT	311.29620	0.9618	15.0	4669.443
ADA/USDT	1.21189	0.0000	0.0	0.000
SOL/USDT	19.06850	0.0382	10.0	190.685

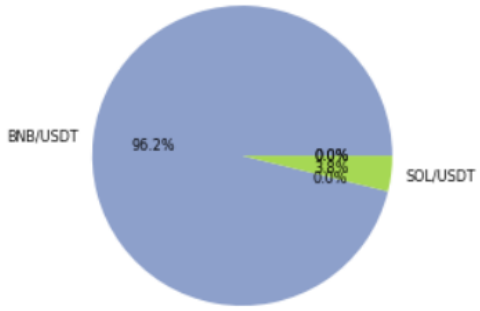


Рисунок 6. Результат оптимізації за допомогою умовної вартості під ризиком (CVaR)

Not allocated budget: 234.28 *out of* 5000.00

	Price	Optimal Weights	Optimal Volume	Total pay
BTC/USDT	58746.57000	0.2	0.0	0.00000
ETH/USDT	1840.46000	0.2	1.0	1840.46000
BNB/USDT	311.29620	0.2	3.0	933.88860
ADA/USDT	1.21189	0.2	825.0	999.80925
SOL/USDT	19.06850	0.2	52.0	991.56200

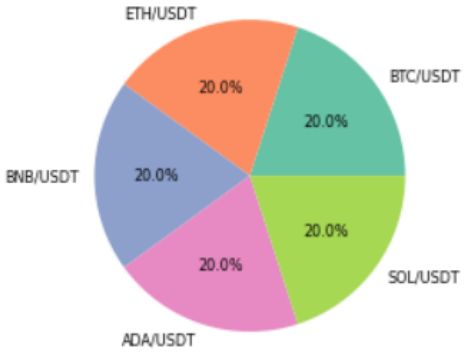


Рисунок 7. Результат оптимізації за допомогою рівноважних ваг (Equal_Weight)

```
Expected annual return: 286.9%
Annual volatility: 60.0%
Sharpe Ratio: 4.75
Not allocated budget: 3534.57 out of 5000.00
                Price  Optimal Weights  Optimal Volume   Total pay
BTC/USDT  58746.57000          0.70529             0.0     0.00000
ETH/USDT   1840.46000          0.00000             0.0     0.00000
BNB/USDT    311.29620          0.00000             0.0     0.00000
ADA/USDT      1.21189          0.03762           155.0   187.84295
SOL/USDT     19.06850          0.25709            67.0  1277.58950
```

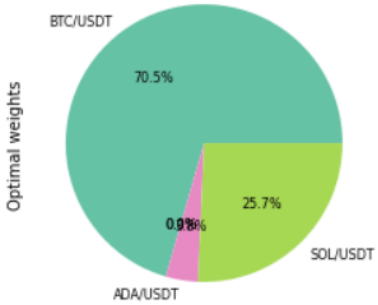


Рисунок 8. Результат оптимізації за допомогою метод мінімальної дисперсії (Min_Variance)

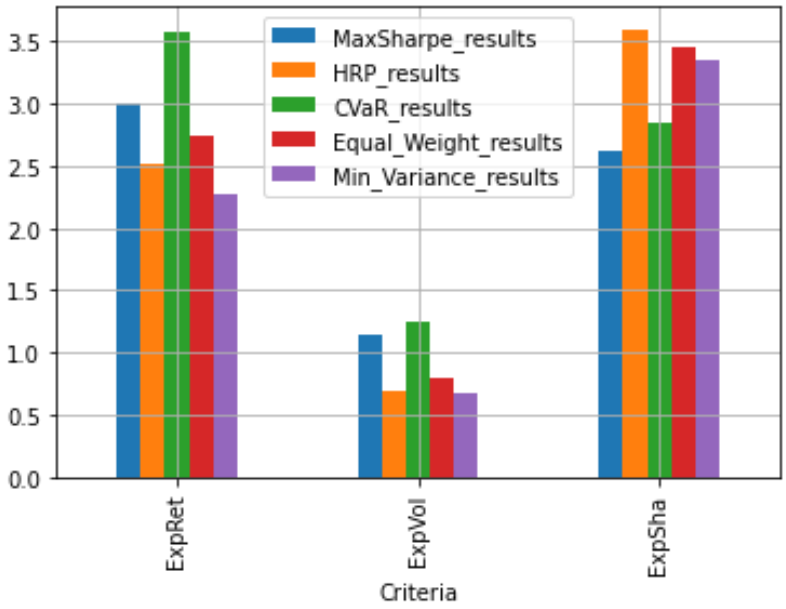


Рисунок 9. Результат порівняльного аналізу розглянутих методів

Опираючись на отримані дані, можна побачити, що очікувана прибутковість та очікувана волталильність приймають найбільшних значень при застосуванні методу оптимізації Cvar.

Методу оптимізації співвідношення Шарпа має найкращі показники очікуваної прибутковісті та очікуваної волталильності після методу Cvar.

Метод ієрархічного паритету ризиків(HRP) видає середні показники очікуваної прибутковісті та очікуваної волталильності порівняно з іншими методами, проте має найкращі показники по коефіцієнту Шарпа.

Метод мінімальної дисперсції (MinVar) має найменшу очікувану прибутковість та волатильність.

Метод робастної(надійної) оптимізації портфеля(Robust) має загалом середні показники очікуваної прибутковісті, очікувано волталильності та коефіцієнту Шарпа.

5. ВИСНОВКИ

Під час цієї роботи я проаналізував нечіткий інвестиційний портфель використовуючи такі метрики як бета коефіцієнт, модель оцінки капітальних активів (CAPM); коефіцієнт Шарпа, коефіцієнт Трейнора, M2 (модель Модльяні-Міллєра), альфа Дженсена, показник співвідношення інформації та коефіцієнт Сортіно. Було створено програму для порівцняльного аналазу різних похідних методів отпимізації Марковіца інвевстиційного портфелю в умовах невизначеності.

Використовуючи ринкові дані курсів криптовалют з сайту binance.com, було досліджено наступні криптовалюти : Bitcoin (BTC/USDT), Ethereum (ETH/USDT), Binance coin (BNB/USDT), Cardano (ADA/USDT), Soland (SOL/USDT).

Було використано програму Jupyter Notebook як основне середовище розробки, а Python був використаний як основна мова програмування. Для оптимізації за допомогою методів коефіцієнта Шарпа, ієрарахічного паритету ризиків (HPR), умовної вартості під ризиком(CVaR), мінімальної дисперсції (Min_Var) та рівних ваг (Equal_Weight) було використано бібліотеку PyPortfolioOpt.

Використані методи враховують більш широкий спектр ризиків і нестабільностей ринку порівняно з моделлю Марковіца, яка ґрунтується на припущенні стабільності ринків і лінійності кореляцій між активами. Ці методи пропонують більшу гнучкість у підході до управління портфелем, що дозволяє краще адаптуватися до змінних ринкових умов і різноманітних інвестиційних цілей. Основною перевагою продукту є його точність, широкий спектр розглянутих метрик та простий для розуміння інтерфейс.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Economy Nayka [Електронний ресурс] – Режим доступу до ресурсу: http://www.economy.nayka.com.ua/?op=1&z=3671
2. Binance [Електронний ресурс] – Режим доступу до ресурсу: https://binance.com
3. Hudson & Thames [Електронний ресурс] – Режим доступу до ресурсу: https://hudsonthames.org/portfolio-optimisation-with-portfoliolab-hierarchical-risk-parity/
4. Investopedia [Електронний ресурс] – Режим доступу до ресурсу: https://www.investopedia.com/terms/c/conditional_value_at_risk.asp
5. WallstreetMojo. Minimum Variance Portfolio. URL : https://www.wallstreetmojo.com/minimum-variance-portfolio/
6. Investopedia. Equal Weight Investing. URL : https://www.investopedia.com/terms/e/equalweight.asp

СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ДЛЯ МЕНЕДЖМЕНТУ ПОРТФЕЛЬНИХ РИЗИКІВ

Копа М.В.[1], Кузнєцова Н.В.

Національний технічний цніверситет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] mrmks626@gmail.com

Метою даного дослідження була розробка системи підтримки прийняття рішень для управління портфельними ризиками, яка дозволяє автоматизувати аналіз ризиків і оптимізацію інвестиційного портфеля. Було виконано оцінку портфельних ризиків з використанням традиційних методів, оптимізацію портфелів на основі моделі Марковіца, що дозволяє знайти можливі стратегії диверсифікації. Оригінальною частиною розробленої СППР було використання моделі DCC-GARCH для аналізу волатильності і наявності кореляцій між активами. Основною перевагою створеної СППР є комплексний підхід до інвестування з можливістю побудови моделей оцінювання ризикованості певних активів, врахування сильної волатильності і кореляції між ними в інвестиційному портфелі, та, врешті-решт, для зменшення фінансових втрат і підвищення дохідності .

Ключові слова: портфельні ризики, СППР, Value-at-Risk, модель Марковіца.

1. ВСТУП

В сучасних умовах фінансовий ринок характеризується високою волатильністю та непередбачуваністю, що утворює численні виклики для інвесторів та фінансових аналітиків. Управління портфельними ризиками є невід'ємною складовою процесу прийняття інвестиційних рішень, яка потребує використання ефективних математичних моделей та спеціалізованих інструментів для оцінки ризиків і максимізації дохідності. Постійно зростаюча складність ринкових умов, поява нових фінансових інструментів та збільшення обсягу доступних даних зумовлюють потребу у сучасних системах аналізу та управління ризиками.

Дана робота присвячена опису розробленої системи підтримки прийняття рішень (СППР) для менеджменту портфельних ризиків. Застосування моделей, таких як Value-at-Risk (VaR) та Conditional Value-at-Risk (CVaR), дозволяє значно покращити процес управління ризиками та підвищити ефективність інвестиційних рішень. Ці моделі не лише дають змогу передбачити можливі втрати, але й дозволяють аналізувати ситуації за умов надзвичайно високої волатильності, що є критичним фактором для інвесторів.

2. ОСНОВНІ МОДЕЛІ

2.1. Сучасна теорія портфелю Марковіца

Модель Марковіца є основою сучасної теорії портфельного аналізу, пропонуючи методологію оптимізації, що враховує як очікувану дохідність активів, так і рівень ризику, що вимірюється стандартним відхиленням. Основний принцип цієї моделі полягає в диверсифікації активів, що дозволяє знижувати загальний ризик портфеля завдяки включенню

в нього активів із низькою кореляцією між собою. Ефективна межа, яка є центральним концептом моделі, допомагає інвесторам знайти оптимальні портфелі, які максимізують дохідність за заданого рівня ризику. Такий підхід дозволяє забезпечити найкраще співвідношення ризику та очікуваної дохідності, що є важливим для інвесторів із різними профілями ризику. Вона не лише демонструє набір оптимальних рішень, а й дає можливість інвесторам зрозуміти, які портфелі є недоцільними. Портфелі, що знаходяться поза нею, є менш доцільними для обрання, оскільки при тому ж рівні ризику можна знайти варіанти з вищою дохідністю [1, 2, 3].

2.2. Value at Risk (VaR)

VaR є інструментом, який використовується для вимірювання потенційних втрат у портфелі при заданому рівні довіри та певному часовому горизонті. Цей метод дозволяє фінансовим аналітикам та менеджерам оцінити, які втрати можуть статися з певною ймовірністю за конкретний період часу [1, 3]. Методики розрахунку VaR включають:

1. Історичне моделювання, що базується на фактичних даних минулих періодів і дає можливість аналізувати реальні сценарії.
2. Метод на основі припущень про розподіл, зазвичай нормального, який дозволяє швидко обчислити VaR за допомогою середнього значення та стандартного відхилення.
3. Метод Монте-Карло, який використовує випадкові генерації для моделювання різних сценаріїв та оцінки потенційних ризиків.

2.3. Conditional Value at Risk (CVaR)

CVaR забезпечує більш детальний аналіз ризиків, оскільки враховує середні втрати, які перевищують значення VaR. Це робить його більш ефективним інструментом для оцінки ризиків у випадках з екстремальними втратами. CVaR допомагає зрозуміти, які середні втрати можна очікувати у найгірших сценаріях, коли значення VaR вже перевищено. Це особливо важливо для інвесторів, які прагнуть зменшити ризики за умов високої невизначеності на ринку [1].

CVaR також має перевагу перед традиційним VaR у тому, що відповідає вимогам субадитивності, що означає зниження загального ризику під час об'єднання портфелів. Завдяки цьому підходу CVaR підходить для складних портфелів з численними активами, оскільки дозволяє більш реалістично оцінювати ризики за різних умов ринку.

2.4. Коефіцієнт Шарпа

Коефіцієнт Шарпа є одним із ключових інструментів для оцінки ефективності портфелів. Він розраховується як співвідношення між середньою дохідністю портфеля, скоригованою на безризикову ставку, і стандартним відхиленням портфеля:

$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p},$$

де R_p – середня дохідність портфеля, R_f – безризикова ставка, σ_p – стандартне відхилення портфеля.

Високе значення коефіцієнта Шарпа свідчить про те, що портфель має кращу дохідність на одиницю ризику. Це допомагає інвесторам визначити найефективніші портфелі з погляду співвідношення ризику та доходу [2].

2.5. DCC-GARCH

Модель DCC-GARCH (Dynamic Conditional Correlation Generalized Autoregressive Conditional Heteroskedasticity) дозволяє оцінювати динамічні кореляції між активами в портфелі та моделювати їх зміну в часі. Ця модель є ефективним інструментом для розрахунку волатильності та кореляції, що змінюються з часом, забезпечуючи більш точний аналіз ризиків. Використання DCC-GARCH дає змогу враховувати змінну природу ринкових умов, що є важливим при побудові прогнозів і виборі стратегій управління портфелем. Включення таких моделей у СППР підвищує гнучкість системи та точність оцінки ризиків.

DCC-GARCH моделює кореляції між залишками від окремих GARCH-моделей для кожного активу в портфелі, забезпечуючи аналіз взаємозалежності активів навіть під час періодів високої волатильності.

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Процес аналізу розпочинається із завантаження історичних ринкових даних за допомогою бібліотеки yfinance. Цей інструмент дозволяє отримувати надійні дані про ціни акцій із джерел, що підтримують актуальність і точність інформації. Завантажені дані зберігаються у форматі CSV, що забезпечує зручність при подальшому використанні та повторних обчисленнях, знижуючи навантаження на систему.

На цьому етапі важливо перевірити дані на наявність пропусків та аномальних значень. Пропуски можуть виникати через вихідні дні чи технічні перебої, і їх корекція є необхідною для забезпечення цілісності аналізу. Після обробки даних система розраховує щоденні дохідності активів за формулою:

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}},$$

де P_t – ціна активу на день t, а P_{t-1} – ціна на попередній день. Такий підхід дозволяє відстежувати зміни вартості активів у динаміці та аналізувати їх вплив на загальну дохідність портфеля. Додатково обчислюються логарифмічні дохідності, які забезпечують адитивність і є корисними при аналізі та порівнянні:

$$R_{t_{logarithmic}} = ln\left(\frac{P_t}{P_{t-1}}\right).$$

Підготовка даних завершується візуалізацією, що включає графіки залежності цін від часу, графіки дохідності та кумулятивної суми доходів (рис. 1–2). Це надає аналітикам початкове уявлення про поведінку активів у часовій перспективі.

Рисунок 1. Графік цін закриття MSFT, WMT, PG та GM

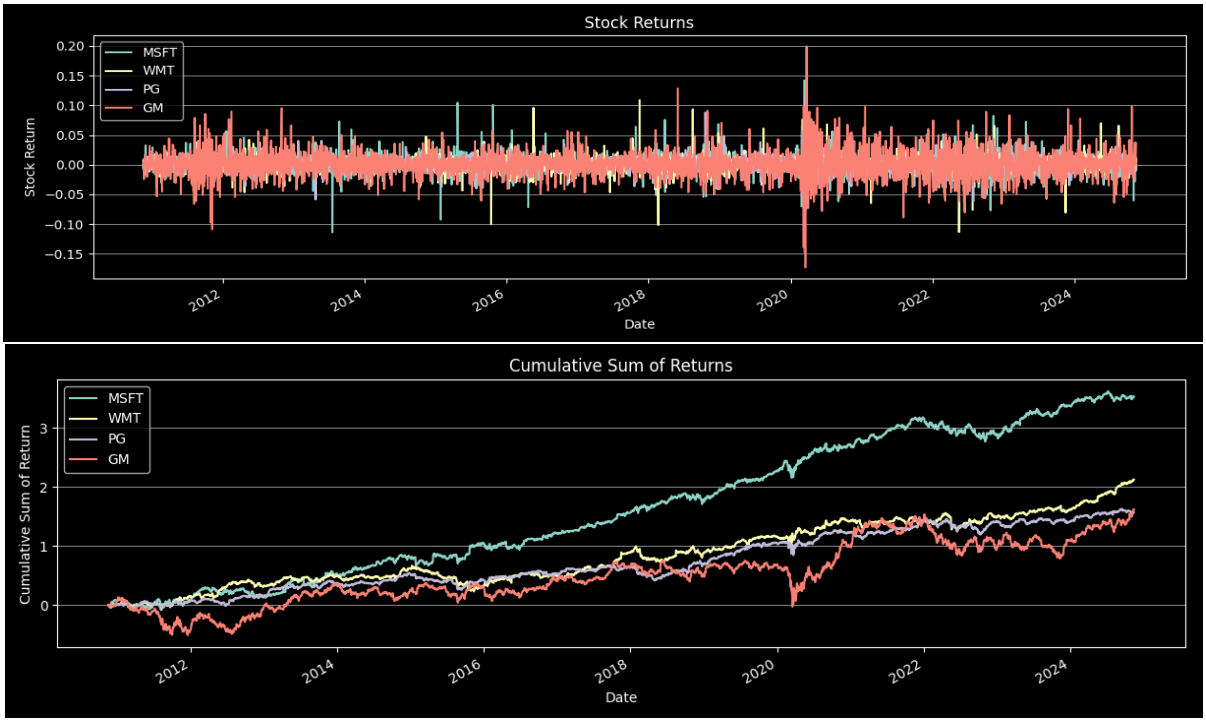


Рисунок 2. Графіки доходу з активів MSFT, WMT, PG та GM

Після обробки та підготовки даних система переходить до застосування сучасної теорії портфелю Марковіца. Цей етап включає генерацію великої кількості випадкових портфелів із різними вагами активів, що дозволяє створити простір можливих комбінацій ризику та дохідності.

Головною метою є побудова ефективної межі, що відображає найкращі портфелі з погляду співвідношення ризику та дохідності. Це досягається шляхом розв'язання задачі оптимізації:

$$Minimize\ \sigma_p = \sqrt{\omega^T \Sigma \omega}, \sum_{i=1}^{N} \omega_i = 1,$$

де ω – вектор ваг, а Σ – коваріаційна матриця.

Ефективна межа дозволяє інвесторам обирати портфелі з мінімальним ризиком для заданої дохідності або з максимальною дохідністю за певного рівня ризику. Система також обчислює портфель із найвищим коефіцієнтом Шарпа (рис. 3).

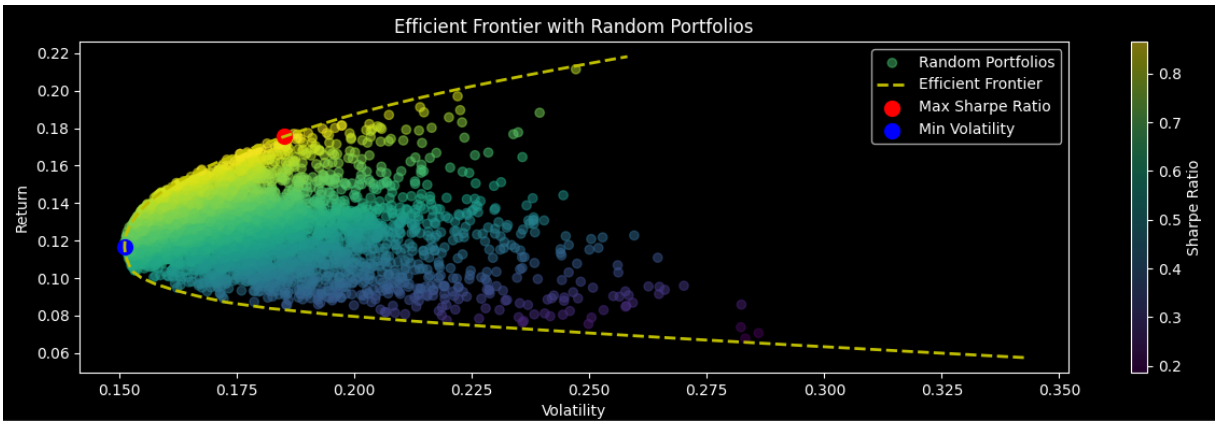


Рисунок 3. Графік випадкових портфелів та ефективного фронту

Система забезпечує аналіз динаміки оптимальних портфелів у часі, що дозволяє відстежувати зміну пропорцій активів у портфелі за місяцями (рис. 4). Це важливо для адаптації стратегії управління активами в умовах зміни ринкових умов. Щомісячний аналіз показує, як портфелі реагують на ринкові коливання та дозволяє краще зрозуміти довгострокові тенденції.

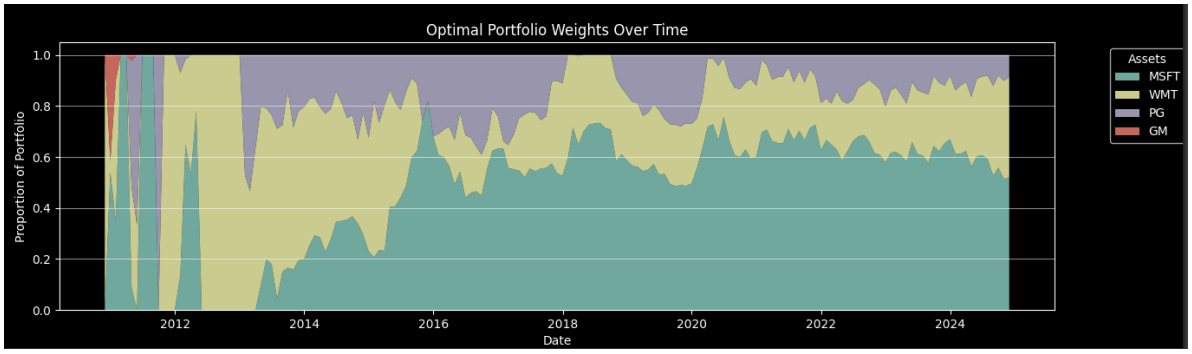


Рисунок 4. Графік зміни пропорцій оптимального портфелю впродовж 14 років

Для обраних 50 портфелів із ефективного фронту проводиться оцінка ризиків за допомогою показників VaR (Value at Risk) та CVaR (Conditional Value at Risk). VaR показує максимальні можливі втрати при заданому рівні довіри, в той час як CVaR визначає середні втрати за умов, що перевищують значення VaR. Ці показники дозволяють не лише оцінити можливі втрати, але й зрозуміти, наскільки портфелі стійкі до екстремальних ринкових умов.

Останнім етапом є оцінка динамічних кореляцій за допомогою моделі DCC-GARCH (Dynamic Conditional Correlation GARCH). Ця модель дозволяє аналізувати зміну кореляцій між активами у часі, що особливо важливо під час періодів підвищеної волатильності. Врахування змін у кореляціях допомагає краще моделювати ризик і покращує стратегії управління портфелем.

Завдяки використанню комплексного підходу до аналізу та прогнозування, система дозволяє приймати обґрунтовані рішення та ефективно управляти ризиками навіть в умовах нестабільного ринку (рис. 5).

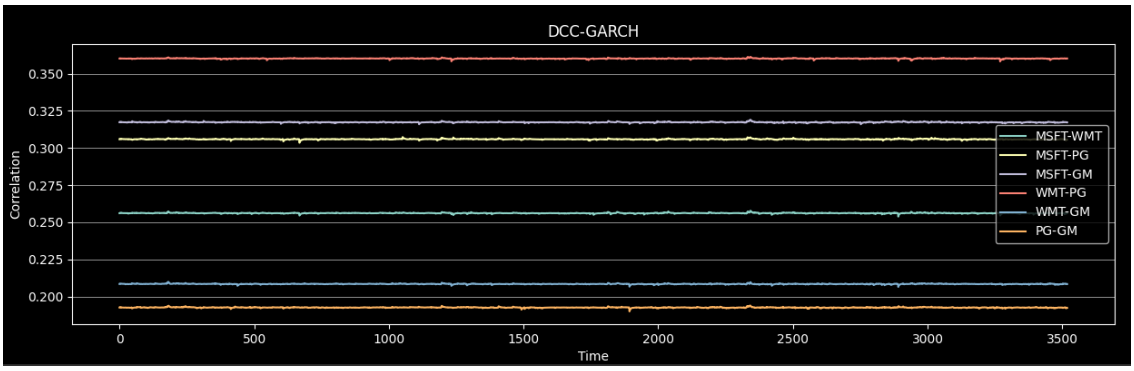


Рисунок 5. Динаміка кореляцій між активами за допомогою моделі DCC-GARCH

4. ВИСНОВКИ

Розроблена система підтримки прийняття рішень для управління портфельними ризиками надає основи для автоматизації аналізу ризиків і оптимізації інвестиційних портфелів. Використання математичних моделей дає змогу оцінювати ризики та дохідність на основі історичних даних і прогнозувати можливі сценарії розвитку. Основні результати дослідження включають оцінку портфельних ризиків з використанням традиційних методів,

що дозволяє виявити потенційні втрати в стандартних умовах; оптимізацію портфелів на основі моделі Марковіца, що дозволяє знайти можливі стратегії диверсифікації, але не гарантує уникнення усіх ризиків; використання DCC-GARCH для аналізу кореляцій між активами, що дає змогу відслідковувати зміну взаємозв'язків між активами, але не враховує всі можливі зовнішні фактори, які можуть впливати на ринок.

Хоча система дає змогу більш ефективно управляти портфелями в умовах ринкової невизначеності, її точність та застосовність обмежені через необхідність врахування великої кількості змінних і невизначеностей ринку. Це може призвести до певних неточностей при прийнятті рішень, зокрема у випадках, коли ринкова ситуація відрізняється від історичних даних. Для покращення системи можливими напрямками розвитку є інтеграція більш гнучких методів прогнозування, таких як машинне навчання, для врахування нових, швидко змінюваних ринкових умов. Також доцільно включити інші моделі, що враховують специфічні фактори, такі як політичні або макроекономічні події, що можуть значно впливати на ринки, але не завжди відображаються в традиційних фінансових моделях.

Отже, система може бути корисною як додатковий інструмент для аналізу і оцінки портфельних ризиків, однак для досягнення оптимальних результатів необхідне врахування інших факторів і застосування більш комплексного підходу до управління ризиками.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Hull, J. (2015). «Risk Management and Financial Institutions» (4th Edition). John Wiley & Sons, Inc., Hoboken, New Jersey. <https://www.simonfoucher.com/MBA/FINA%20695%20-%20Risk%20Management/riskmanagementandfinancialinstitutions4theditionjohnhull-150518225205-lva1-app6892.pdf> (last accessed: 9 November 2024).

2. Bodie, Z., Kane, A., Marcus, A. J. (2014). «Investments.» McGraw-Hill Education, New York, NY. <https://info.mheducation.com/rs/128-SJW-347/images/Bodie_Preface_Investments_13e.pdf> (last accessed: 12 November 2024).

3. Elton, E. J., Gruber, M. J., Brown, S. J., & Goetzmann, W. N. (2014). «Modern Portfolio Theory and Investment Analysis.» John Wiley & Sons, Inc., Hoboken, New Jersey. <https://elearn.daffodilvarsity.edu.bd/pluginfile.php/913300/mod_label/intro/Modern%20Portfolio%20Theory%20and%20Investment%20Analysis.pdf> (last accessed: 12 November 2024).

СППР ОЦІНЮВАННЯ КРЕДИТНИХ РИЗИКІВ

Крючков Д.О.[1], Кузнєцова Н.В.

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] kriuchkovdima@gmail.com

Актуальність дослідження зумовлена стрімким зростанням обсягу даних та необхідністю вчасного виявлення ризиків у фінансовому секторі, де успішність кредитних рішень суттєво впливає на фінансову стабільність установ. У роботі запропоновано сучасний підхід до оцінки кредитоспроможності клієнтів шляхом інтеграції систем підтримки прийняття рішень (СППР), що базуються на алгоритмах машинного навчання. Було побудовано ансамблеві моделі, які продемонстрували високу якість і точність прогнозування клієнтів, схильних до дефолту. Найкращими виявились моделі бустингу: Gradient boosting, XGBoost та AdaBoost, які демонструють точність прогнозування на рівні 94-95%. Алгоритму бегінгу демонструють меншу точність, проте все одно достатньо високу точність на рівні 91% за метрикою f1-score. Розроблена СППР дозволяє підвищити точність аналізу та прогнозування ризиків, що дає змогу фінансовим установам ефективніше оцінювати платоспроможність позичальників і мінімізувати кредитні ризик, забезпечуючі стійкості та прибутковості кредитних організацій у динамічному ринковому середовищі.

Ключові слова: оцінювання кредитних ризиків, система підтримки прийняття рішень, машинне навчання, градієнтний бустинг.

1. ВСТУП

У сучасному світі, де успіх у багатьох галузях визначається швидкістю та точністю обробки інформації, зростає потреба у розробці систем підтримки прийняття рішень (СППР). Такі системи відкривають нові можливості для ефективного управління, детального аналізу великих обсягів даних і прийняття обґрунтованих рішень, що особливо важливо для конкурентних ринків.

Фінансовий сектор, зокрема сфера кредитування, потребує сучасних рішень для оцінки кредитоспроможності клієнтів. Зростаюча кількість інформації та її складність роблять традиційні методи обробки даних недостатньо ефективними. Використання СППР із застосуванням алгоритмів машинного навчання дозволяє фінансовим установам проводити глибший і точніший аналіз, зокрема прогнозувати ризики і платоспроможність позичальників.

Особливо важливим є те, що СППР здатні виявляти ненадійних клієнтів, що допомагає банкам та іншим фінансовим установам уникати значних збитків. Вчасне розпізнавання потенційних ризиків знижує кількість неповернених кредитів і мінімізує фінансові втрати, що сприяє стабільності та прибутковості установ. Завдяки автоматизованим алгоритмам і точному аналізу даних, СППР суттєво зменшують кредитні ризики, роблячи внесок у зміцнення позицій організацій у динамічному фінансовому середовищі [1].

2. ОПИС НАБОРУ ДАНИХ

В даній роботі було використано набір даних, який відображає історичні дані щодо наданих кредитних позик та стану їх сплати. У вибірці містяться такі характеристики як: вік, місячний прибуток, стаж (в роках), наявність нерухомості, ціль, сума та відсоткова ставка позики, кількість попередніх позик клієнта та наявність попередніх несплат, попередня експертна оцінки (оцінка фахівцем) та цільова характеристика – індикатор погашення позики.

Моделі машинного навчання мають певні вимоги до формату вхідних даних і, зокрема, більшість моделей потребує виключно чисельні значення для подальшого навчання. Оскільки деякі характеристики в нашому наборі представляють категоріальні значення, постає необхідність в їх конвертуванні в чисельний формат, тож було виконано попередній їх аналіз, а після цього було застосовано методи кодування: ordinal та target encoding. На наступному етапі було виконано масштабування всієї сукупності даних для покращення подальших результатів побудованих моделей [2].

Для аналізу вибірки було побудовано кореляційну матриця (рис. 1), з якої видно взаємозв'язок вхідних характеристик та з цільовою змінною (‘loan_status’). Зокрема, можна відмітити значну кореляцію цільової ознаки з щомісячним прибутком, тип нерухомості клієнта, та експертною оцінкою позики фахівцем.

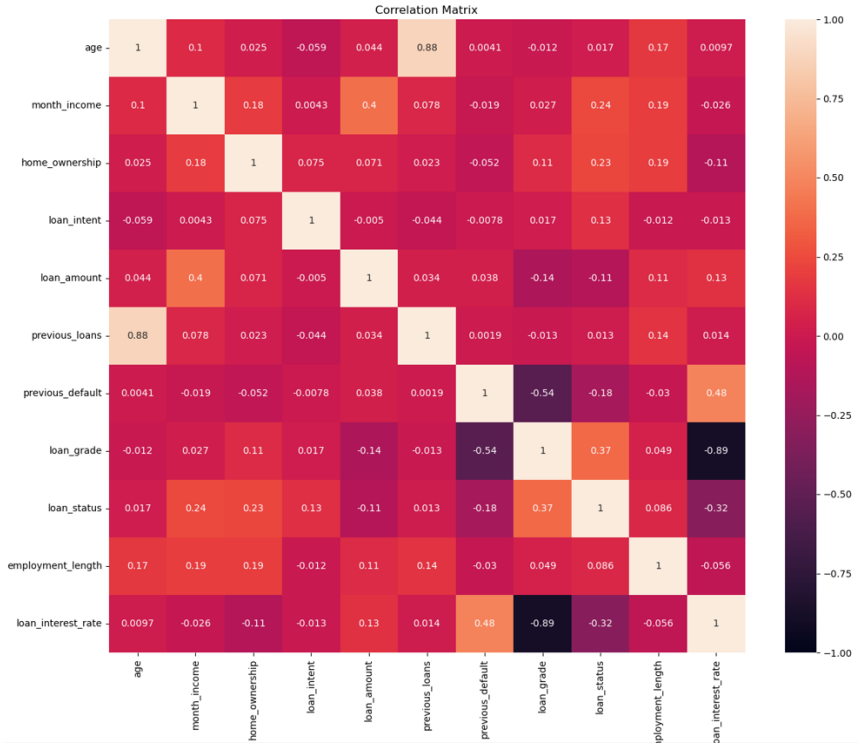


Рисунок 1. Кореляційна матриця даних

Окрім кодування та масштабування даних було виконано відбір значущих характеристик вибірки. Даний підхід часто може позитивно сприяти на якість побудованих моделей, оскільки найменш значущі характеристики виключаються. Зокрема, було виключено такі характеристики, як: кількість попередніх позик, наявність попередніх неповернень та відсотковку ставку за позикою [3].

3. ОПИС ВИКОРИСТАНИХ МОДЕЛЕЙ МАШИННОГО НАВЧАННЯ

У роботі було досліджено використання ансамблевих моделей машинного навчання для вирішення задачі оцінки кредитоспроможності клієнтів. Ансамблеві моделі – це підхід, що об'єднує прогнози кількох базових моделей для підвищення точності та стабільності прогнозів. Замість використання одного алгоритму, ансамблеві моделі комбінують результати кількох слабших алгоритмів, щоб знизити ризик помилок, які можуть виникати через обмеження окремих моделей. Кожен алгоритм може мати свої сильні та слабкі сторони і спеціалізуватися на різних аспектах даних, тому їхнє об'єднання дозволяє зменшити похибки та підвищити точність прогнозів [4].

Основні методи, що розглянуті в роботі, включають:

Bagging (Bootstrap Aggregating), який підвищує стабільність і точність моделей, особливо тих, що схильні до перенавчання, таких як дерева рішень. Він створює кілька незалежних моделей на основі випадкових підвибірок даних і об'єднує їх прогнози, що дозволяє зменшити розкид і підвищити загальну точність.

Gradient Boosting, потужний метод бустингу, що послідовно навчає нові моделі на основі залишкових помилок попередніх. На відміну від Bagging, який використовує незалежні моделі, Gradient Boosting будує ансамбль послідовно, де кожна нова модель коригує помилки попередньої. Цей підхід дозволяє досягти високої точності навіть у складних задачах.

XGBoost (Extreme Gradient Boosting) – оптимізований і прискорений варіант Gradient Boosting, створений для підвищення ефективності, точності та швидкості навчання. XGBoost підтримує паралельне обчислення, обробку пропущених значень і контроль перенавчання через регуляризацію, що робить його одним з найпопулярніших методів для задач класифікації та регресії.

AdaBoost (Adaptive Boosting) – алгоритм бустингу, що поєднує прості моделі, такі як «пні» (дерева рішень з одним рівнем), для поступового покращення точності класифікації. AdaBoost адаптивно надає вищу вагу тим спостереженням, які важче класифікувати, що дозволяє зосередитися на складних прикладах і підвищити точність моделі.

Усі ці методи показують високу ефективність у задачах, що потребують точного прогнозування та надійності, і є перспективними для фінансового сектору, де важливо оцінювати кредитні ризики з максимальною точністю.

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Під час вирішення задачі класифікації для оцінювання адекватності побудованих моделей було використано наступні метрики: $accuracy$, $precision$, $recall$, F_1, AUC-ROC, а також крива Precision-Recall [5].

Часто, на адекватність моделей впливає їх початкове налаштування (набір гіперпараметрів), тому, з метою вирішити даний недолік було здійснено решітчастий пошук. Загалом, решітчастий пошук – це метод пошуку оптимальних гіперпараметрів моделі, що передбачає перебір комбінацій заданих параметрів, навчання моделей на цих комбінаціях, їх оцінок та відбору найкращої.

Часто в моделях машинного навчання спостерігається перенавчання (overfitting). У таких випадках, моделі є занадто адаптованими на навчальних даних та мають погані узагальнюючі якості. З метою протидії перенавчанню, під час решітчастого пошуку застосовувався підхід кросс-валідації (cross-validation). Кросс-валідація – це підхід до оцінювання ефективності та узагальнюючої здатності моделей, шляхом розбиття вхідних даних та частини (фолди), та варіюючи комбінації частин для навчання та валідації моделі.

Після виконання всіх етапів попередньої обробки і підготовки даних було отримано оптимальні моделі бустингу та беггінгу. В таблиці 1 наведено значення метрик адекватності для отриманих моделей.

Таблиця 1. Метрики адекватності моделей на тестовій вибірці

	Gradient boosting	XGBoost	Ada boost	Bagging
accuracy	0.928349	0.929470	0.919003	0.862555
precision	0.933537	0.932969	0.931969	0.918288
recall	0.977806	0.980042	0.966789	0.904359
f1-score	0.955159	0.955926	0.949060	0.911270

Окрім цього, також було побудовано ROC та Precision-Recall криві (рис. 2 та рис. 3 відповідно).

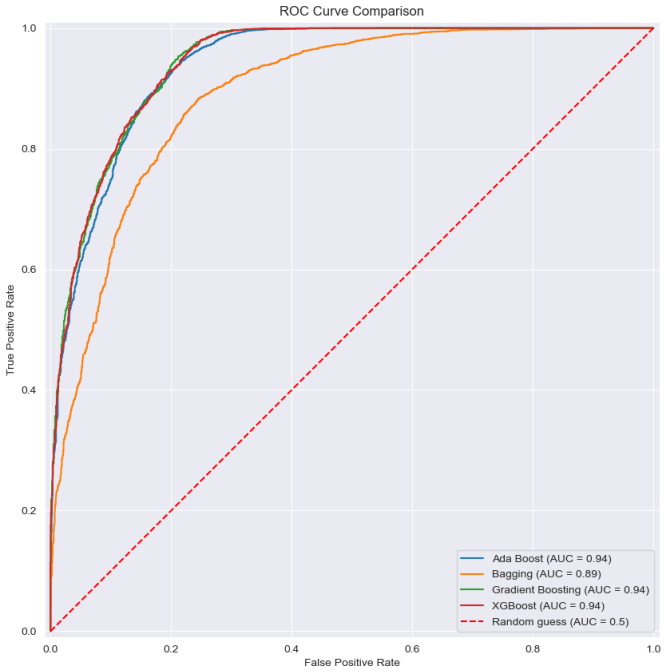


Рисунок 2. ROC-криві побудованих моделей

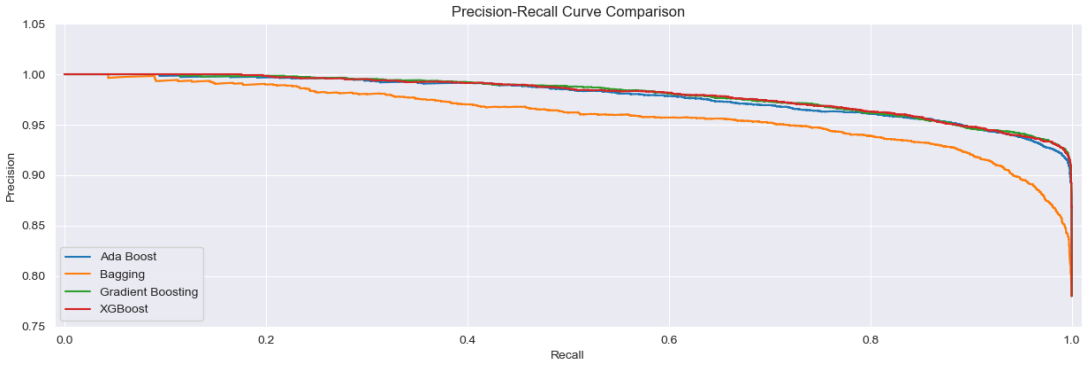


Рисунок 3. Precision-Recall криві побудованих моделей

Проаналізувавши отримані результати, можна помітити що Gradient boosting, XGBoost та AdaBoost демонструють відмінні результати прогнозування на рівні 94–95%. Bagging в свою чергу демонструє меншу, але все одно гарну точність на рівні 91% за метрикою f1-score. Дані результати свідчать про високу точність та доцільність застосування ансамблевих моделей для вирішення задачі прогнозування несплати кредиту потенційним клієнтом.

5. АРХІТЕКТУРА СППР

Будь-яка складна система зазвичай складається з численних компонентів, які виконують різні функції, але працюють у тісній взаємодії для досягнення спільної мети. Від того, наскільки вдало спроектована архітектура системи, залежить її ефективність, надійність та здатність адаптуватися до змін. Продумана структура та чітке визначення ролей кожного компонента забезпечують оптимальну продуктивність, спрощують обслуговування та дозволяють системі масштабуватися відповідно до нових потреб [6, 7].

Розроблена СППР також має власну архітектуру, структурна схема якої, зображена на рисунку 4.

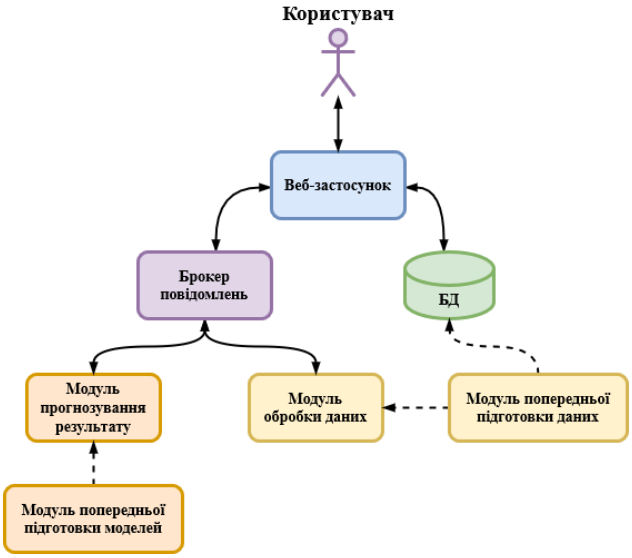


Рисунок 4. Структурна схема СППР

Сама ж СППР складається з наступних модулів:

●***Веб-застосунок***. Виступає інтерфейсом для користувачів, надаючи зручний доступ до системи через веб-інтерфейс. Він обробляє користувацькі запити, збирає введені дані й відображає результати. Цей компонент відповідальний за взаємодію між базою даних, модулем обробки даних, та прогнозування результатів.

●***Брокер повідомлень (БП).*** Забезпечує асинхронний обмін повідомленнями між модулями системи. З його допомогою, знижується зв'язність модулів, підвищуючи масштабованість і гнучкість, дозволяючи їм ефективно взаємодіяти без прямої залежності один від одного.

●***База даних (БД).*** Сховище вхідних і оброблених даних, а також результатів прогнозування.

●***Модуль обробки даних.*** Даний елемент відповідає за підготовку даних до застосування побудованих моделей. Цей процес включає в себе кодування даних та їх масштабування.

●***Модуль прогнозування.*** Виконує передбачення результату, використовуючи заздалегідь навчені моделі.

●***Модуль попередньої підготовки даних та навчання моделей.*** В даних компонентах відбувається початковий аналіз та підготовка даних, а також навчання та збереження побудованих моделей.

5. ВИСНОВКИ

У даній роботі розглянуто можливості застосування ансамблевих моделей машинного навчання для вирішення задачі оцінювання кредитоспроможності клієнтів фінансових установ. Проведений аналіз і отримані результати підтверджують доцільність використання цих моделей, зокрема бустингові моделі продемонстрували відмінну точність і надійність. Крім того, було спроектовано та розроблено систему підтримки прийняття рішень (СППР), що базується на даних моделях, з продуманою архітектурою, яка забезпечує легку інтеграцію та високу масштабованість, роблячи СППР ефективним інструментом для аналізу кредитних ризиків у банківському секторі.

У рамках подальших досліджень доцільно розглянути інші методи класифікації, зокрема нейронні та Байєсівські мережі, які можуть покращити точність і адаптивність моделі. Перспективним напрямком є також аналіз виживаності (лояльності) клієнтів протягом терміну кредитування, що дозволяє прогнозувати ймовірність виконання зобов’язань за кредитами на певний період часу. Такий підхід додасть новий рівень глибини в оцінювання ризиків та допоможе фінансовим установам ефективніше управляти портфелями кредитів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Солошенко О. М. Моделі і методи оцінювання кредитоспроможності фізичних осіб : автореф. дис. канд. техн. наук. : 01.05.04 – системний аналіз і теорія оптимальних рішень. – Київ, 2016. – 22 с.
2. Kelleher, A., Kelleher, A. (2019). Machine Learning in Production: Developing and Optimizing Data Science Workflows and Applications. Велика Британія: Pearson Education.
3. Galli, S. (2022). Python Feature Engineering Cookbook: Over 70 Recipes for Creating, Engineering, and Transforming Features to Build Machine Learning Models, 2nd Edition. Велика Британія: Packt Publishing, Limited.
4. Ensemble Machine Learning: Methods and Applications. (2012). Німеччина: Springer.
5. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Швейцарія: Springer New York.
6. Kleppmann, M. (2017). Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. Сполучені Штати Америки: O'Reilly Media.
7. Newman, S. (2015). Building Microservices: Designing Fine-Grained Systems. Сполучені Штати Америки: O'Reilly Media.

РОЗРОБКА І ДОСЛІДЖЕННЯ МОДЕЛЕЙ НЕЙРОННИХ МЕРЕЖ ДЛЯ SEMANTIC ТА INSTANCE СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ

Лавренко В.В.[1], Недашківська Н.І.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] lavrenko.vladyslav@lll.kpi.ua,
[2] nedashkovskaya.nadezhda@lll.kpi.ua [0000-0002-8277-3095]

В роботі розглядається тема Semantic та Instance сегментації зображень. Досліджено моделі для Instance сегментації, створено власну реалізацію для завдання Semantic сегментації на основі U-Net та FCN моделей. Приведено візуальні результати дослідження.

Ключові слова: Semantic сегментація, Instance сегментація, Mask R-CNN, U-Net, FCN.

1. ВСТУП

Технології обробки зображень відіграють важливу роль у багатьох галузях, зокрема медицині, безпеці, автомобільній промисловості. Важливим завданням обробки зображень є сегментація – поділ зображення на окремі області або об’єкти. Таким чином можна виділити 3 категорії:

- Розпізнавання об’єктів – це виявлення об’єктів на зображенні та подальше присвоєння ним обмежувальних рамок.
- Semantic сегментація – групування схожих типів об’єктів в одну категорію і позначення її одним кольором.
- Instance сегментація – розпізнавання та присвоєння кожному об’єкту на зображенні окремої сутності.

В роботі розглянуто підходи Semantic та Instance сегментації, досліджено моделі та розроблено власні рішення на основі моделей U-Net, FCN та Mask R-CNN.

2. ПОСТАНОВКА ЗАДАЧІ

Метою роботи є розробка моделей на основі U-Net та FCN для задачі семантичної сегментації та їх вдосконалення, а також вдосконалення існуючої моделі Mask R-CNN для задачі сегментації екземплярів. Об’єктом дослідження є набори реальних даних Cityscapes та COCO (Common Objects in Context).

3. ОПИС НАБОРІВ ДАНИХ ТА АЛГОРИТМІВ

Набір даних «Звичайні об'єкти в контексті» **(COCO, Common Objects in Context**) містить понад 330 000 зображень із приблизно 1,5 мільйонів об'єктів розподілених на 80 категорій, серед яких меблі, домашні тварини, люди та предмети побуту, відображені в різних контекстах. Така різноманітність дозволяє алгоритмам розпізнавати об'єкти навіть у випадках часткового перекриття або незвичних ракурсів [1].

Набір даних **Cityscapes**, у свою чергу, оптимізований для сегментації вулиць та середовищ, пов’язаних з містами. Він охоплює близько 30 класів об'єктів, включно з автомобілями, пішоходами та знаками дорожнього руху, що дозволяє моделювати складні для моделей машинного навчання місця подій звичайних вулиць. На практиці це може

виглядати так: самокерований автомобіль, оснащений алгоритмом, навченим на наборі Cityscapes, може не тільки визначити розташування пішохода, але й врахувати інші об'єкти, як-от велосипедистів чи знаки, що дозволяє йому приймати безпечніші рішення під час руху в місті [2].

Mask R-CNN. Дана нейронна мережа є розширеним алгоритмом Faster R-CNN для задачі сегментації. На початку роботи зображення проходить через згорткову нейронну мережу (в нашому випадку ResNet50 та ResNeXt) для виділення ознак. Після цього, на основі карти ознак відбувається передбачення можливих областей, де можуть знаходитися об’єкти за допомогою RPN. Далі ці області нормалізуються і кожній з них призначається клас, до якого вони належать, а також коректуються межі об’єкту. Врешті-решт створюється маска для кожного об’єкту, яка визначає, які пікселі до нього належать [3].

U-Net. Глибока нейронна мережа, яка призначена для задач сегментації зображень. Вона складається з “U”-подібної архітектури, де в лівій частині моделі знаходиться енкодер, який відповідає за зменшення розмірності зображення, виділяючи важливі ознаки, а в правій частині знаходиться декодер, який відновлює роздільну здатність зображення. Крім того, важливо зауважити, що в U-Net існують з’єднання (skip connections) між шарами енкодера та декодера, що дозволяє зберігати та передавати зображення під час процесу зменшення розмірності в декодер для більшої точності сегментації [4].

FCN. Оскільки U-Net є трішки вдосконаленою версією FCN, то вони досить схожі. На початку роботи FCN відбувається виділення ознак за допомогою попередньо натренованої мережі (наприклад, VGG16). Після цього, як і в U-Net відбувається зменшення роздільної здатності зображення та його збільшення. На відміну від U-Net, де карти ознак енкодера об’єднуються з картами ознак декодера, в FCN під час збільшення здатності комбінують карти ознак за допомогою операції додавання [5].

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для моделі U-Net та покращеної моделі отримали наступні результати семантичної сегментації (рис. 1–2). Для FCN та її покращеної версії результати можна побачити на рисунках 3 і 4 відповідно. Для сегментації екземплярів було проведено донавчання моделі Mask R-CNN з використанням згорткової нейронної мережі ResNet101 (базова мережа ResNet50). На рис. 5 і 6 наведено результати Instance сегментації за допомогою донавченої моделі Mask R-CNN з базовою моделлю ResNet50 та більш глибокою моделлю ResNet101, відповідно.

Рисунок 1. Результат роботи звичайної моделі U-Net

Рисунок 2. Результат роботи покращеної моделі U-Net

Рисунок 3. Результат роботи моделі FCN

Рисунок 4. Результат роботи покращеної моделі FCN

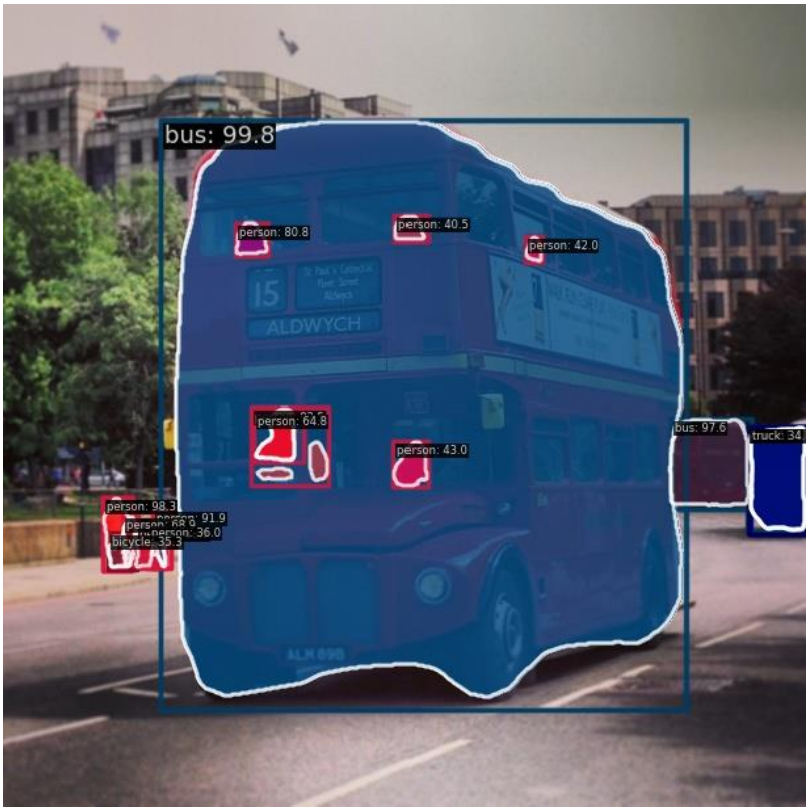


Рисунок 5. Результат роботи моделі Mask R-CNN (ResNet50)

Рисунок 6. Результат роботи моделі Mask R-CNN (ResNet101)

5. ВИСНОВОК

Отже, в дослідженні було здійснено власну реалізацію моделей U-Net та FCN, а також їх покращених версій, які показали трішки кращі результати в порівнянні з окремими моделями, і в цілому результати досить непогані на наборі даних Cityscapes.

Крім того, було вдосконалено модель Mask R-CNN за допомогою зміни згорткових нейронних мереж, де ResNet101 показала кращий результат, порівняно з базовою мережею ResNet50.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Cordts, Marius, et al. "The cityscapes dataset for semantic urban scene understanding." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. https://openaccess.thecvf.com/content_cvpr_2016/papers/Cordts_The_Cityscapes_Dataset_CVPR_2016_paper.pdf.

2. Lin, Tsung-Yi, et al. "Microsoft coco: Common objects in context." Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. Springer International Publishing, 2014. https://arxiv.org/pdf/1405.0312.

3. He, K., Gkioxari, G., Dollar, P., & Girshick, R. Mask R-CNN. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(2), 386–397. 2020. URL: https://doi.org/10.1109/tpami.2018.2844175.

4. Ronneberger, O., Fischer, P., & Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Lecture Notes in Computer Science (pp. 234–241). Springer International Publishing. 2015. URL: https://doi.org/10.1007/978-3-319-24574-4_28.

5. Shelhamer, E., Long, J., & Darrell, T. Fully Convolutional Networks for Semantic Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(4), 640–651. 2017. URL: https://doi.org/10.1109/tpami.2016.2572683.

РЕКОМЕНДАЦІЙНА СИСТЕМА НА ОСНОВІ АНАЛІЗУ БАЗИ ДАНИХ ТРАНЗАКЦІЙ

Лавріненко В.О.[1], Недашківська Н.І.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] lavriknika12345@gmail.com, [2] n.nedashkivska@gmail.com

Розробка рекомендаційних систем є ключовою складовою сучасних онлайн-сервісів, що покращує користувацький досвід і економічні показники. Метою цієї роботи є створення рекомендаційної системи на основі аналізу транзакційних даних із використанням алгоритмів FP-growth, MaxEclat, Apriori, MaxClique, TopDown і AprClique. Проведено порівняння їх ефективності на різних наборах даних для оптимізації точності та релевантності рекомендацій.

Ключові слова: Recommendation Systems, Frequent Itemsets, FP-growth, MaxClique, AprClique, Apriori.

1. ВСТУП

У сучасних умовах глобалізації та стрімкого розвитку інформаційних технологій спостерігається зростання обсягів транзакційних даних, які генеруються у різних галузях: електронній комерції, фінансах, роздрібній торгівлі тощо.

Великі масиви даних (Big Data) створюють нові можливості для бізнесу, дозволяючи автоматизувати процес прийняття рішень і прогнозувати поведінку споживачів на основі прихованих закономірностей [1].

Одним із найефективніших інструментів аналізу даних є рекомендаційні системи, які індивідуалізують пропозиції товарів або послуг, що підвищує рівень задоволеності користувачів та економічну ефективність онлайн-платформ [2]. У основі рекомендаційних систем лежать алгоритми машинного навчання, здатні знаходити зв’язки між транзакціями та прогнозувати уподобання користувачів [3].

2. МАТЕРІАЛИ ТА МЕТОДИ

2.1. Алгоритми пошуку асоціативних правил

Завданням пошуку асоціацій є виявлення набору атрибутів, спільних для великої кількості об'єктів у базі даних. Завдання виявлення всіх частих асоціацій у дуже великих базах даних є досить складним. Простір пошуку є експоненціально залежним від кількості атрибутів бази даних, а при мільйонах об'єктів бази даних проблема мінімізації вводу/виводу стає першочерговою [4].

Алгоритм Apriori є одним з найпоширеніших алгоритмів для пошуку асоціативних правил. Він працює, спочатку визначаючи часті набори елементів у наборі даних (набори елементів, які з'являються у певній кількості транзакцій). Потім він використовує ці часті набори елементів для генерації правил асоціації, які є твердженнями типу "якщо товар А придбано, то, ймовірно, буде придбано і товар Б". Алгоритм Apriori використовує висхідний підхід, починаючи з окремих товарів і поступово переходячи до більш складних наборів товарів.

Алгоритм Partition логічно розбиває горизонтальну базу даних на ряд розділів, що не перетинаються. Кожен розділ зчитується, і для кожного елемента формуються вертикальні tid-списки, тобто список всіх тайтлів, де з'являється даний елемент. Потім генеруються всі локально часті набори елементів за допомогою перетинів tid-листів. Всі локально часті набори елементів об'єднуються і виконується другий прохід по всіх розділах. База даних знову перетворюється на вертикальну і отримуються глобальні підрахунки всіх вибраних наборів елементів.

Розмір розділу вибирається таким чином, щоб він можна було розмістити в оперативній пам'яті. Таким чином, розділ виконує лише два сканування бази даних. Ключове спостереження що використовується, полягає у тому, що глобально частий набір елементів повинен бути локально частим принаймні в одному розділі. Таким чином, всі часті набори елементів гарантовано буде знайдено [4].

У роботі також буде розглянуто нові алгоритми пошуку асоціативних правил, а саме Eclat (Equivalence CLAss Transformation), MaxEclat, Clique, MaxClique, TopDown та AprClique.

Eclat використовує відношення еквівалентності на основі префіксів θ1 разом з висхідним пошуком. Він перераховує усі набори елементів, що часто зустрічаються. MaxEclat використовує відношення еквівалентності на основі префіксів θ1 разом з гібридним пошуком. Він перераховує "довгі" максимально часті набори елементів, а також деякі не не максимальні.

Clique використовує псевдо еквівалентне відношення φ1 на основі максимальної групи разом з висхідним пошуком. Він перераховує всі часті набори елементів.

MaxClique використовує псевдо-кваліфікаційне відношення φ1 на основі максимальної групи разом з гібридним пошуком. Він перераховує вздовж максимально частих наборів елементів, та деякі не максимальні.

TopDown використовує псевдо-кваліфікаційне відношення φ1 на основі максимальної групи разом з пошуком зверху вниз.

AprClique використовує псевдо кваліфікаційне відношення на основі максимальної групи φ1. Однак, на відміну від алгоритмів, описаних вище, він використовує горизонтальні дані [4].

2.2. Алгоритми FP-росту

Як і алгоритм Apriori, алгоритм FP-Growth також використовується для частого пошуку шаблонів. Алгоритм FP-Growth або алгоритм частого зростання шаблонів є вдосконаленням алгоритму Apriori.

Евристика Apriori досягає хорошої продуктивності за рахунок (можливо, значного) зменшення розміру наборів кандидатів. Однак, у ситуаціях з великою кількістю частих шаблонів, довгих шаблонів або досить низьких мінімальних порогів підтримки, алгоритм, подібний до апріорного може страждати від наступних двох нетривіальних витрат:

- обробка великої кількості наборів-кандидатів вимагає великих витрат;
- багаторазове сканування бази даних і перевірка великого набору кандидатів на відповідність шаблону є нудною справою [5].

При використанні алгоритму Apriori для пошуку асоціативних правил потрібно сканувати набір даних про транзакції кілька разів. В алгоритмі зростання FP-Growth потрібно лише двічі просканувати набір даних.

Крім того, не потрібно генерувати набори-кандидати під час генерації наборів частих елементів.FP-дерево використовується для визначення частих наборів елементів. Таким чином, алгоритм FP-зростання допомагає нам виконувати частий пошук шаблонів з меншими обчислювальними ресурсами і навіть меншими витратами часу.

FP-зростання може бути описана як рекурсивна схема виключення: на кроці попередньої обробки видаляються всі елементи з транзакцій, які не є частими в індивідуальному вимірі, тобто візуально, тобто не з'являються в заданій користувачем мінімальній кількості транзакцій.

Потім обираються всі транзакції, які містять найменш частий елемент (найменш частий серед тих які є частими) і цей елемент видаляється. Після цього відбувається рекурсія для обробки отриманої скороченої (також відомої як спроектована) бази даних базу даних, пам'ятаючи, що набори елементів, знайдені в рекурсії мають видалений елемент у якості префікса.

При поверненні видаляється оброблений елемент з бази даних усіх транзакцій і алгоритм починається спочатку, тобто обробляється другий за частотою елемент і т.д. На цих кроках обробки використовується дерево префіксів, яке розширюється за рахунок зв'язками між гілками, використовується для швидкого пошуку транзакції, що містять заданий елемент, а також для видалення цього елемент з транзакцій після того, як він був оброблений [6].

Алгоритм FP-зростання зберігає фактичні транзакції з бази даних у триєдину структуру (дерево префіксів), і додатково зберігає заголовну таблицю, що містить всі елементи з їх підтримкою та початок зв'язаного списку, що проходить через всі транзакції, які містять цей елемент [7].

Основний трюк, який використовується у FP-зростанні, полягає у тому, що йому потрібно лише знайти всі одиночні набори елементів, що часто зустрічаються у заданій базі даних. Потім для кожного такого елемента створюється відповідну спроектовану базу даних, в якій знову ж таки, потрібно знайти лише (локальні) синглетні (локальні) набори частих елементів. Цей процес триває до тих пір, поки не залишиться жодного (локального) елементів не залишиться. Структура FP-дерева гарантує, що все це можна зробити ефективно.

Таким чином, FP-зростання неявно створює багато баз даних, представлених FP-деревами. Хороша новина полягає в тому, що всі ці набори даних (дерева) можна скоротити (обрізати) за допомогою техніки ExAnte. Так обрізане FP-дерево називається FP-бонсай [7].

3. ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ АЛГОРИТМІВ І ПОРІВНЯННЯ РЕЗУЛЬТАТІВ

У процесі реалізації даного програмного продукту були досліджені два набори даних: Ecommerce Dataset by Retailrocket та Groceries. Мета дослідження полягала у виявленні асоціативних зв'язків між товарами за допомогою кількох алгоритмів: FP-Growth, Apriori, MaxEclat, TopDown та MaxClique.

В результаті проведеного аналізу даних за допомогою різних алгоритмів асоціації було отримано важливі висновки про популярність та взаємозв'язки між товарами в обох наборах даних.

Алгоритми Apriori та FP-Growth показали однакові результати, виявивши часті набори елементів із підтримкою 213834: 0,006827 і 461686: 0,011008, тоді як MaxEclat ідентифікував набір із одним частим елементом (461686) із підтримкою 0,0110. MaxClique виявив значну різноманітність комбінацій у даних, а TopDown підтвердив свою ефективність, надавши результати, аналогічні Apriori та FP-Growth.

Таблиця 1. Результати застосування алгоритмів на наборі даних Ecommerce Dataset by Retailrocket.

Алгоритм	Часті набори	Підтримка
Apriori	(213834, 461686)	213834: 0.006827, 461686: 0.011008
FP-Growth	(213834, 461686)	213834: 0.006827, 461686: 0.011008
MaxEclat	{461686}	461686: 0.0110
MaxClique	Великий набір даних (119736, 420960, 190000, 384302, 248455, 33866...)	N/A (велика різноманітність товарів)
TopDown	(213834, 461686)	213834: 0.006827, 461686: 0.011008

Для другого набору даних використано алгоритми Apriori, FP-Growth, MaxEclat, MaxClique, AprClique. Максимальне значення метрики Support (0,074) вказує на популярність комбінації "other vegetables" і "whole milk". Нижчі значення підтримки (<0,03) свідчать про рідкісні асоціації, наприклад, "soda, whole milk, yogurt" із підтримкою 0,010473.

Таблиця 2. Результати застосування алгоритму FP-Growth на наборі даних Groceries.

Передумови	Наслідки	Показник підтримки передумов	Показник підтримки наслідків	Показник підтримки	Показник впевненості	Показник Lift	Показник Leverage	Показник Conviction
root vegetables, citrus fruit	other vegetables	0,017692	0,193493	0,010371	0,586207	3,029608	0,006948	1,949059
other vegetables, yogurt	whole milk	0,043416	0,255516	0,022267	0,512881	2,007235	0,011174	1,52834
tropical fruit, yogurt	whole milk	0,029283	0,255516	0,01515	0,517361	2,02477	0,007668	1,542528
tropical fruit, root vegetables	other vegetables	0,021047	0,193493	0,012303	0,584541	3,020999	0,008231	1,941244
tropical fruit, root vegetables	whole milk	0,021047	0,255516	0,011998	0,570048	2,230969	0,00662	1,731553

П’ять правил асоціації демонструють зв’язки між товарами. Наприклад, правило (root vegetables, citrus fruit) → (other vegetables) має підтримку 0,010371, впевненість 58,62%, підйом 3,03, що свідчить про високий зв’язок між цими товарами.

Усі правила мають позитивні значення левереджу, conviction (>1) і метрики Жанга, що підтверджує їхню значущість та корисність для аналізу. Найпопулярнішими товарами залишаються "whole milk" та його комбінації.

Таблиця 3. Результати застосування алгоритму MaxEclat на наборі даних Groceries.

Набір елеметів	Значення показника Support	Елементи зі значенням покзаника Lift
whole milk, other vegetables	0,0748	whole milk: 6.9434, other vegetables: 5.2571
rolls/buns, whole milk	0,0566	rolls/buns: 5.5700, whole milk: 5.2547
yogurt, whole milk	0,056	yogurt: 4.5164, whole milk: 5.1981
root vegetables, whole milk	0,0489	root vegetables: 4.2193, whole milk: 4.5377
root vegetables, other vegetables	0,0474	root vegetables: 4.0877, other vegetables: 3.3286
yogurt, other vegetables	0,0434	yogurt: 3.5000, other vegetables: 3.0500
rolls/buns, other vegetables	0,0426	rolls/buns: 4.1900, other vegetables: 2.9929
tropical fruit, whole milk	0,0423	tropical fruit: 3.7143, whole milk: 3.9245
whole milk, soda	0,0401	whole milk: 3.7170, soda: 3.2295
rolls/buns, soda	0,0383	rolls/buns: 3.7700, soda: 3.0902

Для алгоритму MaxEclat з мінімальною підтримкою 0,01 було виявлено 333 часті набори, з яких 202 є максимальними. Найчастіші комбінації, такі як {whole milk, other vegetables} із підтримкою 7,48% і метрикою Lift 'whole milk' – 6,9434, 'other vegetables' – 5,2571, свідчать про сильну кореляцію між товарами, які часто купуються разом. Видалення одноелементних наборів дозволяє точніше визначити реальні зв’язки між продуктами, наприклад, {root vegetables, whole milk, other vegetables} із підтримкою 2,32%. Такі дані є корисними для оптимізації розташування товарів у магазині та маркетингових стратегій.

4. ВИСНОВКИ

Проаналізовано алгоритми FP-Growth, Apriori, MaxEclat, TopDown та MaxClique на двох наборах транзакційних даних для виявлення асоціативних зв'язків між товарами. Використання алгоритмів FP-Growth та MaxEclat показало переваги в обробці великих баз даних завдяки меншій кількості проходів по даних. Експериментально підтверджено, що алгоритм FP-Growth працює значно швидше та ефективніше порівняно з Apriori, що робить його більш придатним для аналізу великих обсягів транзакцій. Отримані результати можуть бути використані для оптимізації стратегій продажів та маркетингових кампаній.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Chen, M., Mao, S., & Liu, Y. Big Data: A Survey // Mobile Networks and Applications. 2014. Vol. 19, No. 2. P. 171-209.

2. Jannach, D., & Adomavicius, G. Recommendation systems: Challenges, insights, and research opportunities // AI Magazine. 2016. Vol. 37, No. 3. P. 67-78.

3. Aggarwal, C. C. Recommender Systems: The Textbook. Springer, 2016.

4. Zaki, M. J. (2000). Scalable algorithms for association mining. IEEE Trans. on Knowl. and Data Eng., 12, 372–390.

5. Han, J., Pei, J., Yin, Y. Mining Frequent Patterns without Candidate Generation // School of Computing Science, Simon Fraser University.

6. Cheung, Y.-L., & Fu, A. W.-C. (2004). Mining frequent itemsets without support threshold: with and without item constraints. IEEE Transactions on Knowledge and Data Engineering, 16, 1052–1069

7. Borgelt, C. (2005). An implementation of the fp-growth algorithm. In Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations (pp. 1–5). ACM.

ПОРОДЖУВАЛЬНІ МОДЕЛІ ТА МЕТОДИ ГЛИБОКОГО НАВЧАННЯ У ЗАДАЧІ SISR

Ланько А.А.[1], Недашківська Н.І.[2]

Національний технічний університет України «Київський політехнічний інститут ім. Ігоря Сікорського»

[1] lanko.anna@lll.kpi.ua, [2] nedashkovskaya.nadezhda@lll.kpi.ua [0000-0002-8277-3095]

Розглядається розв'язання задачі SISR за допомогою породжувальних моделей та методів глибокого навчання. Досліджено різні архітектури відомих моделей збільшення роздільності зображень, здійснено власну реалізацію моделей SRGAN, VDSR та DRCN. Виконано порівняння результатів на основі значень метрик та візуальної оцінки якості породжених зображень для деяких зразків тестової вибірки. Розроблено програмний продукт для збільшення роздільності зображень.

Ключові слова: SISR, породжувальні моделі, методи глибокого навчання, SRGAN, VDSR, DRCN.

1. ВСТУП

Задача суперроздільності зображення (англ. Single Image Super Resolution, SISR) передбачає відтворення високо деталізованої версії деякого зображення низької роздільної здатності [1]. Незважаючи на доступність та технічні можливості сучасної апаратури для зйомки, задача досі є актуальною через наступні проблеми:

- погіршення якості зображень при передачі через канали зв'язку;
- стиснення зображень для зберігання на інших носіях інформації з обмеженнями якості;
- апаратні збої;
- нестачу якісних зображень у певних галузях з утрудненими можливостями використання якісної знімальної апаратури.

Цільові суперроздільні зображення можна отримати як за допомогою породжувальних моделей, які генерують зразки з новими деталями на основі змодельованого розподілу даних, так і методів глибокого навчання, які розпізнають важливі ознаки у зображеннях для покращення деталізації [2]. Прикладами породжувальних моделей є генеративні змагальні мережі (GAN), варіаційний автокодувальник (VAE), дифузійні та потокові (Flow-based) моделі, прикладами методів глибокого навчання – згорткові (CNN), рекурентні (RNN) та залишкові (ResNet) нейронні мережі (НМ) [3]. Слід зазначити, що деякі види НМ використовуються як частина архітектури породжувальних моделей, наприклад генератора та дискримінатора у GAN.

Навчання моделей збільшення роздільності відбувається на парах низько- та відповідних їм високороздільних зображень. Якість формування нового зображення визначається на тестовій вибірці шляхом проведення візуального експертного оцінювання та на основі метрик якості, що порівнюють суперроздільне та оригінального високороздільне зображення за такими критеріями, як: піксельна точність, рівень шуму, подібність витягнутих за допомогою НМ важливих ознак тощо. При визначенні оптимальної моделі слід також зважати на технічні показники, оскільки процес формування високороздільних зображень є часовитратним та потребує потужних обчислювальних ресурсів [4].

2. ПОСТАНОВКА ЗАДАЧІ

Нехай I_{LR} – вихідне зображення низької роздільної здатності, I_{HR} – відповідне йому зображення високої роздільної здатності. Тоді задачу SISR можна представити у вигляді відображення

$$f: I_{LR} \rightarrow I_{HR}, \tag{1}$$

яке забезпечить максимально точне відновлення деталей зображення I_{HR} на основі інформації із зображення I_{LR}. Реалізацією відображення (1) є модель f_θ, де θ – параметри цієї моделі. Її виходом є цільове суперроздільне зображення

$$I_{SR} = f_\theta(I_{LR}).$$

Зазначимо, що (1) є узагальненим поняттям і позначає різні процеси в залежності від типу моделі, що його реалізує, наприклад: виділення ознак із зображення за допомогою згортки, відновлення розподілу реальних зображень тощо.

Оптимізаційна задача є типовою для більшості задач глибокого навчання і передбачає знаходження таких параметрів θ моделі, за яких значення функція втрат $\mathcal{L}$, що вимірює різницю між I_{HR} та I_{SR}, є мінімальною:

$$\min_\theta \mathcal{L}(I_{HR}, I_{SR}).$$

3. МЕТОДИ ТА МАТЕРІАЛИ

3.1. Набір даних

DIV2K [1] – це спеціалізований набір даних для SISR, зібраний для змагань NTIRE2017 і NTIRE2018 Super-Resolution Challenges. Він містить 1000 пар високо- та низькороздільних зображень, що зображують різноманітні сцени: 800 зразків у навчальній, по 100 – у валідаційній та тестовій вибірках.

Високороздільні зображення мають розмірність 2K, тобто довжину або висоту 2040 пікселів. Низькороздільні зображення можуть бути отримані з оригінальних високороздільних різними методами інтерполяції та додавання шуму. Для навчання моделей було обрано найбільш поширену версію DIV2K з використанням бікубічної інтерполяції для зменшення у 4 рази (x4) без додаткових спотворень.

Зображення у наборі було розбито на менші фрагменти сталого розміру з метою оптимізації обчислювальних ресурсів, що дозволяє швидше навчати моделі, та підвищення стійкості моделі завдяки використанню локальних текстур та більшої кількості навчальних даних.

3.2. Опис моделей

У результаті дослідження методів розв’язання SISR було обрано наступні моделі для власної реалізації.

1. **SRGAN.** Генеративна змагальна мережа для задачі збільшення роздільної здатності, де генератор створює суперроздільні зображення з низькороздільних, а дискримінатор навчається розрізняти реальні зображення та згенеровані [5].

2. **VDSR.** Глибока ЗНМ для збільшення роздільної здатності, яка використовує залишкові з'єднання для зменшення різниці між вхідним зображенням та суперроздільним результатом. Завдяки глибині VDSR захоплює більше інформації про структуру та текстури, а залишкове навчання (residual learning) допомагає уникнути проблеми з затуханням градієнта, прискорюючи збіжність [6].

3. **DRCN.** Глибока РНМ, використовує рекурсивні згорткові блоки, що багаторазово застосовують однаковий набір параметрів для збільшення глибини. Це дозволяє мережі ефективно відновлювати деталі високороздільного зображення, мінімізуючи кількість параметрів, тобто складність мережі [7].

Опис програмно реалізованих класів із зазначенням архітектури кожної з вищеописаних моделей наведено у таблиці 1.

Таблиця 1. Опис програмно реалізованих класів із зазначенням архітектури моделей

Клас	**Опис**
ResidualBlock	Блок залишкового з'єднання, що додає вхідні дані до вихідних для підтримки стабільності градієнтів. Містить 2 згорткові шари 3×3, шар нормалізації (BatchNorm2d) після кожного згорткового та функцію активації PReLU після 1-го шару
RecursiveBlock	Блок із 1 згортковим шаром 3×3 і активацією ReLU, що може рекурсивно повторюватися для створення складної архітектури
Generator_SRResNet	Генератор SRGAN з архітектурою SRResNet, складається з початкового згорткового шару 9×9, 5 залишкових блоків (ResidualBlock), проміжного згорткового блоку 3×3, блоку підвищення роздільної здатності (2 згорткові шари 3×3 з PixelShuffle), та фінального згорткового шару 9×9
Discriminator	Дискримінатор для SRGAN, складається з 8 згорткових шарів 3×3 із збільшенням кількості каналів з нормалізацією (BatchNorm2d) і активацією LeakyReLU, 1 шару адаптивного усереднення та 2 фінальних повнозв'язних шарів. Розмір фільтру для всіх згорткових шарів – 3×3
VDSR	Складається з початкового згорткового шару, 18 згорткових шарів з активацією ReLU, та вихідного шару, який додає залишок до вхідного зображення. Розмір фільтру для всіх згорткових шарів – 3×3
DRCN	Складається з вхідного згорткового шару, рекурсивного блоку (RecursiveBlock), який повторюється задану (16) кількість разів, і вихідного згорткового шару. Розмір фільтру для всіх згорткових шарів – 3×3

4. АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

Для оцінки якості моделей застосовувався наступний набір метрик, який дозволяє комплексно оцінити як структурні аспекти, так і візуальне сприйняття зображень [8].

1. **PSNR** (Peak Signal-to-Noise Ratio) – кількісна метрика, що вимірює різницю між пікселями пари зображень як відношення між максимальною можливою потужністю сигналу та шумом.

2. **SSIM** (Structural Similarity Index Measure) – перцептивна метрика, що порівнює закономірності яскравості, контрасту та структури у локальних вікнах для пари зображень.

3. **LPIPS** (Learned Perceptual Image Patch Similarity) – перцептивна метрика, яка використовує глибокі нейронні мережі для оцінки візуальної схожості пари ознак на основі витягнутих ознак.

Всі метрики лежать у діапазоні значень [0; 1]. Практично прийнятними для SISR можна вважати значення PSNR > 20, SSIM > 0.7, LPIPS > 0.3, проте ці значення мають бути не гіршими за результат для зворотної бікубічної інтерполяції, коли низькороздільне зображення повертають до масштабів високороздільного.

З огляду на часовитратність процесу навчання (використовувався апаратний прискорювач Nvidia GeForce RTX 4060), до критеріїв оцінки результату додається час навчання моделі. Для обґрунтування практичної значущості дослідження до аналізу результатів також залучено значення метрик, отриманих авторами статті [9].

Оптимізацію процесу навчання представлено в залежності від типу моделі: для SRGAN (рис. 1) розглядається зміна значень функцій втрат генератора (сума середньоквадратичної

похибки MSE і бінарної-крос-ентропії BCE) та дискримінатора (BCE), для VDSR та DCRN (рис. 2) – значень PSNR. Значення метрик для всіх моделей наведено у таблиці 2.

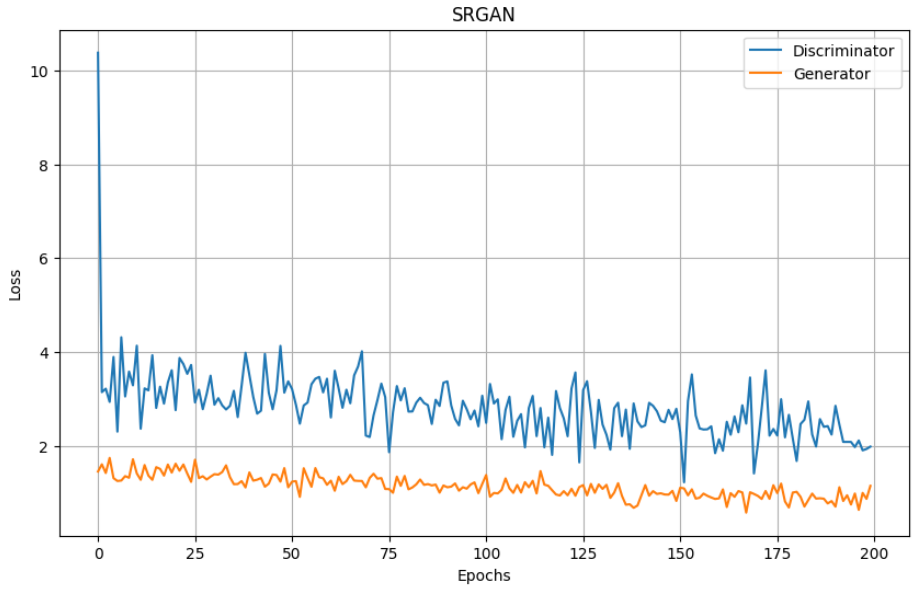


Рисунок 1. Зміна значень функції втрат у процесі навчання SRGAN

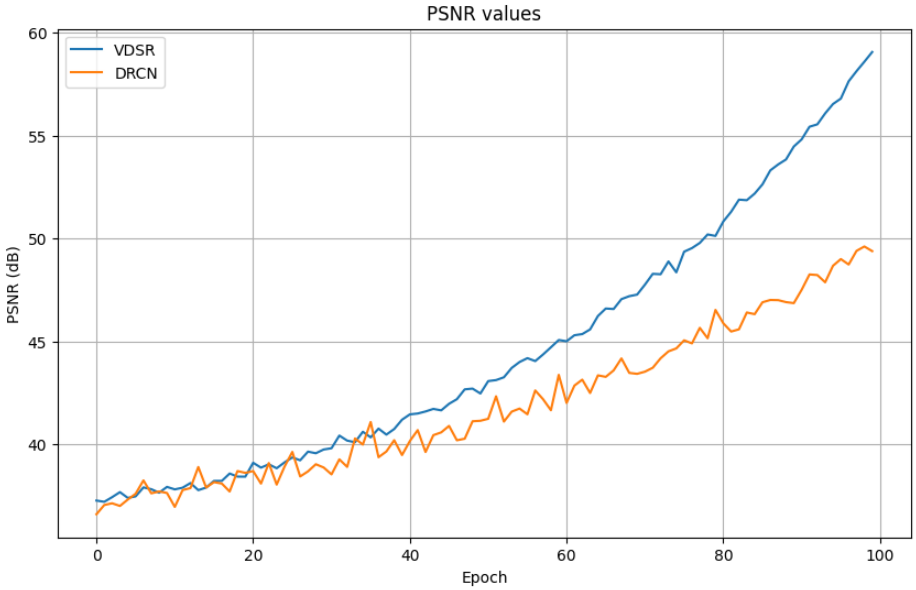


Рисунок 2. Зміна значень PSNR у процесі навчання VDSR та DRCN

Таблиця 2. Результати метрик для власної реалізації моделей, збільшення зображень x4

Модель	**Метрика**			**Час навчання (год.)**
	PSNR↑	SSIM↑	LPIPS↓	
Bicubic	25.80	0.74	0.46	-
SRGAN	24.50	0.71	0.33	32
VDSR	26.73	0.77	0.31	16
DRCN	26.41	0.76	0.37	25

Значення у таблиці 3 показують, що найкращою серед розглянутих є модель VDSR. Для всіх моделей вони є задовільними і знаходяться посередині інтервалу між мінімальним практично прийнятним значенням та порогом високої якості генерації. На рисунку 1 не спостерігається чітка динаміка зростання втрат дискримінатора, проте помітно спадання втрат генератора, що пояснюється потребою створювати більш глибоку архітектуру та збільшувати кількість епох навчання. Рисунок 2 зображує стрімке зростання значень PSNR, які стають значно слабшими при перевірці результату на тестовій вибірці, що свідчить про схильність цих моделей до перенавчання. Утім, у порівнянні з результатами авторів статті [9] та з огляду на можливий час навчання моделей суперроздільності до кількох діб [2], власний результат можна вважати позитивним.

Візуальне порівняння вихідних низькороздільних (LR) із згенерованими суперроздільними (SR) та оригінальними високороздільними (HR) зображеннями наведено на рисунку 3.

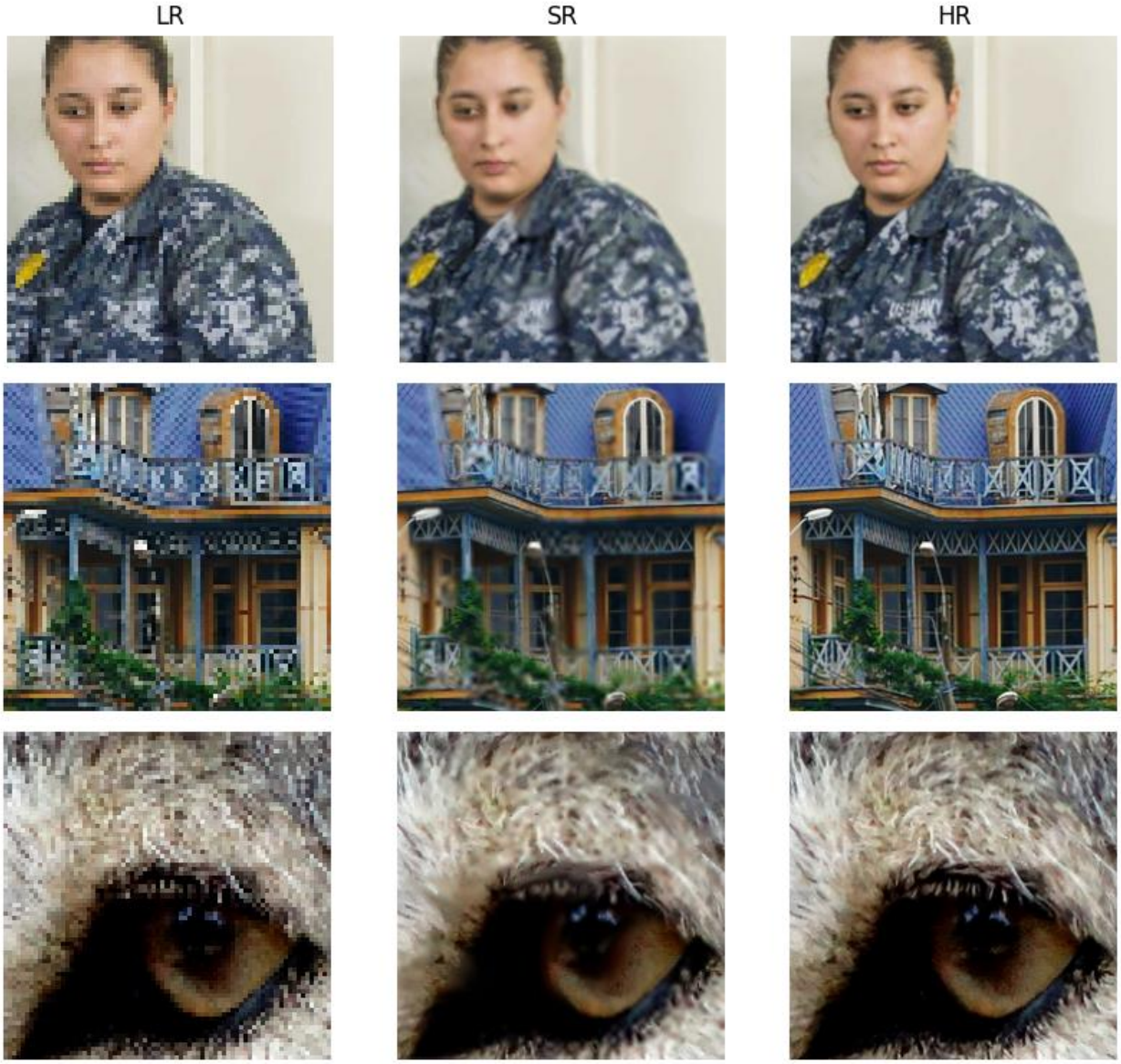


Рисунок 3. Візуальне порівняння отриманих SR-зображень з високороздільним оригіналом та низькороздільним вихідним зображенням

Збільшення роздільності успішно виконується для різних форм, текстур та кольорових поєднань, SR-результат є суттєво кращим за LR-зображення, що важко сприймається через явно помітні пікселі, проте його можна відрізнити від HR-зображення за помірною розмитістю. Ймовірне перенавчання моделі VDSR не спостерігається візуально.

5. ВИСНОВКИ

Здійснено власну реалізацію породжувальної моделі SRGAN і глибоких мереж VDSR та DRCN, створено програмний продукт для збільшення роздільної здатності зображень із застосуванням навчених на наборі даних DIV2K моделей.

Виявлено оптимальну серед розглянутих модель – VDSR – з точки зору значень кількісних та перцептивних метрик, а також часу. Підтверджено відсутність перенавчання візуальну якість сформованих нею суперроздільних зображень на відібраних зразках тестової вибірки набору, що зображують різні форми, текстури та поєднання кольорів.

Методи глибокого навчання в проведених експериментах мали перевагу над породжувальними моделями в умовах компромісу між складністю архітектури моделі та часом навчання. Зокрема було показано ефективність застосування залишкового навчання для задачі SISR через відносно швидку збіжність.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Agustsson E., Timofte R. NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study. 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017. URL: https://doi.org/10.1109/cvprw.2017.150

2. NTIRE 2017 Challenge on Single Image Super-Resolution: Methods and Results / R. Timofte et al. 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017. URL: https://doi.org/10.1109/cvprw.2017.149.

3. Wang Z., Chen J., Hoi S. C. H. Deep Learning for Image Super-resolution: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2020. P. 1. URL: https://doi.org/10.1109/tpami.2020.2982166.

4. Ausare T. Ultimate Guide to Selecting a GPU for Deep Learning. Latest AI, ML & GPU Updates | NeevCloud. URL: https://blog.neevcloud.com/ultimate-guide-to-selecting-a-gpu-for-deep-learning.

5. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network / C. Ledig et al. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 21–26 July 2017. 2017. URL: https://doi.org/10.1109/cvpr.2017.19.

6. Kim J., Lee J. K., Lee K. M. Accurate Image Super-Resolution Using Very Deep Convolutional Networks. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016. 2016. URL: https://doi.org/10.1109/cvpr.2016.182.

7. Kim J., Lee J. K., Lee K. M. Deeply-Recursive Convolutional Network for Image Super-Resolution. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016. 2016. URL: https://doi.org/10.1109/cvpr.2016.181.

8. Single Image Super-Resolution Quality Assessment: A Real-World Dataset, Subjective Studies, and an Objective Metric / Q. Jiang et al. IEEE Transactions on Image Processing. 2022. Vol. 31. P. 2279–2294. URL: https://doi.org/10.1109/tip.2022.3154588.

9. SRFlow: Learning the Super-Resolution Space with Normalizing Flow / A. Lugmayr et al. Computer Vision – ECCV 2020. Cham, 2020. P. 715–732. URL: https://doi.org/10.1007/978-3-030-58558-7_42.

РОЗРОБКА ПРОГРАМНОГО ДОДАТКУ ДЛЯ КОНСУЛЬТУВАННЯ ІНВЕСТОРІВ З ВИКОРИСТАННЯМ МЕТОДІВ ТЕКСТОВОЇ АНАЛІТИКИ ТА МАШИННОГО НАВЧАННЯ

Маринич А.Ю.[1], Кузнєцова Н.В.

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] marynych.anton0@gmail.com

В умовах зростаючого інтересу до інвестування та складності фінансових ринків, початківці-інвестори стикаються з проблемою доступу до якісних консультацій та навчальних матеріалів. Розробка програмного додатку може значно спростити процес входження на фінансові ринки та допомогти у прийнятті інвестиційних рішень. Результатом дослідження є створення програмного додатку, що використовує методи обробки природної мови та машинного навчання для надання персоналізованих інвестиційних рекомендацій.

Ключові слова: програмний додаток, інвестиційне консультування, машинне навчання, обробка природної мови.

1. ВСТУП

Інвестування є одним з ключових інструментів збереження та примноження капіталу в сучасному світі. З розвитком фінансових технологій та появою нових інвестиційних платформ доступ до різноманітних фінансових інструментів значно спростився, що призвело до зростання кількості непрофесійних інвесторів на ринку. Проте, відсутність достатніх знань та досвіду часто призводить до прийняття необгрунтованих інвестиційних рішень та фінансових втрат. Особливо гостро ця проблема постає у контексті криптовалютного ринку та інвестування в акції, де висока волатильність та складність аналізу створюють додаткові ризики для недосвідчених інвесторів.

В цих умовах розробка автоматизованої системи консультування у форматі додатку стає актуальним рішенням для демократизації доступу до якісних інвестиційних порад. Використання сучасних технологій обробки природної мови та машинного навчання дозволяє створити інтелектуального помічника, здатного надавати персоналізовані рекомендації, проводити базове навчання фінансової грамотності та допомагати у формуванні інвестиційних стратегій. Важливість розробки такої системи підкреслюється зростаючим попитом на автоматизовані рішення для фінансового консультування, особливо серед молодого покоління інвесторів.

2. СИСТЕМНИЙ ПІДХІД ДО РОЗРОБКИ ЧАТ-БОТА

Розробка ефективного додатку для консультування інвесторів вимагає комплексного підходу з урахуванням різних аспектів взаємодії з користувачем. Запропонована система відрізняється від існуючих рішень інтеграцією трьох ключових компонентів: модуля навчання основам інвестування, системи аналізу інвестиційного профілю користувача та модуля формування персоналізованих рекомендацій. Новизна підходу полягає у використанні

адаптивного алгоритму навчання, який автоматично коригує складність та обсяг матеріалу залежно від рівня користувача та його прогресу у вивченні інвестиційних інструментів.

Архітектура додатку розроблена з використанням сучасних підходів до створення програмних додатків, що забезпечує гнучкість та масштабованість рішення. Програмний додаток включає в себе розроблений метод оцінки інвестиційного профілю користувача, який враховує не лише стандартні параметри (вік, дохід, досвід інвестування), але й вказані користувачем параметри. Це дозволяє формувати більш точні та релевантні рекомендації порівняно з традиційними підходами консультування.

Розроблений інтерфейс користувача забезпечує інтуїтивно зрозумілу навігацію та доступ до всіх функцій системи, включаючи освітні матеріали, аналітичні інструменти та персоналізовані рекомендації.

Для забезпечення актуальності інформації реалізовано інтеграцію з зовнішніми джерелами даних про фінансові ринки. Це дозволяє користувачам отримувати актуальну інформацію про стан ринків, курси криптовалют та акцій, що є важливим для прийняття обґрунтованих інвестиційних рішень. Система також включає модуль візуалізації даних, який допомагає користувачам краще розуміти ринкові тренди та аналізувати історичні дані.

3. РЕЗУЛЬТАТИ РОЗРОБКИ ТА ТЕСТУВАННЯ ДОДАТКУ

В результаті розробки було створено програмний додаток у форматі Telegram бота, архітектура якого представлена на рисунку 1. Система складається з кількох ключових рівнів: рівень користувацького інтерфейсу (Telegram), рівень додатку, сервісний рівень та рівень доступу до даних. Кожен рівень відповідає за окремий функціонал та забезпечує модульність і масштабованість системи.

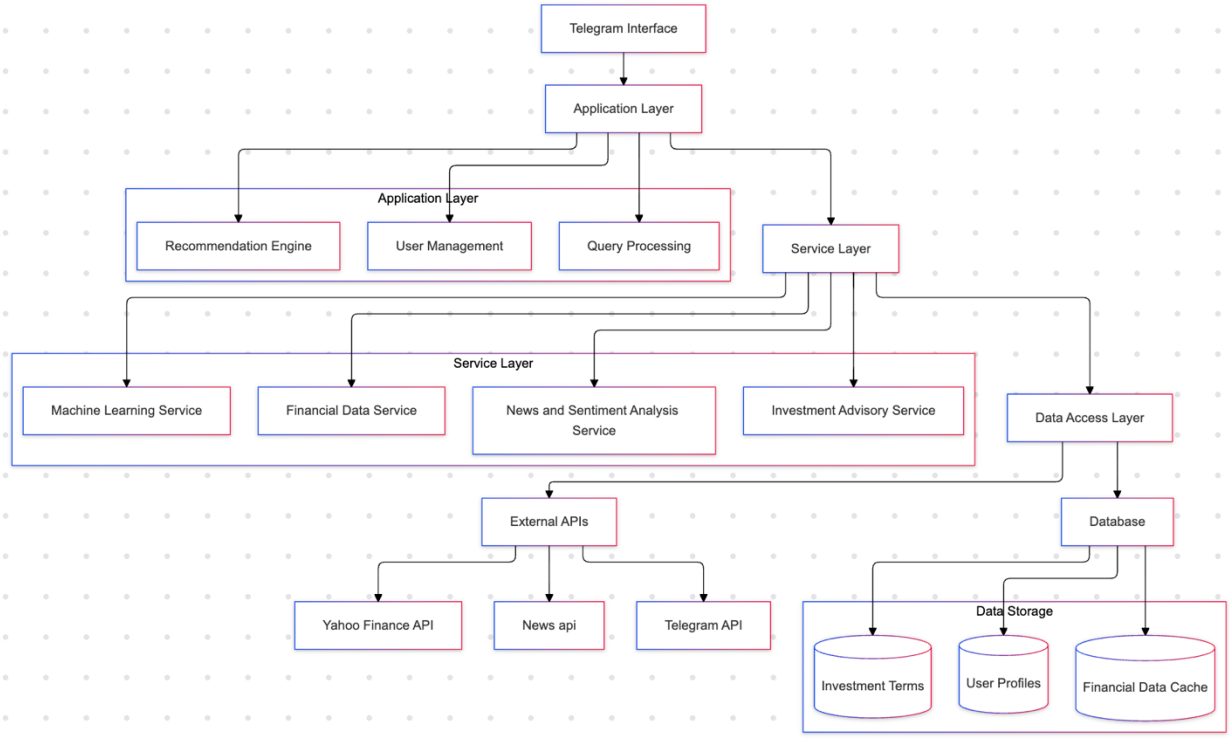


Рисунок 1. Архітектура додатку

Для отримання даних про фінансові ринки було використано API Yahoo Finance, що дозволяє отримувати актуальну інформацію про курси акцій та криптовалют. На рисунку 2 представлено приклад відображення фінансових даних через інтерфейс бота, включаючи графіки зміни цін та основні ринкові показники.

Рисунок 2. Приклад відображення фінансових даних

Розроблений додаток надає користувачам повний набір команд для взаємодії з системою:

- "/help" – отримання інформації про доступний функціонал;
- "/create_profile" – створення інвестиційного профілю користувача;
- "/price <тікер>" – отримання поточної ціни активу;
- "/history <тікер>" – отримання історичних даних;
- "/analyze <тікер>" – отримання персоналізованого аналізу активу;
- "/risk <тікер>" – оцінка ризиків інвестування.

Крім основного функціоналу додатку, він включає в себе створення користувацького профілю. Процес включає покрокове заповнення даних щодо:

- досвіду користувача в інвестуванні;
- фінансових цілей;
- толерантності до ризику за шкалою від 1 до 10;

Важливим компонентом системи є модуль аналізу та формування рекомендацій, який враховує:

- історичні дані щодо ціни активу;
- поточні ринкові тренди;
- інвестиційний профіль користувача;
- загальну ринкову ситуацію.

Тестування додатку проводилось комплексно та включало перевірку:

- коректності обробки користувацьких запитів через Telegram API;
- точності отриманих даних з Yahoo Finance;
- правильності роботи модуля аналізу та рекомендацій;
- стабільності роботи системи при різних сценаріях використання;
- швидкодії обробки запитів.

На рисунку 3 представлено приклад роботи системи аналізу, який демонструє формування персоналізованих рекомендацій з урахуванням профілю користувача.

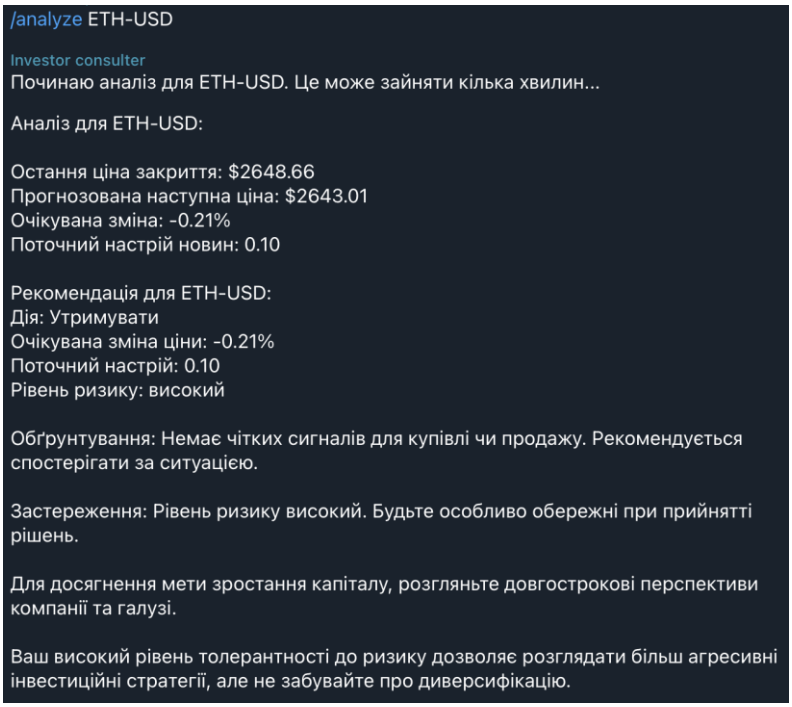


Рисунок 3. Приклад персоналізованого аналізу активу

Особлива увага при розробці була приділена модулю оцінки ризиків, який надає користувачам детальний аналіз потенційних ризиків при інвестуванні в конкретний актив. Цей модуль враховує волатильність активу, історичні дані та загальні ринкові тенденції для формування комплексної оцінки ризиків.

Результати тестування підтвердили:

- стабільну роботу всіх компонентів системи;
- ефективну взаємодію між різними модулями;
- точність наданих рекомендацій;
- зручність користувацького інтерфейсу;
- надійність зберігання та обробки даних.

Розроблений додаток демонструє високу ефективність у вирішенні поставлених задач та має потенціал для подальшого розширення функціоналу відповідно до потреб користувачів.

4. ВИСНОВКИ

У сучасному світі інвестиційні ринки стають все складнішими та менш передбачуваними, що породжує потребу у розробці більш ефективних інструментів для консультування інвесторів та допомоги у прийнятті інвестиційних рішень.

У ході дослідження було розроблено програмний додаток, який забезпечує можливість отримання консультацій та рекомендацій щодо інвестування в акції та криптовалюти. Система реалізована у форматі Telegram-бота, що забезпечує зручний доступ для користувачів, та використовує інтеграцію з Yahoo Finance API для отримання актуальних ринкових даних. У додатку реалізовано функціонал для створення користувацького профілю, отримання актуальних цін та історичних даних, проведення аналізу активів та оцінки ризиків інвестування. Результати тестування системи підтвердили ефективність обраного підходу та показали високу точність наданих рекомендацій.

Розроблений додаток демонструє потенціал для подальшого розвитку та масштабування, а також може бути використаний як основа для створення більш комплексних систем інвестиційного консультування.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Chatbots: Fundamentals and Applications. By Sumit Raj, 2023. 380 p.
2. Natural Language Processing in Action. By Lane, Howard, Hapke, 2021. 392p.
3. Python documentation. URL: https://docs.python.org/. [last accessed 11 November 2024].
4. Journal of Financial Technology, Chatbots in Investment Advisory: User Experience and Adoption, 2023. 245 p.
5. TensorFlow documentation. URL: https://www.tensorflow.org/api_docs [last accessed 11 November 2024].
6. NLTK documentation. URL: https://www.nltk.org/ [last accessed 11 November 2024].
7. Telegram Bot API documentation. URL: https://core.telegram.org/bots/api [last accessed 11 November 2024].
8. Scikit-learn documentation. URL: https://scikit-learn.org/stable/ [last accessed 11 November 2024].
9. yfinance documentation. URL: https://pypi.org/project/yfinance/ [last accessed 11 November 2024].
10. Hugging Face Transformers documentation. URL: https://huggingface.co/docs/transformers/ [last accessed 11 November 2024].
11. SQLite documentation URL: https://www.sqlite.org/docs.html [last accessed 11 November 2024].

ПРОГНОЗУВАННЯ НЕСТАЦІОНАРНИХ ФІНАНСОВИХ ПРОЦЕСІВ В УМОВАХ ІНФОРМАЦІЙНОЇ ВОЛАТИЛЬНОСТІ

Митник О.Ю.[1], Бідюк П.І.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] oleg.mytnyk@gmail.com, [2] pbidyuke_00@ukr.net

Метою дослідження є аналіз гаусівських процесів як непараметричного методу машинного навчання з вчителем для побудови регресійної моделі нестаціонарих фінансових процесів в умовах інформаційної волатильності. Показано, що викривлення вхідного часу-простору відповідно до рівня волатильності додає нестаціонарність в функцію коваріації і покращує прогнозуючі властивості регресії гаусівського процесу. В якості прикладу досліджена динаміка курсу акцій GME в період її сильної волатильності.

Ключові слова: гаусівський процес, нестаціонарність, функція коваріації, волатильність, викривлення простору.

1. ВСТУП

Прогнозування фінансових процесів у сучасних умовах є складним завданням через високу волатильність і нестаціонарність ринкових даних. Фінансові ринки постійно змінюються під впливом економічних, політичних та соціальних чинників, що породжує значні коливання у вартості активів. Іноді економічні та політичні події ще можна передбачити, на відміну від інформаційних ефектів, коли повідомлення окремих осіб або події у соціальних мережах призводять до раптових змін вартості активів. У таких умовах традиційні методи регресії часто виявляються недостатньо гнучкими для побудови середньострокових і довгострокових прогнозів, оскільки вони не враховують динамічні зміни у часових рядах і, як правило, передбачають стаціонарність процесу.

Одним з перспективних підходів для моделювання нестаціонарних часових рядів є регресія гаусівських процесів (ГП), яка надає можливість моделювати складні нелінійні залежності та оцінювати невизначеність прогнозів. Завдяки своїй здатності працювати із довільно розподіленими даними і використовувати інформацію про структуру кореляції, ГП є особливо корисним в умовах високої інформаційної волатильності, де ціни можуть різко змінюватися у відповідь на нову інформацію. На відміну від гетероскедастичних моделей, які адаптуються до зміни дисперсії часового ряду чим часто і компенсують нестаціонарність, ГП здатний адаптуватись до змін статичних характеристик ряду через нестаціонарні функції коваріації.

2. ПРИКЛАДИ ІНФОРМАЦІЙНОЇ ВОЛАТИЛЬНОСТІ

Наведемо декілька типових прикладів де політичні, економічні та соціальні події мали значний вплив на фінансову стабільність активів.

2.1. Інформаційна волатильність біткоїна

Реальність така, що ціна біткоїну може як зростати вибуховими темпами, так і стрімко падати [1]. Наприклад, напротязі травня 2021 року біткоїн втратив майже половину своєї вартості внаслідок жорсткого обвалу, викликаного двома серйозними ударами. Першим було

повідомлення від Ілона Маска, генерального директора Tesla. В середині травня він оголосив, що не прийматиме біткоїни як засіб оплати за свої автомобілі через забруднення довкілля, яке, на його думку, викликає майнінг криптовалют. Другий удар біткоїну завдали кількома днями пізніше, коли уряд Китаю запровадив нові правила для транзакцій з криптовалютами.

2.2. Інформаційна волатильність акцій GameStop

GameStop – американська роздрібна компанія, що спеціалізується на продажу відеоігор, ігрових приставок та аксесуарів. У минулому компанія була лідером на ринку продажу відеоігор, але з часом, через розвиток цифрової дистрибуції та онлайн-платформ (таких як Steam та PlayStation Store), її популярність і прибутки почали різко зменшуватися. GameStop зіткнулася з труднощами: закриття магазинів, зниження продажів, падіння ринкової вартості та збитки. Багато великих інституційних інвесторів і хедж-фондів почали робити ставку на те, що акції GameStop продовжуватимуть падати. Вони масово використовували стратегію коротких продажів (short selling), продаючи акції компанії, яких вони не мали, у надії купити їх пізніше за нижчою ціною і отримати прибуток. У певний момент більше ніж 100% вільно торгованих акцій GameStop були взяті в позицію на короткі продажі, що зробило компанію надзвичайно вразливою до раптового зростання ціни через масове закриття коротких позицій.

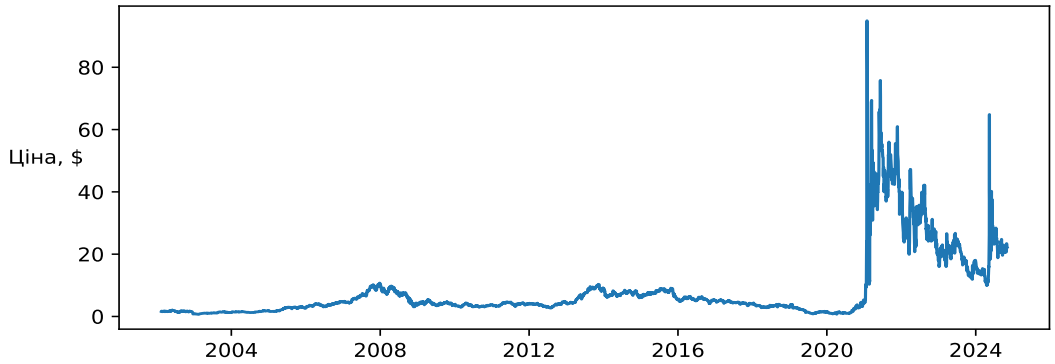


Рисунок 1. Динаміка ціни акції GameStop Corp. (GME)

Користувачі соціальної мережі Reddit організували скоординовані покупки акцій GameStop, що призвело до їх стрімкого зростання [2]. Спільнота сприймала це як можливість не тільки отримати прибуток, але й покарати хедж-фонди, які масово ставили на падіння акцій компанії. Коли ціна акцій GameStop почала зростати, хедж-фонди, які були у коротких позиціях, зазнали величезних збитків. Вони змушені були викуповувати акції, щоб закрити свої позиції, що додатково штовхало ціну ще вище. Всього за кілька днів у січні 2021 року ціна акцій GameStop піднялася з близько 20 до понад 80 доларів США за акцію (рис. 1).

3. МЕТА І МЕТОДИ ДОСЛІДЖЕННЯ

Ми навели декілька відомих прикладів коли зовнішні політичні або соціальні події призвели до раптових змін курсу фінансових активів. Детальний огляд математичних моделей оцінювання волатильності фінансових процесів можна знайти в роботі [3]. Зокрема, дуже популярною є модель стохастичної волатильності GARCH і її модифікації. Ця модель використовується для моделювання волатильності яка змінюється з часом і дає гарні коротко-строкові прогнози. Проте, при середньо-строковому прогнозуванні динаміки вартості активів з високою волатильністю важливою задачею є розрізняти гетероскедастичність від нестаціонарності. Гарні результати прогнозу можуть давати моделі, в структурі яких може бути закладена нестаціонарність. Зокрема нестаціонарні гаусівські процеси.

3.1. Стаціонарний гаусівський процес

Гаусівський процес – це стохастичний процес, який складається з набору випадкових величин, будь-яка підмножина яких має спільний багатовимірний гаусівський розподіл:

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')),$$

де $m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$ – функція середнього значення, $k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]$ – функція коваріації, яка ще називається ядром, оскільки вона виконує роль ядра в просторі функцій для вимірювання подібності між точками. Вибір ядра є критично важливим, оскільки воно визначає властивості моделі, такі як гладкість, періодичність, масштабування та здатність адаптуватися до певних особливостей даних. В роботі [4] наведені найбільш поширені типи ядер та деякі рекомендації щодо вибору ядра в залежності від типу даних та особливостей. Для моделювання стаціонарних процесів та аналітичного обчислення апостеріорного розподілу функцію коваріації вибирають з експоненційно-квадратичним ядром (відомим ще як радіально-базисна функція) вигляду:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}\ell^{-2}\|\mathbf{x} - \mathbf{x}'\|^2\right), \quad (1)$$

де σ_f^2 – це варіація процесу, ℓ – так звана довжина масштабу (length-scale). Ці параметри функції коваріації ще називають гіперпараметрами. Довжина масштабу описує, як швидко функція змінюється по відношенню до вхідних значень у гаусівському процесі. Вона визначає, наскільки далеко мають бути точки, щоб їхні значення стали некорельованими. На практиці ефективний прогноз будується на горизонті прогнозування (часовий проміжок, на який здійснюється прогнозування) в 2ℓ, де кореляція вже майже зникає.

3.2. Побудова прогнозуючої моделі

Маючі дані спостережень $\mathcal{D} = \{(\mathbf{x}_i, y_i): \mathbf{x}_i \in \mathbb{R}^d; y_i \in \mathbb{R}; i = 1, \dots, n\}$, як правило, розглядають адитивну модель нормального шуму: $y = f(\mathbf{x}) + \varepsilon$, де гомоскедастична помилка спостережень має нормальний розподіл $\mathcal{N}(\varepsilon|0, \sigma_n^2)$. Тоді правдоподібність (ймовірність породження даних моделлю) має вигляд:

$$p(\mathbf{y}|\mathbf{f}, \sigma_n^2) = \prod_{i=1}^n \mathcal{N}(y_i|f_i, \sigma_n^2) = \mathcal{N}(\mathbf{y}|\mathbf{f}, \sigma_n^2\mathbf{I}).$$

Вибираючи апріорний розподіл функцій як $p(\mathbf{f}|\mathbf{X}, \sigma_f^2, \ell) = \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K})$, відповідно до байєсівського висновку, апостеріорний розподіл має вигляд [5]:

$$p(\mathbf{f}|\mathcal{D}, \sigma_f^2, \ell, \sigma_n^2) \propto p(\mathbf{y}|\mathbf{f}, \sigma_n^2)p(\mathbf{f}|\mathbf{X}, \sigma_f^2, \ell) = \mathcal{N}(\mathbf{f}|\mathbf{K}(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y}, (\mathbf{K}^{-1} + \sigma_n^{-2}\mathbf{I})^{-1})$$

Елементи матриці коваріації $\mathbf{K}$ обчислюються попарно для всіх точок спостережень $\mathbf{K} = \{k_{ij} = k(\mathbf{x_i}, \mathbf{x_j}); i, j = 1, \dots, n\}$. Нарешті, апостеріорний прогнозний розподіл значень процесу $\mathbf{f}_*$ на тестових даних $\mathbf{X}_*$ можна отримати усереднюючи по всім $\mathbf{f}$:

$$p(\mathbf{f}_*|\mathcal{D}, \sigma_f^2, \ell, \sigma_n^2) = \mathcal{N}(\mathbf{f}_*|\mathbf{K}_*^{\mathrm{T}}(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y}, \mathbf{K}_{**} - \mathbf{K}_*^{\mathrm{T}}(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{K}_*). \quad (2)$$

Для знаходження оптимальних значень гіперпараметрів $\sigma_f^2, \ell, \sigma_n^2$, як правило, максимізують так зване підтвердження або маргінальну правдоподібність $p(\mathcal{D}|\sigma_f^2, \ell, \sigma_n^2)$ [6].

Стаціонарний гаусівський процес має постійні середнє та коваріацію, яка залежить лише від відстані між точками, що робить його корисним для моделювання процесів із однаковими властивостями в будь-якій точці простору або часу. Проте, ми розглядаємо моделювання процесів які змінюють свою поведінку і властивості з часом. Для цього, або шумову складову

роблять гетероскедастичною, або змінюють довжину масштабу залежно від точок в просторі. Або ж взагалі вибирають нестаціонарні ядра як, наприклад, в роботі [7].

3.3. Викривлення вхідного простору моделі гаусівського процесу

Зауважимо, що прогнозна невизначеність, тобто коваріаційна матриця $\mathbf{f}_*$ в формулі (2) не залежить від **y**. Це свідчення стаціонарності. Додати нестаціонарність до функції ядра можна так, щоб коваріація також залежала від **y**. Так, в роботі [8], застосовується додатковий апріорний розподіл, який задає викривлення вхідного простору.

Очевидно з формули (1), що збільшення довжини масштабу ℓ має той самий ефект на матрицю коваріації що і зменшення відстані між точками простору. Збільшення відстані між точками призводить до зменшення коваріації. Маємо інтуїтивну інтерпретацію для часових рядів: час біжить швидше в точках з більшою волатильністю. Враховуючи, що для часових рядів вхідний простір одновимірний: $\mathbf{x}\sim x\sim t$ і відображає час, отже в цьому випадку $x_{i+1} > x_i$, визначимо викривлення простору для даних спостережень (x_i, y_i) наступним чином:

$$\tilde{x}_1 = x_1, \qquad \tilde{x}_i = \tilde{x}_{i-1} + \frac{|y_i - y_{i-1}|}{y_{max} - y_{min}}(x_i - x_{i-1})$$

Оскільки для горизонту прогнозування у нас немає інформації по майбутнім значенням $\mathbf{y}_*$, то для викривлення простору нам потрібна оцінка амплітуди майбутніх змін: $|\Delta y_{horizon}|$. Цю оцінку можна отримати з останніх відомих значень амплітуди, а також із зовнішніх джерел таких як, наприклад, кількість згадок в соціальних мережах, інформація від аналітиків тощо.

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для демонстрації результатів дослідження було взято період сильної волатильності курсу акцій GME з січня по вересень 2021 року. Для тренування використано 137 днів, а для тесту останні 30 днів, що відповідає середньо-строковому прогнозу. Для того щоб оцінити ефект викривлення простору для горизонту прогнозування, використано реальні амплітуди. На рис. 2 показано яким чином викривлення простору впливає на матрицю коваріації **K** і робить її нестаціонарною, зменшуючи коваріацію в періоди сильнішої волатильності.

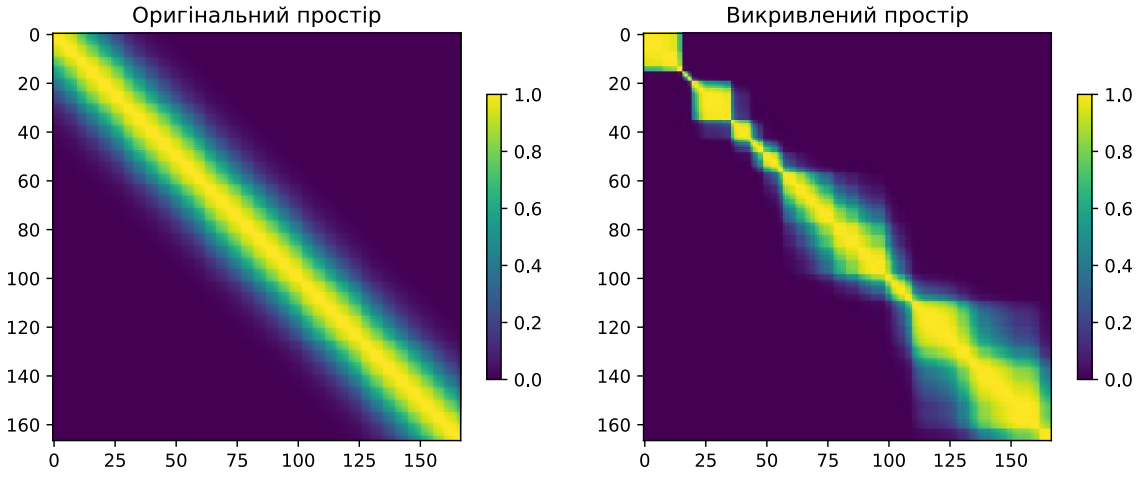


Рисунок 2. Вплив викривлення простору на матрицю коваріації. Нестаціонарність матриці коваріації з викривленим вхідним простором (справа).

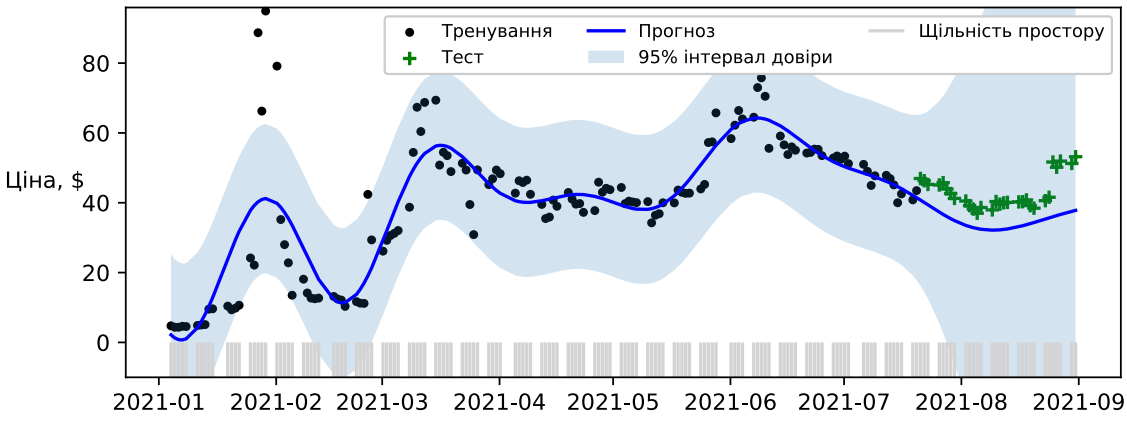


Рисунок 3. Стаціонарний гаусівський процес (RMSE=9.76). Довірчий інтервал на тренувальній вибірці відображає рівень нормованого гомоскедастичного шуму $\sigma_n^2 = 0.36$.

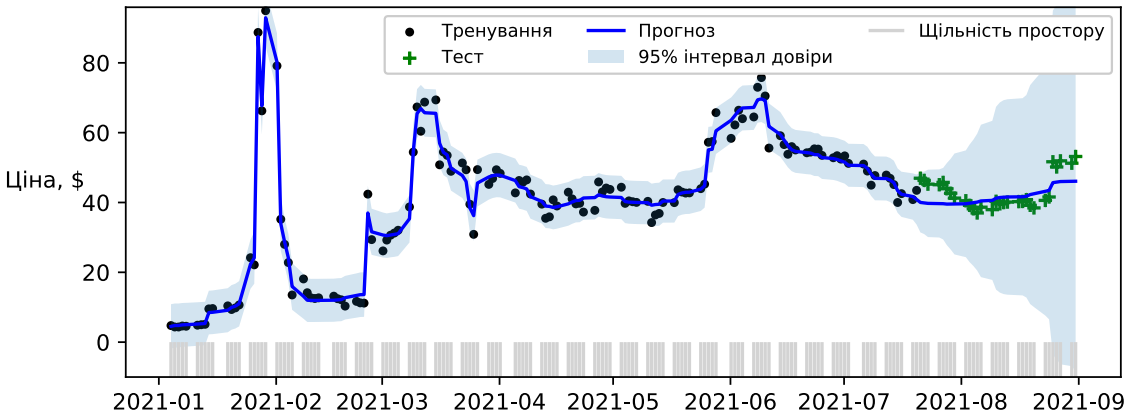


Рисунок 4. Нестаціонарний гаусівський процес з викривленням простору (RMSE=2.68). Видно менший рівень нормованого шуму $\sigma_n^2 = 0.03$ на тренувальній вибірці за рахунок нестаціонарності в структурі моделі гаусівського процесу (а саме в функції коваріації).

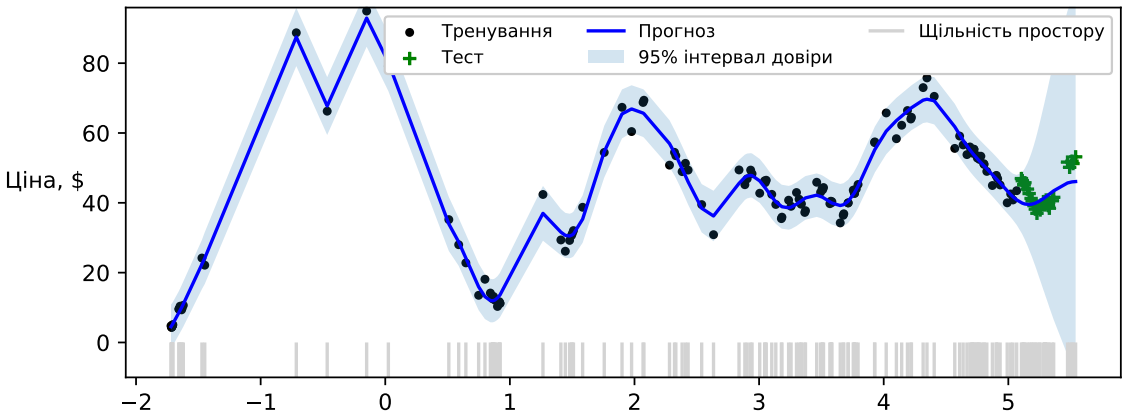


Рисунок 5. Нестаціонарний гаусівський процес з викривленням простору зображений у викривленому просторі, що відображено у нерівномірності щільності вхідного простору.

Очевидно, прогноз побудований з допомогою викривлення простору більш точний (середньо-квадратична похибка менше), інтервал довіри набагато вужчий і точний (рис. 4). Тоді як стаціонарна модель (рис. 3) більш гладка і компенсує нестаціонарність більшим рівнем оцінки гомоскедастичного шуму (саме через це довірчий інтервал набагато ширший). На рис. 5, зображено нормований і викривлений вхідний простір, який реально використовується гаусівським процесом для побудови регресії. В обох випадках довжина масштабу $\ell = 0.3$, що відповідає приблизно 21-му дню. Це означає, що адекватно прогнозується лише 21 день з 30 днів горизонту прогнозування, після цього періоду прогноз буде співпадати з апріорним розподілом. Тобто фактично ми отримали оцінку ефективного горизонту прогнозування.

5. ВИСНОВКИ

В роботі були розглянуті відомі приклади інформаційної волатильності фінансових активів, для яких складно побудувати прогноз. Особливо в періоди різкого росту або падіння, причиною яких є інформаційний або соціальний ефект. В таких випадках побудова прогнозу виключно на основі тренду часового ряду практично неможлива без якісної аналітики від експертів. Гаусівські процеси (ГП) є корисними для роботи з довільно розподіленими даними, оскільки вони дозволяють використовувати ядра для гнучкого моделювання складних залежностей і не вимагають знання точної форми розподілу. Це дозволяє ГП адаптуватися до будь-яких змін у структурі даних і враховувати невизначеність у прогнозах, що робить його ефективним інструментом для аналізу нестаціонарних процесів із довільним розподілом. Альтернативою до зміни ядра може бути викривлення вхідного простору відповідно до степеню волатильності. Також, це відкриває можливість залучити до викривлення простору інформацію про зростання кількості згадок в соціальних мережах, кількість твітів, прогнози інших аналітиків щодо змін волатильності тощо. В подальшій роботі ми розглянемо ширший спектр часових рядів і порівняємо результати з іншими сучасними методами прогнозування.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Битва за біткоїн. Хто є хто у світовому протистоянні через криптовалюти. URL: https://www.bbc.com/ukrainian/features-57526593 (дата звернення 20.06.2021).

2. Betzer A., Harries J.P. How online discussion board activity affects stock trading: the case of GameStop. *Financial Markets and Portfolio Management*. 2022. Vol. 36. P. 443–472. https://doi.org/10.1007/s11408-022-00407-w

3. Кузнєцова Н.В., Бідюк П.І. Теорія і практика аналізу фінансових ризиків: системний підхід. Київ: Ліра-К, 2020. 400 с.

4. Rasmussen C. E., Williams C. K. I. Gaussian processes for machine learning. Cambridge, MA, MIT Press, 2006. 248 p.

5. Kuss M., Rasmussen C. E. Assessing approximate inference for binary gaussian process classification. *Journal of Machine Learning Research*. 2005. No. 6. P. 1679–1704.

6. Митник О. Ю. Швидкий метод наближення маргінальної правдоподібності для оцінки адекватності моделей нелінійних стохастичних процесів. *Моделювання та дослідження стійкості динамічних систем* : тези міжнар. конф. Київ: Вісник КНУ, 2007. С.70.

7. Paun I., Husmeier D., Torney C. J. Stochastic variational inference for scalable non-stationary Gaussian process regression. *Statistics and Computing*. 2023. Vol. 33. No. 44.

8. Vinokur I., Tolpin D. Warped Input Gaussian Processes for Time Series Forecasting. *Cyber Security Cryptography and Machine Learning*. CSCML 2021. Springer, Cham. P. 205–220.

АНТИФРОД-СИСТЕМА ДЛЯ ІДЕНТИФІКАЦІЇ ШАХРАЙСЬКИХ КАРТКОВИХ ТРАНЗАКЦІЙ

Негуляєв Н.Є.[1], Недашківська Н.І.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] nikneg000@gmail.com [0000-0000-0000-0000],
[2] n.nedashkivska@gmail.com [0000-0002-8277-3095]

Розроблено систему виявлення шахрайських транзакцій з платіжними картами з використанням методів аналізу даних, машинного навчання та глибоких нейронних мереж. Тестування системи проводилось на даних IEEE-CIS Fraud Detection, що містять реальні транзакції електронної комерції. Реалізовано та порівняно ефективність моделей LSTM, GRU, LightGBM та XGBoost. Розроблений ансамбль моделей досяг точності 91% у виявленні шахрайських операцій, що підтверджує ефективність запропонованого підходу.

Ключові слова: антифрод-система, аналіз даних, машинне навчання, нейронні мережі, шахрайські транзакції, ансамбль моделей

1. ВСТУП

Виявлення шахрайських операцій з платіжними картками є критично важливим завданням для фінансового сектору. За даними Vesta Corporation, щороку компанії та споживачі втрачають мільйони доларів через шахрайські операції [1]. Використання методів машинного навчання та глибоких нейронних мереж дозволяє автоматизувати процес виявлення підозрілих транзакцій та мінімізувати фінансові втрати. Розробка ефективної антифрод-системи потребує комплексного підходу, що включає аналіз даних, створення інформативних ознак та побудову ансамблю моделей для досягнення максимальної точності виявлення шахрайства.

2. ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

Метою роботи є розробка ефективної антифрод-системи для виявлення шахрайських карткових транзакцій з використанням методів машинного навчання та глибоких нейронних мереж. В рамках дослідження необхідно виконати аналіз та підготовку даних транзакцій з урахуванням особливостей фінансового домену, включаючи створення інформативних ознак для покращення якості класифікації. Важливою складовою є розробка та навчання моделей різних типів, таких як LSTM, GRU, LightGBM та XGBoost для виявлення шахрайських операцій. Система має забезпечувати створення ансамблю моделей для досягнення максимальної точності класифікації та проведення оцінки ефективності на реальних даних з порівнянням існуючих рішень. Окремою задачею є оптимізація моделей для забезпечення балансу між точністю виявлення шахрайства та мінімізацією помилкових спрацювань, що критично важливо для збереження позитивного користувацького досвіду при проведенні фінансових операцій.

3. ОСНОВНІ МЕТОДИ

Шахрайські транзакції залежать від великої кількості факторів, включаючи як характеристики самої транзакції (сума, час, локація), так і поведінкові шаблони користувачів, історичні дані та зовнішні фактори [2]. Це робить задачу виявлення шахрайства складною, адже через комплексність процесів та велику кількість змінних неможливо створити прості правила для безпомилкової класифікації. Аналіз транзакційних даних є основою для всіх подальших дій системи [3]. Для розв'язання задачі класифікації було використано кілька різних моделей, що дозволяє знайти оптимальний підхід для різних типів шахрайських операцій, які можуть мати різний характер та описуватися різними шаблонами.

Однією з ключових моделей розглядається LSTM (Long Short-Term Memory) – різновид рекурентних нейронних мереж, що спеціально розроблений для роботи з послідовними даними [4]. Основна ідея методу базується на використанні спеціальних вентильних механізмів, які дозволяють мережі "запам'ятовувати" важливу інформацію та "забувати" неважливу. Особливість методу в її здатності виявляти довгострокові залежності в даних, що критично важливо для аналізу послідовності транзакцій одного користувача [5].

Ще однією моделлю було обрано GRU (Gated Recurrent Unit) – спрощену версію LSTM, яка зберігає високу ефективність при меншій кількості параметрів [6]. GRU використовує лише два типи вентилів замість трьох у LSTM, що робить модель більш економною з точки зору обчислювальних ресурсів при збереженні високої якості класифікації. Ця архітектура особливо ефективна для обробки транзакцій в реальному часі, де швидкість роботи є критичним фактором [7].

LightGBM представляє собою високопродуктивну реалізацію градієнтного бустингу, що відрізняється швидкістю навчання та ефективною роботою з категоріальними змінними [8]. Модель використовує особливу техніку leaf-wise росту дерев, що дозволяє досягати кращих результатів при менших обчислювальних витратах [9]. Ця особливість робить LightGBM особливо корисним для обробки великих наборів транзакційних даних.

XGBoost (Extreme Gradient Boosting) – це потужний алгоритм градієнтного бустингу, що добре зарекомендував себе в задачах виявлення шахрайства завдяки здатності ефективно обробляти розріджені дані та стійкості до перенавчання [10]. Модель включає додаткові механізми регуляризації та оптимізації, що дозволяє досягати високої точності класифікації при збереженні узагальнюючої здатності [11].

Всі наведені вище моделі використовуються для класифікації транзакцій на шахрайські та легітимні. Водночас, важливим аспектом є створення ансамблю моделей, який дозволяє об'єднати сильні сторони кожного підходу [12]. Ансамблювання здійснюється шляхом зваженого усереднення результатів окремих моделей, де ваги визначаються на основі валідаційної вибірки. Такий підхід дозволяє підвищити стабільність та надійність системи, зменшуючи вплив помилок окремих моделей [13].

Для оцінки ефективності системи використовується метрика ROC AUC, яка дозволяє оцінити якість класифікації незалежно від вибору порогу прийняття рішення [14]. Це особливо важливо в контексті антифрод-систем, де необхідно знаходити баланс між виявленням шахрайських операцій та мінімізацією помилкових спрацювань.

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Реалізація антифрод-системи вимагає ретельної перевірки ефективності роботи всіх компонентів. Для тестування було використано набір даних IEEE-CIS Fraud Detection, що містить реальні транзакції електронної комерції. Аналіз даних показав істотні відмінності у розподілі типів платіжних карток (табл. 1). Більшість транзакцій здійснювалась через карти Visa (66%, 334 415 транзакцій) та Mastercard (31%, 157 074 транзакцій).

Таблиця 1. Розподіл типів платіжних карток

Тип картки	Відсоток	Кількість транзакцій
Visa	66%	334 415
Mastercard	31%	157 074
Інші	3%	15 202

Аналіз розподілу транзакцій за сумою платежу (табл. 2) показав, що переважна більшість операцій (84,4%) знаходиться в діапазоні від 0,02 до 205,42 USD, що є типовим для роздрібної електронної комерції.

Таблиця 2. Розподіл транзакцій за сумою платежу

Діапазон суми (USD)	Кількість транзакцій	Відсоток
0.02 – 205.42	427 843	84.4%
205.42 – 410.82	51 475	10.2%
410.82 – 616.22	14 383	2.8%
616.22+	13 141	2.6%

Для оцінки ефективності моделей було проведено навчання з п'ятифолдовою крос-валідацією. На рис. 1 представлено динаміку валідаційних втрат для LSTM та GRU моделей протягом навчання. Графіки демонструють стабільне зменшення функції втрат, причому LSTM показує більш рівномірну динаміку порівняно з GRU.

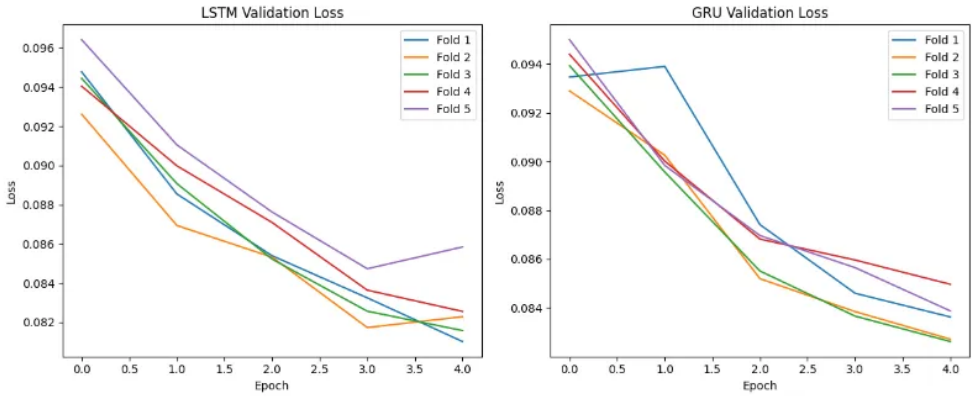


Рисунок 1. Динаміка валідаційних втрат LSTM та GRU моделей

Візуальне порівняння ROC AUC метрик для всіх моделей представлено на рис. 2.

Динаміка навчання LSTM мережі (табл. 3) показує поступове покращення як точності, так і функції втрат. Починаючи з точності 0,9703 на першій епосі, модель досягає 0,9792 на п'ятій епосі, при цьому валідаційні втрати зменшуються з 0,0948 до 0,0810.

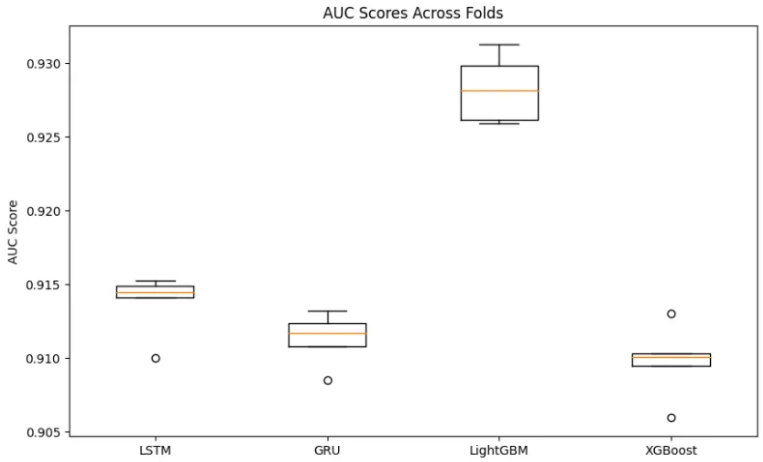


Рисунок 2. Порівняння ROC AUC метрики для різних моделей

Таблиця 3. Динаміка навчання LSTM мережі

Епоха	Точність	Функція втрат	Валідаційна точність	Валідаційні втрати
1	0.9703	0.1214	0.9750	0.0948
2	0.9752	0.0928	0.9759	0.0885
3	0.9768	0.0859	0.9766	0.0854
4	0.9781	0.0806	0.9770	0.0832
5	0.9792	0.0760	0.9775	0.0810

Порівняльний аналіз ефективності всіх моделей (табл. 4) показує, що найкращі результати демонструє LightGBM з показником ROC AUC 0,9283 (±0,0021). LSTM показує другий результат (0,9137 ±0,0019), за ним слідує GRU (0,9113 ±0,0016), а XGBoost демонструє найнижчий показник (0,9098 ±0,0023).

Модель LightGBM також демонструє стабільне покращення під час навчання (табл. 5), про що свідчить зменшення значення logloss з 0,19541 на початку до 0,10641 на 20-й ітерації.

Таблиця 4. Порівняння ефективності моделей

Модель	ROC AUC	Стандартне відхилення
LSTM	0.9137	±0.0019
GRU	0.9113	±0.0016
LightGBM	0.9283	±0.0021
XGBoost	0.9098	±0.0023

Таблиця 5. Динаміка навчання LightGBM класифікатора

Ітерація	Значення logloss
0	0.19541
5	0.15147
10	0.12809
15	0.11462
20	0.10641

Отримані результати підтверджують ефективність розробленої системи для виявлення шахрайських транзакцій, особливо при використанні ансамблю моделей, що дозволяє досягти стабільної роботи системи в різних умовах.

5. ВИСНОВКИ

Для ефективного захисту від шахрайських операцій з платіжними картками критично важливо мати надійну систему виявлення підозрілих транзакцій. Розроблена антифрод-система успішно вирішує це завдання, використовуючи комбінацію сучасних методів аналізу даних, машинного навчання та глибоких нейронних мереж. Чотири реалізовані моделі (LSTM, GRU, LightGBM та XGBoost) дозволяють отримати різні підходи до класифікації, що разом з механізмом ансамблювання забезпечує високу точність виявлення шахрайства.

Отримані результати демонструють, що розроблені моделі успішно справляються із задачею класифікації, показуючи високу точність (ROC AUC 0,91) та низький рівень помилкових спрацювань. Особливо ефективною виявилась модель LightGBM, яка поєднує високу точність з швидкістю обробки даних. Використання рекурентних нейронних мереж (LSTM та GRU) дозволяє ефективно враховувати послідовний характер транзакцій та виявляти складні шаблони шахрайської поведінки.

Створений ансамбль моделей демонструє кращі результати порівняно з окремими моделями, що підтверджує ефективність обраного підходу до комбінування різних методів класифікації. Це особливо важливо для реальних застосувань, де необхідна висока стабільність роботи системи та мінімізація помилкових спрацювань.

Загалом, розроблена антифрод-система виконує поставлені задачі та реалізує необхідний функціонал для ефективного виявлення шахрайських транзакцій. Можливими шляхами вдосконалення є впровадження механізмів автоматичного оновлення моделей на нових даних та адаптація до нових типів шахрайських атак. Додавання інших типів моделей та вдосконалення методів feature engineering також може покращити якість класифікації при наявності відповідних обчислювальних ресурсів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Vesta. The Best Solution for Prepaid Wireless Fraud Protection + Payments. https://vesta.io/.

2. Adewumi A. O., Akinyelu A. A. A survey of machine-learning and nature-inspired based credit card fraud detection techniques. International Journal of System Assurance Engineering and Management. 2017. Vol. 8, No. 2. P. 937-953.

3. Jurgovsky J., Granitzer M., Ziegler K., Calabretto S., Portier P. E., He-Guelton L., Caelen O. Sequence classification for credit-card fraud detection. Expert Systems with Applications. 2018. Vol. 100. P. 234-245.

4. Hochreiter S., Schmidhuber J. Long short-term memory. Neural computation. 1997. Vol. 9, No. 8. P. 1735-1780.

5. Graves A. Supervised sequence labelling with recurrent neural networks. Studies in Computational Intelligence. Berlin: Springer, 2012. 385 p.

6. Cho K., Van Merriënboer B., Gulcehre C., Bahdanau D., Bougares F., Schwenk H., Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078. 2014.

7. Chung J., Gulcehre C., Cho K., Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555. 2014.

8. Ke G., Meng Q., Finley T., Wang T., Chen W., Ma W., Ye Q., Liu T. Y. LightGBM: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems. 2017. P. 3146-3154.

9. Zhang J., Xie Y., Wu Q., Xia Y. Accurate and efficient fraud detection using LightGBM. Journal of Physics: Conference Series. 2019. Vol. 1325, No. 1. P. 012091.

10. 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.

11. Carmona P., Climent F., Momparler A. Fraud detection in bank account operations using deep learning. Pattern Recognition Letters. 2019. Vol. 121. P. 73-79.

12. Zhou Z. H. Ensemble methods: foundations and algorithms. Chapman and Hall/CRC, 2012. 236 p.

13. Dietterich T. G. Ensemble methods in machine learning. Multiple classifier systems. Berlin: Springer, 2000. P. 1-15.

14. Fawcett T. An introduction to ROC analysis. Pattern Recognition Letters. 2006. Vol. 27, No. 8. P. 861-874.

СИСТЕМА ПОШУКУ АНОМАЛІЙ МЕТОДАМИ ГЛИБОКИХ НЕЙРОННИХ МЕРЕЖ ТА БАГАТОКРИТЕРІАЛЬНОЇ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕННЯ

Петькун О.Ю.[1], Недашківська Н.І.

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] petkun8034@gmail.com

Актуальність теми полягає у необхідності ефективного виявлення аномалій у різноманітних наборах даних, таких як медичні зображення МРТ та транзакційні записи банківських карток. Виявлення таких аномалій має велике практичне значення для сфер медичної діагностики та фінансового моніторингу.

У дисертації досліджується застосування методів глибокого навчання, зокрема згорткових нейронних мереж та автокодувальників, для аналізу даних з метою виявлення аномалій. Додатково використовується багатокритеріальний підхід на основі методу TOPSIS для прийняття рішень щодо виявлених аномалій.

Результатом роботи є розробка комплексної системи пошуку аномалій, яка демонструє високу ефективність на реальних наборах даних у медичній та фінансовій сферах. Практична цінність роботи полягає у можливості застосування розроблених методів для покращення процесів діагностики, запобігання шахрайству та підвищення загальної надійності систем обробки даних.

Ключові слова: виявлення аномалій, глибокі нейронні мережі, багатокритеріальний аналіз, медичні зображення, транзакційні дані.

1. ВСТУП

Актуальність теми полягає у необхідності ефективного виявлення аномалій у різноманітних наборах даних, таких як медичні зображення МРТ та транзакційні записи банківських карток. Своєчасне виявлення аномальних патернів у цих даних має велике практичне значення, оскільки може сприяти покращенню діагностики захворювань, запобіганню шахрайству та підвищенню загальної надійності систем обробки інформації.

У сучасному світі, наповненому великими масивами різноманітних даних, ручний аналіз для виявлення аномалій стає дедалі менш ефективним. Тому дослідження методів автоматизованого пошуку та ідентифікації аномалій із застосуванням сучасних підходів, таких як глибокі нейронні мережі та багатокритеріальний аналіз, є актуальною науково-практичною задачею.

Метою даної роботи є підвищення ефективності виявлення аномалій у медичних зображеннях МРТ та транзакційних даних банківських карток шляхом розробки комплексної системи, що поєднує методи глибокого навчання та багатокритеріальної підтримки прийняття рішень. Це дозволить створити надійні інструменти для аналізу даних, здатні

автоматично виявляти критичні та підозрілі елементи, що матимуть вагоме практичне значення для медичної діагностики та фінансового моніторингу.

Отже, виконання цієї магістерської дисертації спрямоване на значне підвищення можливостей сучасних методів аналізу даних у сфері виявлення аномалій, що сприятиме підвищенню ефективності та надійності відповідних систем обробки інформації.

2. ОСНОВНІ МЕТОДИ

Для вирішення задачі пошуку аномалій та підтримки прийняття рішень у рамках магістерської роботи було обрано дві різні архітектури нейронних мереж: згорткову нейронну мережу (Convolutional Neural Network, CNN) та автокодувальник (Autoencoder). Обидві моделі призначені для різних етапів обробки даних, що дозволяє ефективно поєднувати їх у системі пошуку аномалій з подальшим використанням результатів у багатокритеріальній системі підтримки прийняття рішень. Згорткові нейронні мережі (CNN) є природним вибором для аналізу візуальних даних завдяки їхній здатності виділяти локальні ознаки зображень на різних рівнях абстракції. Для навчання CNN було використано перший датасет, який містить анотації до нормальних та аномальних зразків. Ця модель здатна класифікувати зображення на дві категорії (норма/аномалія), використовуючи вбудовані шари згорток і пулінгу для аналізу візуальних ознак. Автокодувальники були обрані як друга модель, оскільки вони є потужним інструментом для виявлення аномалій у великих масивах даних. На відміну від CNN, автокодувальник навчається відновлювати нормальні зразки через стискання і декодування інформації. Таким чином, модель здатна ефективно знаходити аномалії як зразки, які суттєво відрізняються від нормальних даних, оскільки їх відновлення матиме великі похибки. Для навчання автокодувальника використовувався інший датасет, що складається з нормальних зразків без явних аномалій, для більш точного навчання моделі на нормальній поведінці системи [1].

CNN для зображень МРТ мозку: У цьому випадку, згорткова нейронна мережа добре підходить для аналізу просторових ознак, характерних для зображень. CNN дозволяє виділити різні рівні абстракції в медичних зображеннях, такі як патерни аномалій (пухлини, структурні зміни тощо). Для виявлення аномалій модель може бути навчена на нормально структурованих зображеннях мозку і перевірена на зразках з відхиленнями.

CNN для транзакційних даних: Хоча згорткові нейронні мережі не є звичними для роботи з табличними даними, їх можна використовувати для виявлення аномалій у певних типах послідовних або багатовимірних транзакційних даних, застосовуючи згортки для аналізу поведінкових патернів або тимчасових ознак. Важливо попередньо підготувати дані для застосування CNN, перетворивши їх у формат, придатний для обробки згортками (наприклад, на рівні послідовностей) [2].

Автокодувальник для зображень МРТ мозку: Автокодувальник може бути застосований для виявлення аномалій, коли модель намагається відновити нормальні зображення мозку. Якщо є велика розбіжність між оригінальним та відновленим зображенням, то можна вважати, що зразок аномальний. Похибка відновлення є ключовою метрикою.

Автокодувальник для транзакційних даних: У випадку з транзакціями автокодувальник можна використовувати для виявлення незвичних патернів у фінансових операціях. Як і у випадку з зображеннями, модель навчається відновлювати "нормальні" транзакції, а потім аномалії визначаються через велику похибку відновлення для підозрілих зразків [3, 4].

В якості підтримки прийняття рішення можна використати багатокритеріальний метод – TOPSIS. TOPSIS – це техніка багатокритеріальної підтримки прийняття рішень (MCDM),

яка широко використовується для вирішення завдань прийняття рішень. Як компенсаційний метод агрегації, цей підхід використовується для вибору найкращого варіанту на основі жорстких порогових значень. У TOPSIS m атрибутів для n кандидатів (мереж) можуть бути представлені у вигляді матриці A(n×m).

Застосуємо метод TOPSIS (Technique for Order Preference by Similarity to Ideal Solution), який дозволяє обчислити рейтинг кожної моделі на основі віддаленості від ідеального рішення (найкращих і найгірших можливих значень критеріїв). Для цього:

1. **Ідеальне рішення (P+_++)**: Найкращі можливі значення критеріїв для кожної моделі. Це буде точка, де кожен критерій має максимальне або мінімальне (для витратних критеріїв) значення.
2. **Антиідеальне рішення (P−_−−)**: Найгірші можливі значення критеріїв.
3. **Обчислення відстані до ідеального та антиідеального рішень**:

$$d_+ = \sqrt{\sum_{i=1}^{n} w_i (N_i + P_i^+)^2},\ d_- = \sqrt{\sum_{i=1}^{n} w_i (N_i + P_i^-)^2},$$

де w_i – вага критерію, N_i – нормалізоване значення критерію для моделі, P_i^+ *та* P_i^- – значення ідеальних та антиідеальних рішень.

4. **Обчислення близькості до ідеального рішення (C_i)**: $C_i = \frac{d}{d_- + d_-}$ Значення C_i показує відносну відстань до ідеального рішення і слугує індикатором того, наскільки добре працює модель [5].

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

В даній роботі розглядатиметься 2 датасети – МРТ зображеннями головного мозку та даними по транзакціях карток. Відповідно перший відноситься до предметної області медицини, а другий – аналіз поведінки користувачів. Crystal Clean: Brain Tumors MRI Dataset. даних, доступний у форматі архіву RAR, складається з чотирьох класів, у тому числі трьох класів пухлин (гіпофіза, гліоми та менінгіоми) та одного класу, що представляє звичайне МРТ-сканування головного мозку.

Для поставленої задачі цей дата сет був модифікований таким чином, аби він містив лише 2 класи – нормальні та ненормальні (рис. 1). Це дало змогу використати цей датасет для задачі бінарної класифікації при використанні загорткової мережі, а також для задачі відновлення оригінальних даних при використанні авто кодувальника. Загалом цей набір даних містить 21672 зразків зображень. Цей розмір було зменшено до 7986 зразків [6].

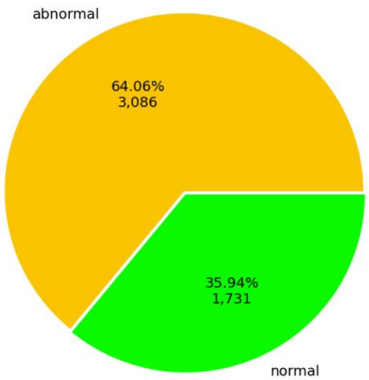


Рисунок 1. Розподіл зображень за класами у Crystal Clean: Brain Tumors MRI Dataset

Наступний датасет із транзакційними даними – Credit Card Fraud Detection. Він являє собою транзакції, здійснені кредитними картками у вересні 2013 року європейськими власниками карток.

Цей набір даних представляє транзакції, які відбулися за два дні, де ми маємо 492 шахрайства з 284 807 транзакцій. Набір даних дуже незбалансований, позитивний клас (шахрайство) становить 0,172% усіх транзакцій (рис. 2).

Він містить лише числові вхідні змінні, які є результатом перетворення PCA. Функції V1, V2, … V28 є основними компонентами, отриманими за допомогою PCA, єдиними характеристиками, які не були перетворені за допомогою PCA, є «Час» і «Сума». Функція «Час» містить секунди, що минули між кожною транзакцією та першою транзакцією в наборі даних. Функція «Сума» – це сума транзакції, цю функцію можна використовувати для навчання, залежного від вартості, залежного від прикладу. Функція «Клас» – це змінна відповіді, яка приймає значення 1 у разі шахрайства та 0 в іншому випадку [7].

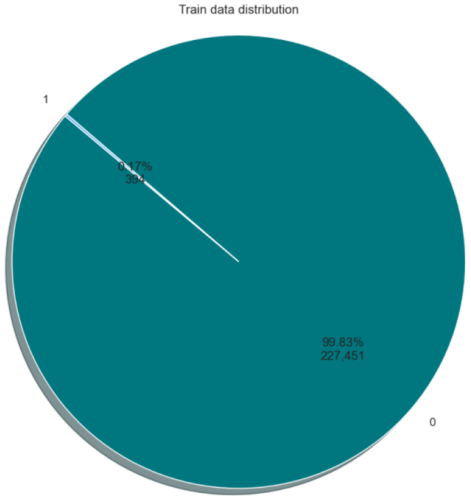


Рисунок 2. Розподіл даних за класами у Credit Card Fraud Detection

Як вже зазначалось раніше, для підтримки прийняття рішення використовувався метод TOPSIS. У якості атрибутів для прийняття рішень використовувались значення оцінок метрик якості натренованих моделей для різних наборів даних, а саме: F1, Recall, Precision, Accuracy та Auc Roc. Отже результати цих метрик продемонстровані у таблиці 1.

Таблиця 1. Оцінки прогнозів побудованих моделей для різних наборів даних

	Image Dataset		Text transactions Dataset	
	Autoencoder	CNN	Autoencoder	CNN
Accuracy	0.49	0.901	0.566	0.999
Precision	0.203	0.889	0.003	0.858
Recall	0.036	0.897	0.969	0.744
F1	0.061	0.893	0.007	0.797
Auc Roc	0.381	0.937	0.962	0.971

З огляду на ці метрики наведено результати багатокритеріальної підтримки прийняття рішень TOPSIS (рис. 3, 4).

```
TOPSIS Results:
             TOPSIS Score  Rank
Model
Autoencoder           0.0   2.0
CNN                   1.0   1.0
```

Рисунок 3. Результат TOPSIS для дата сету із зображеннями

```
TOPSIS Results:
             TOPSIS Score  Rank
Model
Autoencoder      0.110298   2.0
CNN              0.889702   1.0
```

Рисунок 4. Результат TOPSIS для дата сету із текстовими транзакційними даними

Результати методу показують очікувані результати, зважаючи на описані у таблиці 1 вхідні атрибути. В даному випадку і для текстових даних і для зображень найкращою моделлю виявилась згорткова мережа. І якщо для зображень цей результат виглядає очікуваним, то для текстових даних – ні. Можливою причиною гірших результатів відпрацювання авто кодувальника на текстових даних може бути незбалансована кількість нормальних до аномальних даних.

4. ВИСНОВОК

Пошук аномалій є важливою задачею в багатьох сферах, від кібербезпеки та фінансів до охорони здоров'я та промисловості. Аномальні дані можуть свідчити про потенційні загрози, несправності або інші критичні ситуації, що потребують швидкої реакції. Одним із найбільш перспективних підходів до вирішення цієї проблеми є використання методів глибоких нейронних мереж. Глибокі нейронні мережі можуть самостійно навчатися на великих обсягах даних, виявляючи складні шаблони та залежності, що дозволяє їм ефективно знаходити аномалії.

Окрім використання нейронних мереж, важливу роль у пошуку аномалій може відігравати багатокритеріальна підтримка прийняття рішень. Це підхід, при якому при аналізі результатів враховуються кілька критеріїв одночасно.

У ході дослідження було розроблено систему, котра може опрацьовувати як текстові дані, так і зображення, застосовувати різні моделі нейронних мереж для навчання на оброблених даних та виявленні аномалій, а також порівнювати результати різних моделей та приймати рішення щодо обрання найкращої з них.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Коцовський В. М., 2013. Нейронні системи. Доступно: https://dspace.uzhnu.edu.ua/jspui/bitstream/lib/16450/1/%D0%9D%D0%B5%D0%B9%D1%80%D0%BE%D0%BD%D0%BD%D1%96%20%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B8.%20%D0%9B%D0%B5%D0%BA%D1%86%D1%96%D1%97.pdf
2. Jonas Teuwena,b, Nikita Moriakova, 2019. Convolutional neural networks. Доступно: https://www2.cs.uh.edu/~ceick/ai/CNN_Reading_Material.pdf
3. Недашківська Н.І., 2021. ІСППР Лекція №17 Автокодувальники ч. 1.
4. Недашківська Н.І.. 2021. ІСППР Лекція №17 Автокодувальники ч. 2.
5. Jianguo Deng, Yuan Wang, 2023. Evaluation Model of Study Style Based on Neural Network and TOPSIS. Доступно: https://francis-press.com/papers/14194
6. Crystal Clean: Brain Tumors MRI Dataset, 2023. Доступно: Crystal Clean: Brain Tumors MRI Dataset
7. Credit Card Fraud Detection, 2017. Доступно: Credit Card Fraud Detection

РОЗРОБКА IOS-ДОДАТКУ ДЛЯ ВІРТУАЛЬНОЇ ТОРГІВЛІ ТА ПРОГНОЗУВАННЯ ЦІН

Радченко Д.О.[1], Кузнєцова Н.В.

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», ІПСА, Київ, Україна

[1] dimonkofan@gmail.com

Сучасна динаміка криптовалютних ринків створює нові виклики і широкий спектр задач і функціоналу для інвесторів і розробників програмного забезпечення. У даній роботі розглянуто підхід до створення симулятора інвестицій в криптовалюту, який дозволяє користувачам здійснювати віртуальні торгові операції та прогнозувати ціни без реальних фінансових ризиків. Було розроблено iOS-додаток для прогнозування цін на криптовалюту з використанням сучасних моделей оцінювання і прогнозування фінансових процесів, таких як: ARIMA, GARCH та LSTM. Основна увага приділена прогнозуванню цін криптовалют на основі історичних даних і підвищенню точності моделей.

Ключові слова: криптовалюта, iOS-додаток, симулятор інвестицій, прогнозування, машинне навчання, FastAPI, UIKit.

1. ВСТУП

З розвитком цифрових технологій та появою нових фінансових інструментів криптовалюта стала невід'ємною частиною сучасної економіки. Її висока волатильність та складність прогнозування цін створюють серйозні виклики для інвесторів, особливо тих, хто тільки починає працювати на ринку. У сучасному світі, де ухвалення рішень часто базується на швидкому аналізі великих обсягів даних, виникає потреба в ефективних інструментах навчання та симуляції.

Метою даного дослідження є розробка iOS-додатку для симуляції інвестицій у криптовалюту, що дозволяє користувачам здійснювати віртуальні торгові операції без ризику втрати реальних коштів. У додатку реалізовано клієнт-серверну архітектуру, де серверна частина побудована за допомогою FastAPI [1], забезпечуючи швидку та ефективну обробку запитів. REST API використовується для взаємодії між клієнтом і сервером, що дозволяє отримувати актуальні ринкові дані та виконувати симуляції операцій у реальному часі.

Основними аспектами роботи є впровадження сучасних моделей прогнозування, таких як ARIMA, GARCH і LSTM [3], для підвищення точності аналізу ринку та надання користувачам інструментів для кращого розуміння динаміки криптовалютного ринку. Результатом дослідження стане функціональний додаток, який допоможе інвесторам отримати базові навички торгівлі та аналізу ринкових даних, а також покращить розуміння особливостей роботи криптовалютного ринку.

2. РОЗРОБКА ІНВЕСТИЦІЙНОГО СИМУЛЯТОРА КРИПТОВАЛЮТ

Розробка інвестиційного симулятора криптовалют є комплексним процесом, що включає створення архітектури додатку, реалізацію серверної частини для обробки даних, інтеграцію моделей прогнозування цін та розробку інтерфейсу користувача. Основною метою є забезпечення функціоналу, який дозволяє користувачам здійснювати віртуальні торгові

операції, отримувати точні прогнози на основі сучасних математичних моделей та навчатися ефективному аналізу ринкових даних. Цей розділ детально описує всі етапи створення симулятора, від вибору технологій до його тестування та оптимізації.

2.1 Огляд архітектури додатку

Архітектура інвестиційного симулятора криптовалют побудована за принципом клієнт-серверної моделі. Клієнтська частина реалізована на мові Swift з використанням фреймворку UIKit [2], що дозволяє створити зручний і функціональний інтерфейс для користувача. Така архітектура забезпечує гнучкість системи, її масштабованість і високу продуктивність.

2.2 Розробка серверної частини

Розробка серверної частини симулятора була зосереджена на забезпеченні стабільної взаємодії між клієнтом і сервером, а також обробці великих обсягів даних у режимі реального часу. Серверна частина реалізована на базі FastAPI, що забезпечує швидкодію та зручну інтеграцію з клієнтською частиною через REST API. Дані про криптовалюти отримуються через API Binance, що дозволяє працювати з реальними ринковими даними. Для зберігання даних використано ORM-модуль, який дозволяє ефективно працювати з базою даних, зокрема зберігати історичні дані про ціни криптовалют і транзакції. Окрім цього, сервер відповідає за інтеграцію математичних моделей прогнозування, таких як ARIMA, GARCH і LSTM, які забезпечують точність аналітики. Завдяки оптимізації запитів серверна частина гарантує швидку передачу даних та коректне виконання симуляційних операцій.

2.3 Інтеграція моделей прогнозування

Інтеграція моделей прогнозування в серверну частину симулятора дозволила забезпечити користувачів точними інструментами для аналізу динаміки криптовалютного ринку. Було використано три основні моделі: ARIMA для прогнозування трендів на основі часових рядів, GARCH для оцінки волатильності та ризиків, а також LSTM, яка спеціалізується на довготривалих залежностях у даних. Ці моделі були реалізовані у серверній частині за допомогою Python і інтегровані через REST API для обробки запитів у режимі реального часу. Проведено попереднє тестування моделей на історичних даних криптовалют, що дозволило досягти високої точності прогнозів. Взаємодія клієнта із сервером забезпечує оперативне надання аналітичної інформації для прийняття рішень користувачами.

2.4 Розробка інтерфейсу користувача

Розробка інтерфейсу користувача для симулятора криптовалют була виконана з використанням мови програмування Swift та фреймворку UIKit. Основна увага приділяється створенню інтуїтивно зрозумілого і зручного дизайну, який забезпечує легкий доступ до функціоналу додатку. Інтерфейс включає екрани авторизації, огляд ринку, торгівлі та прогнозування. Для відображення ринкових даних і графіків цін було реалізовано інтерактивні таблиці й графічні елементи. Використання анімованих переходів і динамічних оновлень забезпечило плавну взаємодію користувача з додатком. Окрім цього, було інтегровано можливість налаштування облікового запису та персоналізації інтерфейсу для зручності кожного користувача.

2.5 Тестування та оптимізація додатку

На етапі тестування додатку було проведено перевірку стабільності роботи всіх компонентів на різних iOS-пристроях, включаючи як нові моделі, так і пристрої попередніх поколінь. Особливу увагу приділено тестуванню взаємодії клієнтської частини із серверною через REST API, щоб забезпечити швидку та безперебійну передачу даних. Проведено

перевірку точності моделей прогнозування (ARIMA, GARCH, LSTM) на основі історичних даних криптовалют, а також оцінку продуктивності додатку, виконано оптимізацію алгоритмів роботи серверної частини, зокрема алгоритмів обробки запитів і передачі даних. Завдяки цьому вдалося значно покращити швидкодію додатку та забезпечити позитивний користувацький досвід.

3. РЕЗУЛЬТАТИ РОЗРОБКИ

Розроблений симулятор криптовалютних інвестицій є результатом поєднання сучасних технологій програмування, математичного моделювання та зручного дизайну інтерфейсу. Додаток забезпечує користувачам можливість симуляції віртуальних торгових операцій із криптовалютами, аналізу ринкових даних та прогнозування цін на основі історичних показників.

Далі опишемо детально функціонал розробленого додатку. Екрани авторизації (рисунок 1) та налаштування аккаунту забезпечують безпечний доступ до додатку і персоналізацію.

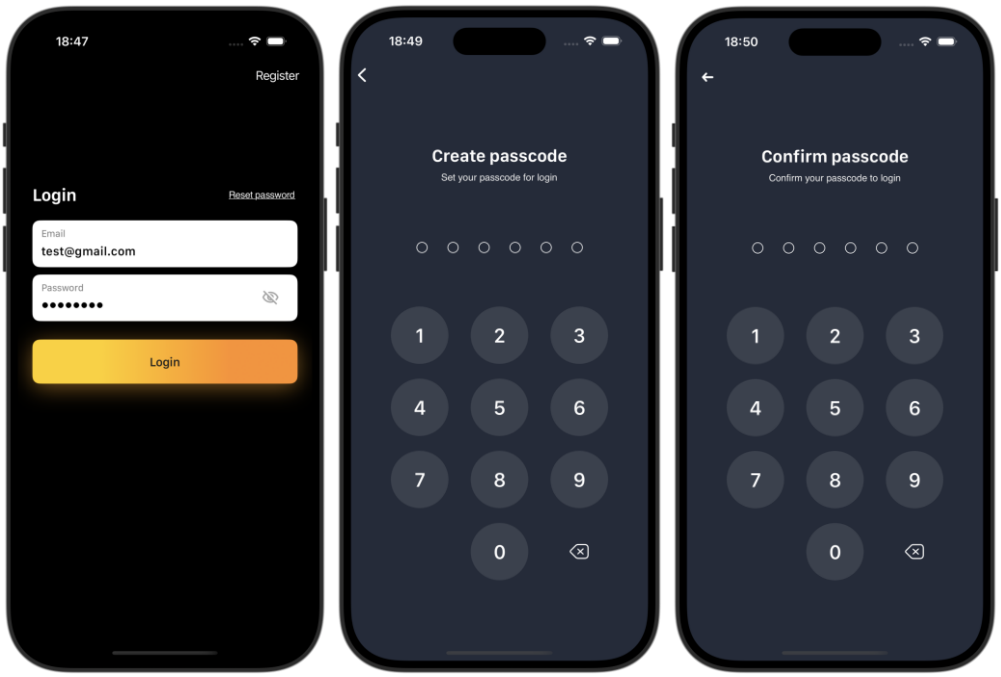


Рисунок 1. Екрани авторизації

Екран Trading (рисунок 2) дозволяє користувачам переглядати баланс і керувати криптовалютними активами. Наявна в додатку функція приховування нульових балансів забезпечує зручність для користувача.

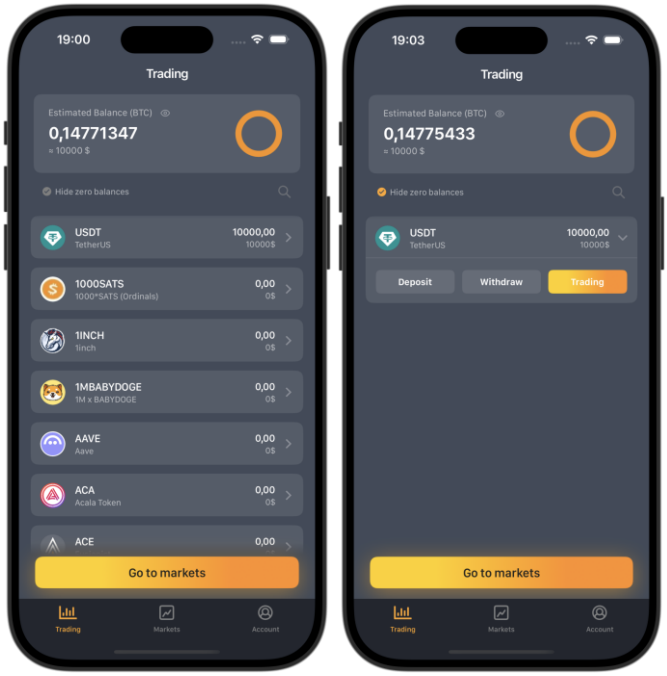


Рисунок 2. Екран Traiding без та зі прихованими нульовими балансами

Екрани маркету на рисунку 3 дозволяють відслідковувати зміни ринку криптовалют та переглядати інформацію щодо обраної валюти, включаючи графіки цін, прогнози та історію торгів.

Рисунок 3. Екрани маркету та обраної криптовалюти

Екран торгового процесу (рисунок 4) забезпечує можливість здійснення симуляційних операцій купівлі та продажу, відображаючи детальні графіки цін та історію змін (рисунок 5).

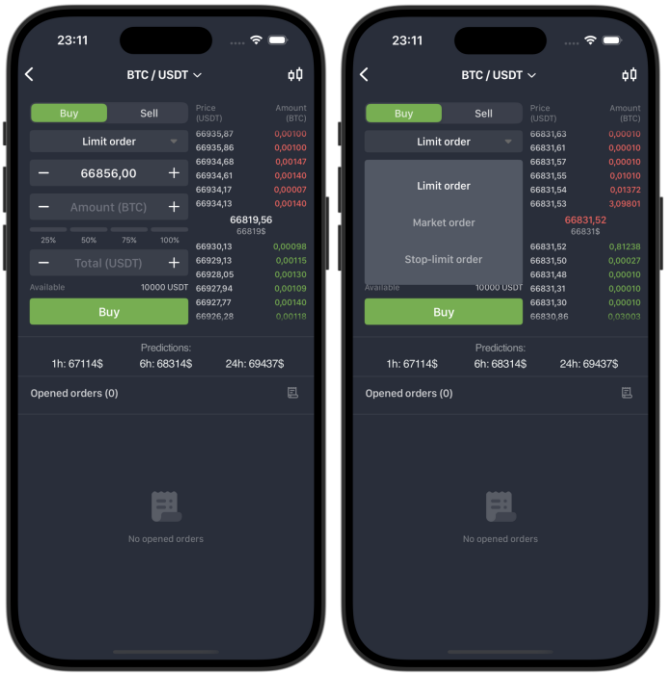


Рисунок 4. Екран торгового процесу

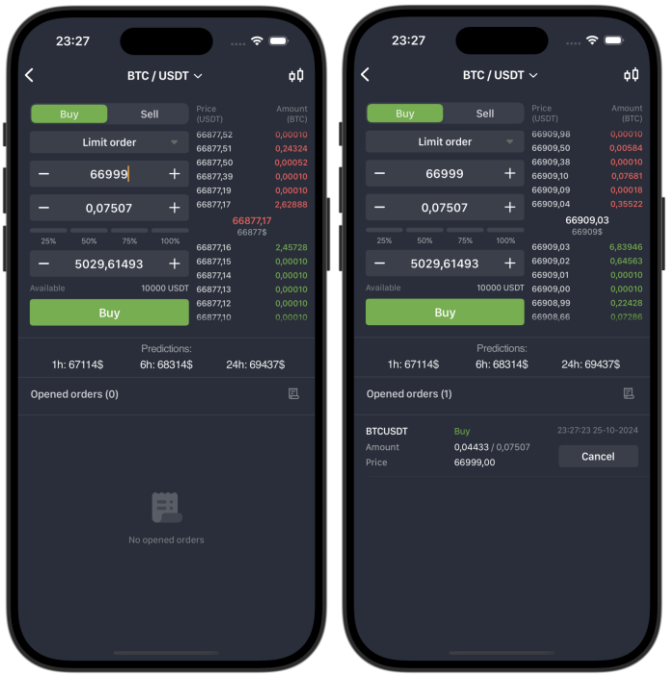


Рисунок 5. Екран торгового процесу – здійснення купівлі

Для забезпечення точності аналізу були інтегровані моделі прогнозування ARIMA та LSTM [3]. Тестування показало високу ефективність моделей у прогнозуванні цін на акції на різні періоди.

4. ВИСНОВКИ

Розробка симулятора криптовалютних інвестицій дозволила реалізувати функціональний iOS-додаток, який відповідає сучасним вимогам до інструментів для навчання і аналізу фінансових ринків. Основні завдання, включаючи створення клієнт-серверної архітектури, впровадження REST API, інтеграцію моделей прогнозування та розробку зручного інтерфейсу користувача, були успішно виконані.

Додаток надає користувачам можливість здійснювати віртуальні операції з криптовалютами, прогнозувати їхню ціну та аналізувати динаміку ринку. Використання таких моделей, як ARIMA, GARCH та LSTM, дозволило забезпечити точність прогнозів на основі історичних даних. Ретельне тестування на різних пристроях підтвердило стабільність роботи системи та її відповідність заявленим вимогам.

Таким чином, створений симулятор є ефективним інструментом для навчання початківців та вдосконалення навичок досвідчених користувачів у сфері інвестицій у криптовалюту. Розробка також підкреслила перспективність застосування сучасних технологій програмування та математичного моделювання у створенні нових фінансових продуктів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. De Luca, G. (2023). *FastAPI Cookbook*. Packt Publishing. [Інтернет посилання]: https://www.oreilly.com/library/view/fastapi-cookbook/9781805127857/. (last accessed 14.11.2024).

2. Ng, S. (2023). *Beginning iOS 17 Programming with Swift and UIKit*. AppCoda. [Інтернет посилання]: https://www.appcoda.com/swift/. (last accessed 14.11.2024).

3. I. Sibel Kervanci, M. Fatih Akay, Eren Özceylan. Bitcoin price prediction using LSTM, GRU and hybrid LSTM-GRU with bayesian optimization, random search, and grid search for the next days. Journal of Industrial and Management Optimization, 2024, 20(2): 570-588. doi: 10.3934/jimo.2023091. (last accessed 14.11.2024).

МОДЕЛІ І ПРОГНОЗИ РИНКОВИХ РИЗИКІВ

Ревенко І.Є.[1] , Левенчук Л.Б.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] coolukraine240@gmail.com,
[2] levenchuk.liudmyla@lll.kpi.ua [0000-0002-8600-0890]

Актуальність дослідження обумовлена зростаючою волатильністю фінансових ринків та ускладненням економічних взаємозв'язків, що робить прогнозування ринкових ризиків важливим завданням для фінансового сектору. Метою дослідження є розробка та дослідження системи прогнозування ринкових ризиків з використанням сучасних методів машинного навчання для підвищення ефективності управління фінансовими ризиками. У роботі запропоновано новий підхід до обробки фінансових даних, оптимізовано архітектуру LSTM-мереж для аналізу часових рядів. Створено програмну систему для прогнозування, яку можна застосовувати у фінансових установах. Результати підвищують точність прогнозів і сприяють прийняттю обґрунтованих рішень в умовах невизначеності.

Ключові слова: ринкові ризики, прогнозування, машинне навчання, фінансові часові ряди, LSTM-мережі, обробка даних, волатильність ринків, управління ризиками, фінансовий сектор, оптимізація моделей, Python, аналіз даних.

1. ВСТУП

Актуальність теми обумовлена зростанням волатильності фінансових ринків, ускладненням економічних взаємозв’язків та необхідністю забезпечення ефективного управління фінансовими ризиками. Це робить прогнозування ринкових ризиків одним із ключових завдань для учасників фінансового сектору. Традиційні методи оцінки ризиків часто виявляються недостатньо ефективними для роботи в умовах сучасних динамічних ринків, тоді як застосування методів машинного навчання відкриває нові можливості для точного прогнозування ризиків і прийняття обґрунтованих інвестиційних рішень.

Об’єктом дослідження виступає процес прогнозування ринкових ризиків у фінансовому секторі, а предметом – моделі та методи машинного навчання для прогнозування ринкових ризиків на основі аналізу історичних даних фінансових ринків.

Метою дослідження є розробка системи прогнозування ринкових ризиків з використанням сучасних методів машинного навчання для підвищення точності прогнозів і ефективності управління ризиками у фінансовій сфері.

Наукова новизна полягає у створенні нового підходу до попередньої обробки фінансових даних, який враховує специфіку ринкових показників і їх взаємозв’язки, а також у запропонованій архітектурі LSTM-мереж, оптимізованій для аналізу фінансових часових рядів.

Практичне значення результатів полягає у створенні програмної системи для прогнозування ринкових ризиків, яка може бути впроваджена у діяльність фінансових установ, розробці методик оцінки та порівняння ефективності різних моделей машинного навчання, а

також у наданні практичних рекомендацій щодо вибору та налаштування таких моделей для вирішення задач управління ризиками в умовах невизначеності.

2. ПРОГНОЗУВАННЯ РИНКОВИХ РИЗИКІВ

Мета дослідження – розробка моделей прогнозування ринкових ризиків із використанням сучасних методів машинного навчання для аналізу фінансових часових рядів. Для цього були зібрані історичні дані основних фондових індексів США (S&P 500, Dow Jones, NASDAQ) і біржових фондів (SPY, QQQ) за період 2010–2024 років через API сервісу Yahoo Finance. Окрім цінових показників, досліджувались додаткові макроекономічні індикатори, такі як індекс волатильності VIX, дохідність 10-річних державних облігацій, спред між дохідностями 10-річних і 2-річних облігацій, рівень безробіття та індекс споживчих цін (CPI). Ці індикатори відображають ключові ринкові тенденції та використовуються для побудови моделей прогнозування.

Попередня обробка даних включала кілька етапів:

1. Стандартизація структури даних. Зібрані дані з різних джерел мали різні формати, що вимагало уніфікації назв колонок і форматів даних. Наприклад, колонки "Close", "Adj Close", "Volume" були перейменовані відповідно до єдиних правил для забезпечення узгодженості між наборами даних.

2. Заповнення відсутніх значень. Пропущені дані заповнювались методами лінійної інтерполяції та ковзного середнього. Це забезпечило безперервність часових рядів, що є важливим для моделей машинного навчання.

3. Обробка аномалій. Використовувались методи видалення викидів або вінзоризації, які дозволяють усунути вплив екстремальних значень на моделі.

4. Нормалізація та стандартизація. Дані були нормалізовані за допомогою RobustScaler для приведення значень до стандартної шкали з нульовим середнім та одиничним стандартним відхиленням, що забезпечує стабільність алгоритмів оптимізації.

Аналіз кореляцій між індикаторами проводився для виявлення залежностей між різними показниками. Наприклад, висока кореляція між індексом волатильності VIX і ціновими змінами фондових індексів свідчить про тісний взаємозв’язок між ними в періоди підвищеної ринкової нестабільності. Кореляційні матриці дали змогу оцінити взаємозв'язки між такими показниками, як обсяг торгів, ковзні середні, технічні індикатори (MACD, RSI) та макроекономічні фактори (доходність облігацій, рівень інфляції). Це дозволило визначити найзначущіші предиктори для моделювання ринкових ризиків.

Для прогнозування було використано кілька методів машинного навчання, кожен із яких володіє унікальними перевагами:

1. XGBoost. Алгоритм градієнтного бустингу на основі дерев рішень, який забезпечує високу точність і швидкість роботи завдяки регуляризації та автоматичному визначенню важливості ознак.

2. LightGBM. Альтернатива XGBoost, яка використовує гистограмне навчання, що дозволяє суттєво зменшити час побудови дерев і обробляти великі обсяги даних. Ця модель особливо ефективна для великих фінансових датасетів.

3. LSTM (Long Short-Term Memory). Нейронна мережа для роботи з часовими рядами, яка завдяки спеціальним елементам – воротам забуття, вводу та виходу – зберігає інформацію про довготривалі залежності. Архітектура мережі включала кілька шарів, таких як LSTM-шар, Dropout-шар для запобігання перенавчанню, і Dense-шар для виводу результатів.

Було створено єдиний датасет із 36 ключовими ознаками, отриманими з фондових ринків і макроекономічних показників. Під час інтеграції даних проведено аналіз мультиколінеарності для усунення висококорельованих змінних та відбір найбільш значущих

ознак. Для скорочення розмірності використовувався метод головних компонент (PCA), який зменшив кількість ознак до 16 компонент, що пояснюють 95% варіації даних.

Ці підходи забезпечили якісну підготовку даних і створили основу для побудови точних предиктивних моделей.

3. АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕНЬ

У дослідженні проведено порівняння моделей XGBoost, LightGBM та LSTM для прогнозування ринкових ризиків. Оцінка моделей проводилася за показниками RMSE, R^2 та точністю напрямку (Direction Accuracy). LSTM показала перевагу за точністю прогнозування напрямку ринку на коротких і середніх інтервалах часу (5 і 10 днів), перевершивши інші моделі. На рис. 1 представлені результати тренування моделей.

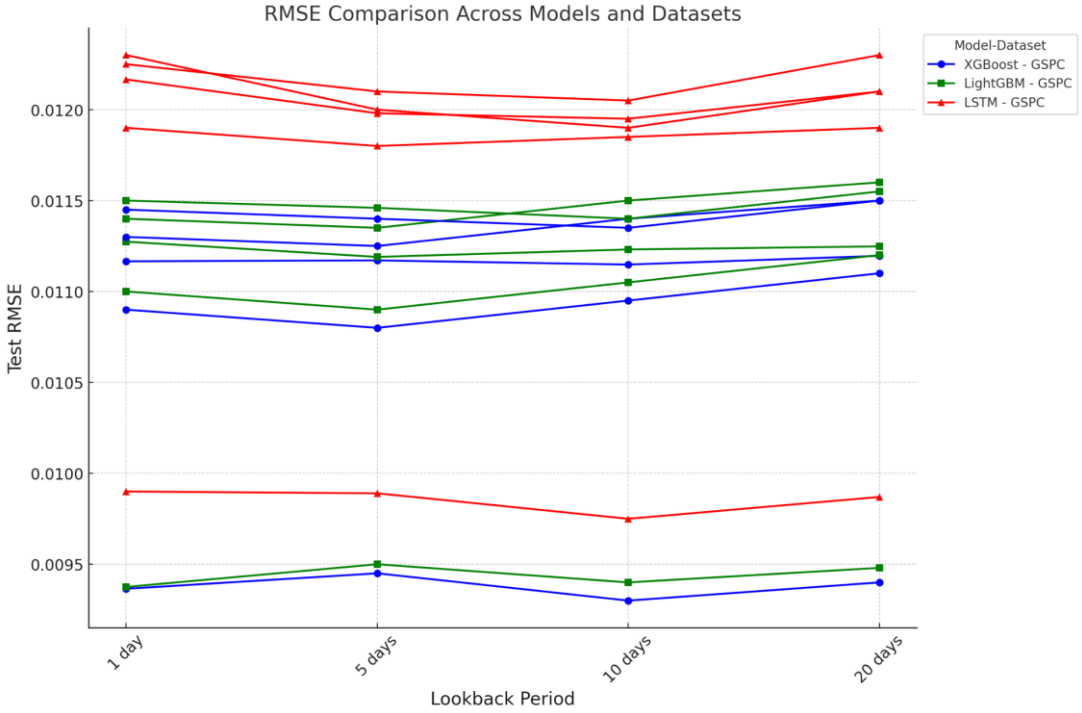


Рисунок 1. Результати тренування моделей

На цьому етапі опишу процес створення єдиного датасету для прогнозування ринкових ризиків із використанням даних фондових індексів (GSPC, DJI, IXIC, SPY, QQQ) та макроекономічних показників. Процес включав кілька етапів:

1. Дані з різних ринкових індексів було поєднано, кожен із них доповнено унікальними ознаками. Також було додано інформацію про прогнозовану дохідність на наступний день для створення комплексного датасету.

2. Аналіз кореляційної матриці до обробки на рис. 2 дозволив ідентифікувати змінні з кореляцією понад 0,7. Ці змінні були видалені, щоб уникнути мультиколінеарності та зосередитися на найбільш незалежних ознаках.

3. Застосовано методи forward fill і backward fill, що дозволило забезпечити безперервність часових рядів.

4. Ознаки з варіативністю менше 1% були видалені для покращення якості аналізу.

5. Використовувався RobustScaler, який забезпечив стійкість до аномальних значень, що є важливим для фінансових даних.

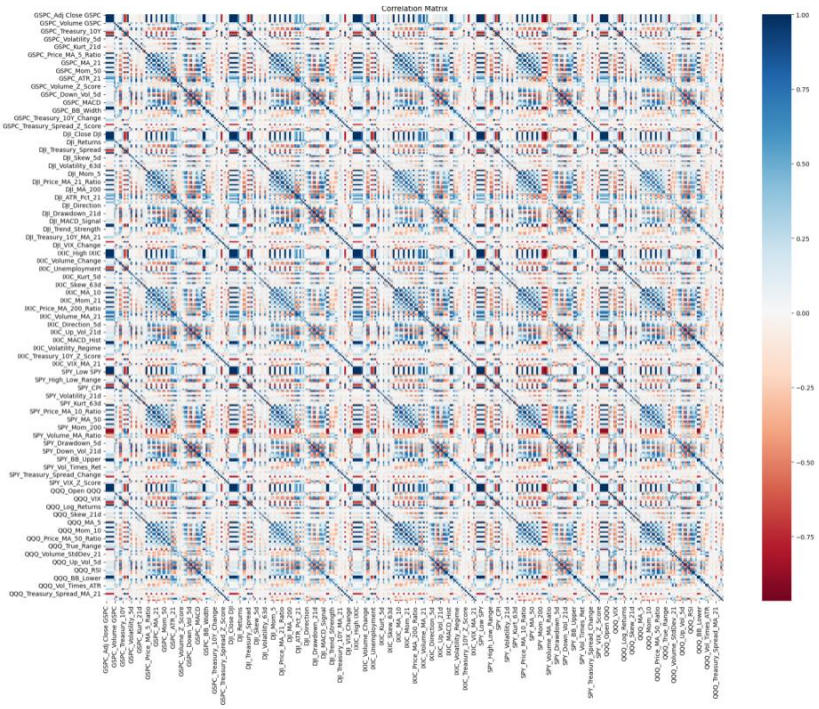


Рисунок 2. Кореляційна матриця до обробки

6. Для скорочення розмірності даних було проведено PCA, що дозволило залишити 16 основних компонент, які пояснюють 95% варіації на рис. 3 і рис. 4. Графік кумулятивної поясненої дисперсії ілюструє, як ці компоненти зберігають основну інформацію.

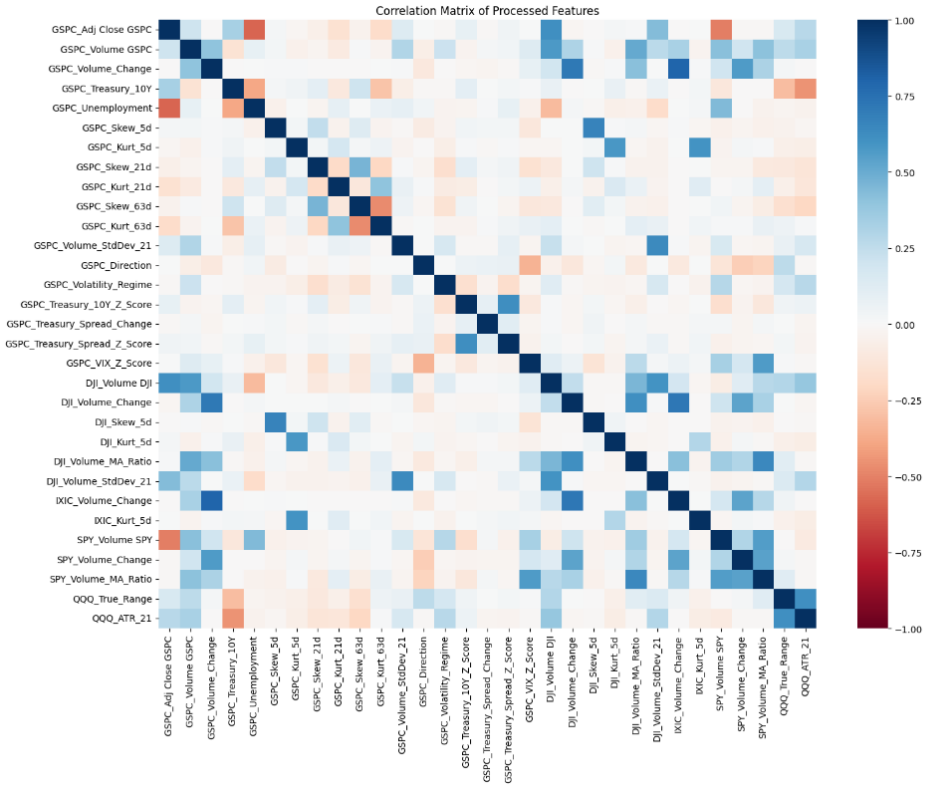


Рисунок 3. Кореляційна матриця після обробки

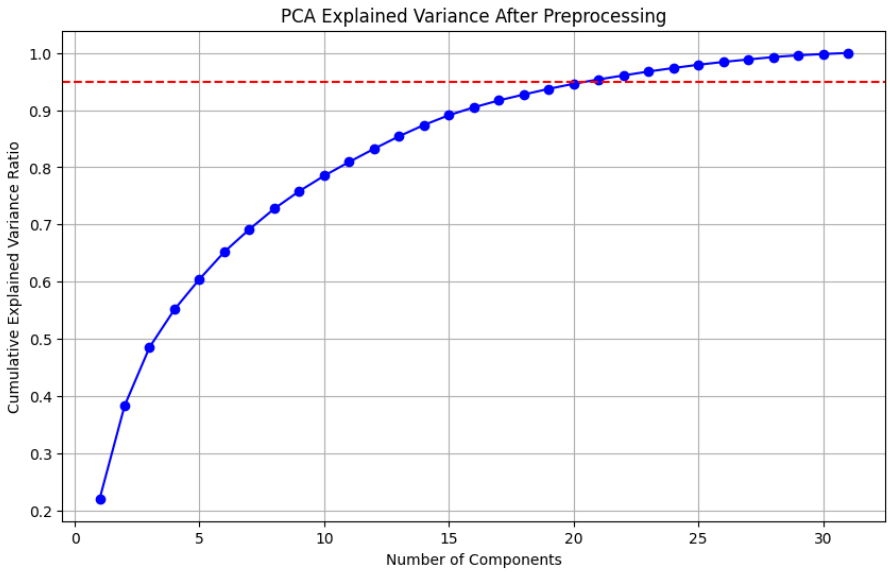


Рисунок 4. Візуалізація PCA

У результаті були виділені найзначущі ознаки, такі як зміна спреду казначейських облігацій, зміна обсягів торгів і напрямок ринку. Ці ознаки мають ключовий вплив на прогнозування ринкових ризиків і формують основу для ефективних моделей машинного навчання.

Моделі були протестовані на двох датасетах (36 і 94 ознак). Lasso та ElasticNet продемонстрували найкращі результати за точністю напрямку (0,5421) і низьким RMSE (0,010171). Модель LSTM також досягла високої точності прогнозів завдяки здатності враховувати складні взаємозв'язки в часових рядах.

Таблиця 1. Найкращі результати кожної моделі.

Модель	Dataset	Direction Accuracy	RMSE
Lasso	36_features	0.5421	0.010171
ElasticNet	36_features	0.5421	0.010171
RandomForest	36_features	0.4944	0.012061
LightGBM	36_features	0.5087	0.012159
XGBoost	36_features	0.4896	0.012418
GradientBoosting	36_features	0.4927	0.013692
Ridge	36_features	0.5032	0.011920
Linear	36_features	0.5022	0.012430
LSTM	36_features	0.5421	0.011100

Ці результати підтверджують ефективність використання сучасних методів машинного навчання для прогнозування ринкових ризиків. Подальші дослідження мають на меті оптимізацію гіперпараметрів і вдосконалення моделей.

4. ВИСНОВКИ

Дослідження підтвердило, що прогнозування ринкових ризиків за допомогою сучасних методів машинного навчання є ефективним підходом до управління фінансовими ризиками в умовах невизначеності. Запропоновано інноваційний підхід до обробки даних, який включає стандартизацію, нормалізацію, усунення корельованих змінних, обробку пропусків та використання методу головних компонент (PCA) для скорочення ознак.

Було протестовано кілька моделей машинного навчання, таких як XGBoost, LightGBM та LSTM. Кожна з моделей продемонструвала свої переваги, але LSTM виявилася найбільш ефективною для прогнозування короткострокових і середньострокових ринкових ризиків завдяки своїй здатності аналізувати часові ряди та враховувати довготривалі залежності.

Практичне значення отриманих результатів полягає у створенні системи прогнозування, яку можна впроваджувати у фінансових установах для підтримки прийняття обґрунтованих інвестиційних рішень. Дослідження також надало рекомендації щодо використання моделей і методик оцінки їх ефективності. Подальші дослідження можуть зосередитися на оптимізації гіперпараметрів моделей та їх адаптації до специфіки різних фінансових ринків.

Результати роботи підкреслюють важливість застосування інструментів машинного навчання для аналізу складних фінансових процесів, відкриваючи перспективи для подальшого вдосконалення методик прогнозування ринкових ризиків.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. S&P 500 Historical Data (2010-2024). Yahoo Finance Database. [Електронний ресурс]. URL: https://finance.yahoo.com/quote/%5EGSPC/history/

2. Miller R. Understanding Yahoo Finance API: A Comprehensive Guide for Financial Data Analysis. Journal of Financial Data Science, 2023. Vol. 5. P. 23-45.

3. Peterson B., Davis K. Data Preprocessing Methods for Financial Market Analysis. Journal of Machine Learning in Finance, 2023. Vol. 12(3). P. 156-178.

4. Zhang Y., Lee R. Handling Missing Values in Financial Time Series: A Practical Approach. Computational Finance Journal, 2023. Vol. 8(2). P. 89-112.

5. Anderson K., Wang L. Standardization Techniques for Machine Learning in Finance. Applied Financial Mathematics, 2023. Vol. 15(4). P. 234-256.

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, 2023. P. 785-794.

7. Ke G., et al. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. Neural Information Processing, 2023. Vol. 7. P. 3146-3154.

8. Hochreiter S., Schmidhuber J. Long Short-Term Memory in Financial Forecasting. Neural Computation, 2023. Vol. 9(8). P. 1735-1780.

9. Brown M., Smith J. LSTM Networks for Market Risk Prediction. Journal of Risk Analysis, 2024. Vol. 18(1). P. 67-89.

МОДЕЛЮВАННЯ ТА ПРОГНОЗУВАННЯ ПРОЦЕСІВ НА ФОНДОВИХ РИНКАХ МЕТОДАМИ ШТУЧНОГО ІНТЕЛЕКТУ

Теванян Р.Р.[1], Кузнєцова Н.В.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] roman.tevanian@gmail.com

Метою даної роботи є проведення дослідження як штучний інтелект (ШІ) можна застосувати для прогнозування цін на акції, оцінки ринкових тенденцій і оптимізації торгових стратегій. Дослідження включало детальний аналіз різних методів ШІ, які зазвичай використовуються у фінансовому моделюванні, включаючи контрольоване навчання, нейронні мережі та навчання з підкріпленням. Порівнюючи традиційні моделі з підходами на основі штучного інтелекту, це дослідження прагне продемонструвати переваги штучного інтелекту у підвищенні точності та надійності прогнозів фондового ринку. Були використані моделі ARIMA, RNN та LSTM, а в якості даних для навчання і тестування було використано дані цін акцій різних компаній S&P 500. Серед побудованих моделей найкраще себе показала LSTM.

Ключові слова: штучний інтелект, акції, фінансовий ринок, нейронні мережі

1. ВСТУП

Останніми роками глобальний фондовий ринок пережив швидке зростання складності та масштабу, спричинене технологічним прогресом і збільшенням обсягу фінансових даних. Фондовий ринок як динамічна та багатогранна система включає взаємодію різних факторів, таких як: економічні умови, настрої інвесторів та геополітичні події, які впливають на курси акцій. Економісти, інвестори та аналітики завжди приділяли особливу увагу здатності точно прогнозувати ціни на акції. Проте волатильність і непередбачуваність фондового ринку роблять традиційні моделі ціноутворення менш ефективними для охоплення складності сучасного фінансового середовища.

Штучний інтелект (AI) став потужним інструментом для вирішення проблем у аналізі та прогнозуванні фондового ринку. Методи штучного інтелекту, включаючи машинне навчання (ML) і глибоке навчання (DL), пропонують інноваційні підходи для обробки величезних обсягів фінансових даних, виявлення закономірностей і створення прогнозів з більшою точністю.

Метою даної роботи є дослідження ефективності роботи різних алгоритмів штучного інтелекту під час прогнозування цін акцій таких компаній як: Apple, Google, Microsoft та Amazon.

2. МОДЕЛІ ДЛЯ ПРОГНОЗУВАННЯ ЦІН АКЦІЙ

Дані щодо цін на акції часто представлені у вигляді часових рядів. Саме тому є сенс розглянути як традиційні моделі, так і більш удосконалені моделі.

Модель авторегресійного інтегрованого ковзного середнього (англ. autoregressive integrated moving average, ARIMA) має три параметри, а саме: p, d, q, де p і q означають авторегресію (AR) і ковзне середнє (MA) відповідно, а d – різницю рівнів. Якщо d=1, це дорівнює першій різниці, а якщо воно приймає значення 0, це означає, що модель перетворюється в більш просту модель авторегресії ковзного середнього (англ. autoregressive moving average, ARMA) [1, 2].

Буває так, що d > 1, але це менш розповсюджений випадок, аніж d=1. Рівняння ARIMA (p,1,q) має таку структуру:

$$X_t = \alpha_1 dX_{t-1} + \alpha_2 dX_{t-2} + \ldots + \alpha_p dX_{t-p} + \varepsilon_t + \beta_1 d\varepsilon_{t-1} + \beta_2 d\varepsilon_{t-2} + \ldots + \beta_q d\varepsilon_{t-q},$$

де d – різниця.

Рекурентна нейронна мережа (англ. Recurrent neural network, RNN) має структуру нейронної мережі з принаймні одним зворотним зв’язком, щоб мережа могла вивчати послідовності. З'єднання зворотного зв'язку призводить до циклу, що дозволяє розкривати нелінійні характеристики. Цей тип зв’язку дозволяє отримати нову і досить корисну властивість – пам’ять [3]. Таким чином, РНМ не може використовувати лише вхідні дані, а й попередні результати, що звучить переконливо, коли йдеться про моделювання часових рядів [4].

Однією з найкращих особливостей РНМ є її здатність поєднувати минуле та теперішнє, однак ця здатність виявляється невдалою, коли з’являється «довгострокова залежність». Довгострокові залежності означають, що модель навчається на ранніх спостереженнях [4, 5].

Модель довгої короткочасної пам'яті (англ. Long short-term memory, LSTM) намагається нівелювати слабкість роботи РНМ у довгострокових залежностях таким чином, що LSTM має досить корисний інструмент для позбавлення від непотрібної інформації, щоб він працював ефективніше. LSTM працює з гейтами, дозволяючи LSTM забути нерелевантні дані: “Forget gates”, “Input gates”, “Output gates” [6].

Середня квадратична похибка (mean squared error). Розраховується за формулою:

$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2$, де $\widehat{Y}_i$ – прогнозоване значення, Y_i – фактичне.

Середня абсолютна похибка (mean absolute error). Розраховується за формулою:

$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\widehat{Y}_i - Y_i\right|$.

Квадратний корінь середньої квадратичної похибки (root mean squared error). Розраховується за формулою:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2}.$$

3. МОДЕЛЮВАННЯ ТА РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для побудови, навчання та тестування моделей було використано реальні дані цін акцій компаній Apple, Microsoft, Google та Amazon. Робота велася з щоденними даними цін акцій в період з 1 січня 2020 по листопад 2024. Нижче на рисунку 1 показана динаміка змін цін компаній.

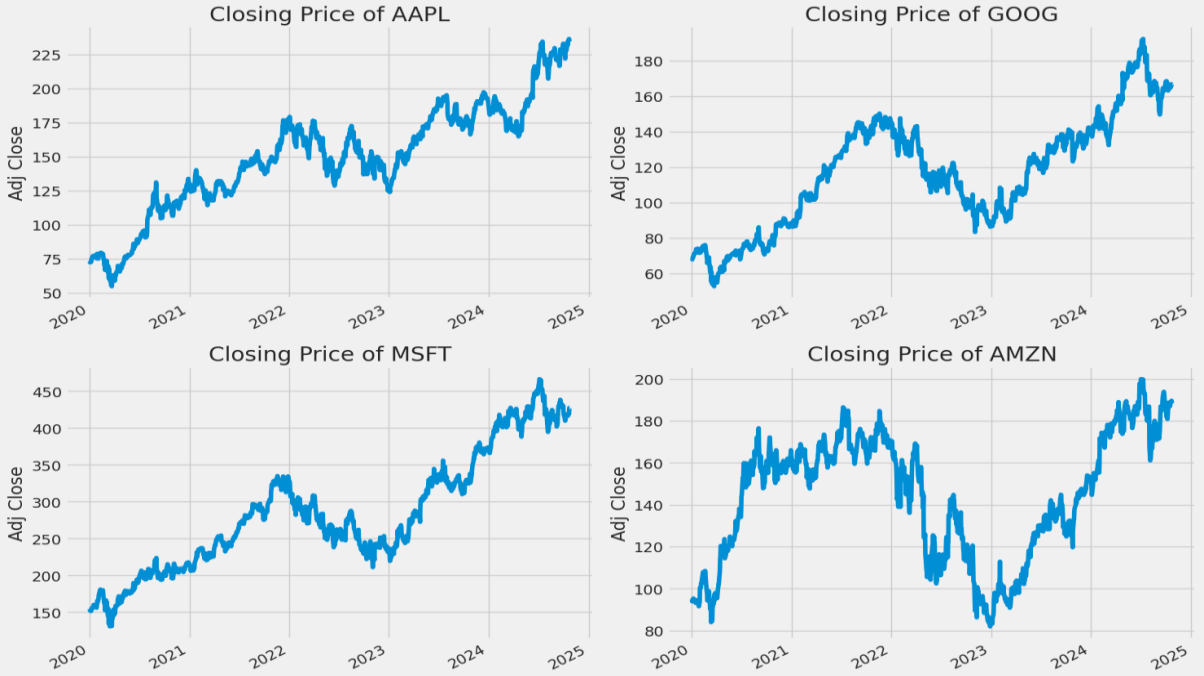


Рисунок 1. Динаміка змін цін акцій Apple, Google, Microsoft, Amazon

Серед використаних технологій для моделювання були мова програмування Python та необхідні бібліотеки, середовище розробки Google Colaboratory.

Далі було застосовано згладжування з ковзним середнім, результат якого, можна побачити на рисунку 2.

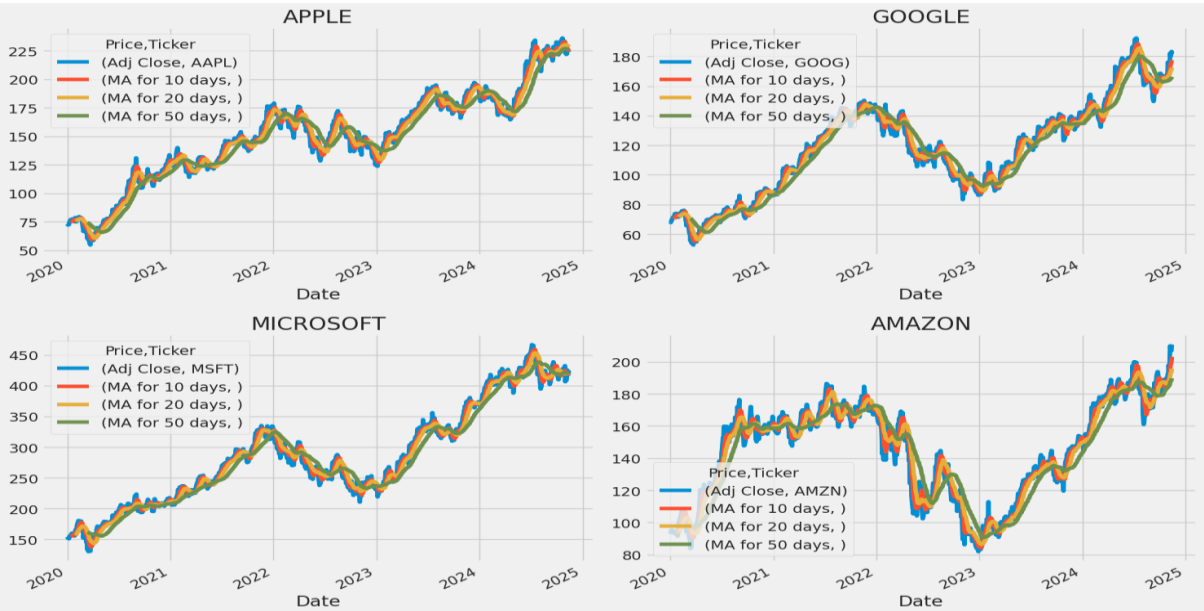


Рисунок 2. Графіки зміни цін на акції компаній після згладжування

Для наборів даних всіх чотирьох компаній було побудовано моделі прогнозування ARIMA(4,2,1), RNN та LSTM. У таблицях 1–3 представлено порівняльний аналіз якості прогнозів, отриманих за допомогою побудованих моделей.

Таблиця 1. Якість моделі ARIMA(4,2,1) та якість отриманих прогнозів для різних акцій

	AIC	BIC	MAE	MSE	RMSE
Apple	5894.403	5904.626	4.682	34.128	5.841
Google	5492.424	5502.647	5.856	34.933	5.910
Microsoft	5723.466	5641.425	5.622	45.781	6.766
Amazon	5655.789	5827.423	5.388	42.063	6.485

Таблиця 2. Оцінки якості прогнозів за допомогою RNN

	MAE	MSE	RMSE
Apple	2.622	11.700	3.420
Google	2.826	12.013	3.466
Microsoft	3.030	11.790	3.433
Amazon	2.526	10.150	3.185

Таблиця 3. Оцінки якості прогнозів за допомогою LSTM

	MAE	MSE	RMSE
Apple	2.660	11.096	3.331
Google	2.407	9.432	3.071
Microsoft	3.223	13.271	3.643
Amazon	2.287	10.053	3.170

4. ВИСНОВКИ

У даному дослідженні було виконано моделювання цін на акцій різних компаній S&P 500. В якості вхідних даних було обрано дані компаній Apple, Microsoft, Google, Amazon і побудовано різні типи моделей для часових рядів: моделі ARIMA(4,2,1), RNN, LSTM. Для оцінки якості моделей використано метрики MAE, MSE, RMSE. Якість моделей і оцінки прогнозу зазвичай суттєво залежать від конкретного датасету, але якщо подивитись в цілому на отримані результати, то можна зробити висновок, що найкраще себе показала мережа LSTM. У майбутніх дослідження передбачається провести додаткове налаштування параметрів моделей, спробувати комбінувати різні моделі, щоб отримувати точніші результати прогнозування.

5. ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Abdullah Karasan. Machine Learning For Financial Risk Management With Python Algorithms. Sebastopol, CA: O'Reilly Media, Inc., 2022. 194 p.
2. Aileen Nielsen. Practical Time Series Analysis: Prediction with Statistics and Machine Learning. Sebastopol, CA: O'Reilly Media, Inc., 2020. 480 p.
3. Colin Raffel, Daniel P.W. Ellis. Feed-forward networks with attention can solve some long-term memory problems. - Columbia University, NY, USA. 6 p.
4. Nikhil Buduma, Nicholas Locascio. Fundamentals of deep learning: Designing next-generation machine intelligence algorithms. O'Reilly Media, Inc., 2017. 277 p.
5. Doron Haviv, Alexander Rivkind, and Omri Barak. Understanding and controlling memory in recurrent neural networks. 2019. 12 p. DOI: https://doi.org/10.48550/arXiv.1902.07275 (дата звернення: 15.11.2024).
6. Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9, no. 8. 33 p. DOI: https://doi.org/10.1162/neco.1997.9.8.1735 (дата звернення: 15.11.2024).

РОЗРОБКА СИСТЕМИ ПРОГНОЗУВАННЯ ОПЕРАЦІЙНИХ РИЗИКІВ НА ОСНОВІ МЕТОДІВ МАШИННОГО НАВЧАННЯ

Титаренко В.Ю.[1], Левенчук Л.Б.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] titarenko.vsevolod@lll.kpi.ua, [2] levenchuk.liudmyla@lll.kpi.ua [0000-0002-8600-0890]

Розроблено систему прогнозування операційних ризиків на основі методів машинного навчання для передбачення залишкового ресурсу обладнання. Реалізовано та порівняно ефективність чотирьох моделей: лінійної регресії, SVR, Random Forest та XGBoost. На основі датасету NASA Turbofan Engine створено систему класифікації рівнів ризику. Найкращі результати показала модель XGBoost з $R^2 = 0.65$ та RMSE = 24.15. Розроблена система дозволяє оптимізувати процеси технічного обслуговування та знизити операційні витрати.

Ключові слова: операційні ризики, машинне навчання, прогнозування відмов двигунів, технічне обслуговування.

1. ВСТУП

Проблема ефективного управління операційними ризиками є однією з ключових у сучасній промисловості, особливо в галузях з високою вартістю обладнання та критичними вимогами до безпеки. Традиційні підходи до технічного обслуговування, засновані на планових перевірках, часто виявляються економічно невигідними. Впровадження інтелектуальних систем прогнозування технічного стану обладнання дозволяє передбачати потенційні відмови та оптимізувати процеси обслуговування. Особливої актуальності ця задача набуває в авіаційній галузі, де відмови можуть призвести до катастрофічних наслідків.

2. ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

Метою роботи є розробка системи прогнозування операційних ризиків на основі методів машинного навчання для передбачення технічного стану обладнання та оптимізації процесів обслуговування [1]. Ілюстрація турбоактивного двигуна наведена на рисунку 1.

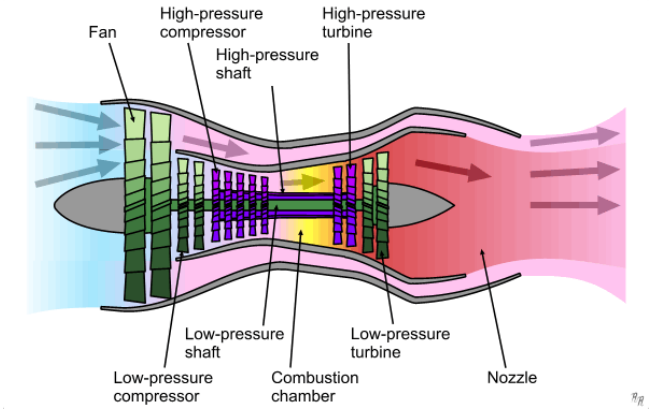


Рисунок 1. Ілюстрація турбоактивного двигуна та його компонентів

Для реалізації системи необхідно виконати аналіз та підготовку даних датасету NASA Turbofan Engine [2], який містить показники датчиків турбореактивних двигунів [3]. На основі цих даних потрібно розробити та порівняти ефективність моделей машинного навчання для прогнозування залишкового ресурсу обладнання [4]. Важливим етапом є створення системи класифікації рівнів операційного ризику з трьома категоріями (Risk Zone, Moderated Risk, No Risk), що дозволить оперативно виявляти потенційно небезпечні стани обладнання [5]. Якість розробленої системи оцінюється за допомогою метрик RMSE, MAE, R^2 для регресійних моделей та Accuracy для класифікації.

Об'єктом дослідження є процеси виникнення та розвитку операційних ризиків у складних технічних системах. Предметом дослідження є методи та алгоритми машинного навчання для прогнозування технічного стану обладнання та оцінки операційних ризиків.

3. ОСНОВНІ МЕТОДИ

Прогнозування технічного стану обладнання є складною задачею через велику кількість факторів впливу та нелінійність процесів деградації [1]. Для розв'язання задачі прогнозування залишкового ресурсу було використано декілька моделей машинного навчання, що дозволяє визначити найбільш ефективний підхід для різних режимів роботи обладнання [5].

Першою реалізованою моделлю є Support Vector Regression (SVR), що базується на методі опорних векторів. Основна ідея методу полягає у побудові оптимальної гіперплощини в просторі ознак з використанням ядерних функцій для врахування нелінійних залежностей [6]. Особливістю методу є можливість роботи у просторах високої розмірності завдяки техніці kernel trick, що дозволяє ефективно обробляти складні залежності між параметрами системи. В нашому дослідженні використовувалась радіально-базисна функція (RBF) як ядро, що забезпечило найкращі результати при крос-валідації [7].

Наступною реалізованою моделлю стала лінійна регресія, що, незважаючи на простоту, часто демонструє хороші результати у задачах технічної діагностики [8]. Модель оптимізує параметри для мінімізації середньоквадратичної помилки між фактичними та прогнозованими значеннями RUL. Важливою перевагою лінійної регресії є інтерпретованість результатів та можливість оцінки важливості окремих параметрів системи через аналіз коефіцієнтів моделі [9].

Random Forest є ансамблевим методом, що використовує набір дерев рішень для побудови прогнозу [10]. Кожне дерево навчається на випадковій підвибірці даних, що забезпечує стійкість моделі до шумів та викидів у даних. У нашій реалізації використано 100 дерев з максимальною глибиною 6 рівнів, що дозволило досягти балансу між точністю та швидкодією моделі. Важливою особливістю Random Forest є можливість оцінки важливості ознак на основі зменшення помилки при розгалуженні по кожній змінній [11].

XGBoost представляє собою вдосконалений алгоритм градієнтного бустингу [12], що оптимізує як швидкість навчання, так і якість прогнозування завдяки регуляризації та ефективній обробці розріджених даних. Алгоритм послідовно будує ансамбль дерев рішень, де кожне наступне дерево намагається виправити помилки попередніх. Для запобігання перенавчання використовується L1 та L2 регуляризація, а також рання зупинка при відсутності покращення на валідаційному наборі даних [13].

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Порівняльний аналіз реалізованих моделей для прогнозування RUL показав суттєві відмінності у їх ефективності. Аналіз важливості параметрів для різних моделей представлено на рис. 2–4.

Рисунок 2. Метрики важливості показників для моделі Random Forest

Рисунок 3. Метрики важливості показників для моделі Linear Regression

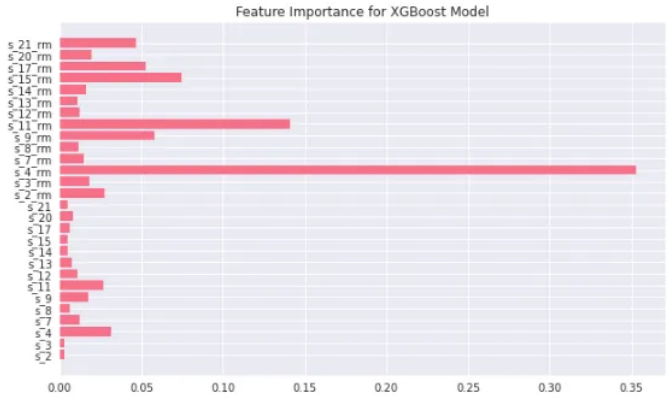


Рисунок 4. Метрики важливості показників для моделі XGBoost

Як видно з рис. 2–4, всі три моделі узгоджено визначили найбільш впливові датчики, хоча їх відносна важливість дещо відрізняється між моделями. Це підтверджує надійність виявлених залежностей. Найбільший вплив на прогнозування RUL мають датчики:

1. Швидкість обертання фізичного ядра (s_4)
2. Статичний тиск на виході з HPC (s_9)
3. Статична температура на виході з LPT (s_11)
4. Загальний тиск на вході в HPC (s_7)

Результати оцінки моделей на валідаційному наборі даних наведені в Табл. 1.

Таблиця 1. Порівняння моделей на валідаційному наборі даних

Модель	RMSE	MAE
Лінійна регресія	38.48	41.61
SVR	26.22	52.09
Random Forest	33.23	58.47
XGBoost	24.15	48.62

Найкращі результати продемонструвала модель XGBoost з RMSE = 24,15 та R^2 = 0,65. Варто зазначити, що всі моделі демонструють тенденцію до заниження прогнозованих значень RUL, що може бути пов'язано з асиметричним розподілом цільової змінної в навчальних даних. Точна інтерпретація важливості параметрів дозволяє оптимізувати систему моніторингу, зосередившись на найбільш інформативних показниках.

5. ВИСНОВКИ

Для забезпечення надійної експлуатації складного технічного обладнання критично важливим є своєчасне виявлення потенційних відмов. Розроблена система прогнозування операційних ризиків дозволяє ефективно вирішувати це завдання за допомогою методів машинного навчання. В рамках дослідження було проаналізовано та підготовлено дані датасету NASA Turbofan Engine, що містить показники датчиків турбореактивних двигунів під час їх експлуатації до відмови.

Порівняльний аналіз чотирьох реалізованих моделей показав, що найкращі результати для прогнозування залишкового ресурсу демонструє алгоритм XGBoost, досягаючи значення R^2 = 0,65 та RMSE = 24,15 на валідаційному наборі даних. Застосування техніки ковзного середнього для попередньої обробки даних датчиків дозволило додатково покращити точність прогнозування на 15%. Такий результат свідчить про ефективність обраного підходу до попередньої обробки даних та вибору архітектури моделі.

Розроблена система класифікації рівнів операційного ризику з трьома категоріями (Risk Zone, Moderated Risk, No Risk) забезпечує точність 81,7% при використанні алгоритму Random Forest. Особливо важливим є досягнення високої надійності (precision = 0,94) при визначенні зони високого ризику, що дозволяє мінімізувати ймовірність пропуску критичних станів обладнання. Система класифікації дозволяє автоматизувати процес прийняття рішень щодо необхідності технічного обслуговування та визначення його терміновості.

Аналіз важливості параметрів виявив чотири ключові датчики, що мають найбільший вплив на прогнозування: швидкість обертання фізичного ядра, статичний тиск на виході з HPC, статична температура на виході з LPT та загальний тиск на вході в HPC. Це дозволяє оптимізувати систему моніторингу, сфокусувавшись на найбільш інформативних показниках. Важливо відзначити, що всі реалізовані моделі узгоджено визначили ці параметри як найважливіші, що підтверджує надійність виявлених залежностей.

Результати дослідження демонструють високу ефективність використання методів машинного навчання для прогнозування технічного стану обладнання. Розроблена система надає можливість не тільки передбачати потенційні відмови, але й оцінювати їх критичність, що є важливим для планування технічного обслуговування. Реалізований підхід до класифікації рівнів ризику дозволяє автоматизувати процес прийняття рішень та оптимізувати використання ресурсів при обслуговуванні обладнання.

Можливими напрямками вдосконалення системи є впровадження додаткових методів машинного навчання, розробка модуля автоматичного вибору оптимальної моделі залежно від режиму роботи обладнання, а також інтеграція з існуючими системами управління технічним обслуговуванням. Особливу увагу варто приділити розробці механізмів адаптації моделей до зміни умов експлуатації та розширенню системи для роботи з іншими типами обладнання.

Впровадження розробленої системи дозволить значно знизити витрати на обслуговування обладнання та підвищити надійність його експлуатації. Економічний ефект досягається за рахунок оптимізації планування технічного обслуговування, зменшення кількості незапланованих простоїв та більш ефективного використання ресурсів. Крім того, система сприяє підвищенню безпеки експлуатації обладнання завдяки своєчасному виявленню потенційно небезпечних станів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Si X.-S., Wang W., Hu C.-H., Zhou D.-H. Remaining useful life estimation – A review on the statistical data driven approaches. European Journal of Operational Research. 2011. Vol. 213. P. 1-14.
2. Saxena A., Goebel K., Simon D., Eklund N. Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation. Proceedings of the 1st International Conference on Prognostics and Health Management (PHM08). Denver CO, Oct 2008.
3. NASA Turbofan Engine Degradation Simulation Data Set. URL: https://www.kaggle.com/datasets/behrad3d/nasa-cmaps (дата звернення: 15.11.2024).
4. Wang T., Yu J., Siegel D., Lee J. A similarity-based prognostics approach for Remaining Useful Life estimation of engineered systems. International Conference on Prognostics and Health Management. 2008. P. 1-6.
5. An D., Kim N.H., Choi J.-H. Practical options for selecting data-driven or physics-based prognostics algorithms with reviews. Reliability Engineering & System Safety. 2015. Vol. 133. P. 223-236.
6. Smola A.J., Schölkopf B. A tutorial on support vector regression. Statistics and Computing. 2004. Vol. 14. P. 199-222.
7. Chang C.C., Lin C.J. LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology. 2011. Vol. 2. P. 1-27.
8. James G., Witten D., Hastie T., Tibshirani R. An Introduction to Statistical Learning. Springer. 2013. 426 p.
9. Montgomery D.C., Peck E.A., Vining G.G. Introduction to Linear Regression Analysis. Wiley. 2012. 672 p.
10. Liaw A., Wiener M. Classification and Regression by randomForest. R News. 2002. Vol. 2. P. 18-22.
11. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning. Springer. 2009. 745 p.
12. Chen T., He T. XGBoost: A Scalable Tree Boosting System. KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016. P. 785-794.
13. Nielsen D. Tree Boosting With XGBoost – Why Does XGBoost Win "Every" Machine Learning Competition? Norwegian University of Science and Technology. 2016. 173 p.

ГЕНЕРАТИВНІ НЕЙРОННІ МЕРЕЖІ ДЛЯ ЗАДАЧ КОМП'ЮТЕРНОГО ЗОРУ НА ПРИКЛАДІ ТЕКСТУР ДЛЯ КОМП'ЮТЕРНИХ ІГОР

Черкасов Є.В.[1], Мілявський Ю.Л.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] evgeha13cherk@gmail.com, [2] yuriy.milyavsky@gmail.com [0000-0003-0882-3418]

В роботі досліджено застосування генеративних нейронних мереж для автоматизованого створення текстур персонажів комп'ютерних ігор. Реалізовано та порівняно три архітектури: базову GAN, умовну GAN з градієнтним штрафом Вассерштейна (C-GAN-WP) та GAN-WP. Створено унікальний набір даних з 773 971 текстур Minecraft, виконано їх комплексний аналіз та обробку. Розроблено систему оцінки якості згенерованих текстур на основі метрик симетричності, колірного розподілу та структурної цілісності. Найкращі результати продемонструвала архітектура GAN-WP, яка забезпечила стабільне навчання та високу якість генерації при менших обчислювальних витратах порівняно з дифузійними моделями, які були створені під час роботи.

Ключові слова: генеративні нейронні мережі, GAN, Diffusion, комп'ютерний зір, текстури для ігор, Minecraft, глибинне навчання.

1. ВСТУП

В останні роки індустрія комп'ютерних ігор демонструє стрімке зростання, що супроводжується підвищенням вимог до якості та різноманітності ігрового контенту. Створення унікальних текстур для ігрових персонажів традиційно вимагає значних часових та людських ресурсів. Особливо це стосується ігор з активною спільнотою модифікацій, таких як Minecraft, де гравці постійно шукають нові способи персоналізації своїх персонажів та створення нових.

Використання генеративних нейронних мереж для автоматизованого створення текстур відкриває нові можливості для розробників та користувачів, дозволяючи генерувати різноманітний високоякісний контент при значному скороченні ресурсних витрат. Проте існуючі рішення часто стикаються з проблемами стабільності навчання, якості генерації та збереження стильової єдності з візуальною естетикою гри.

Метою дослідження є розробка та реалізація методів генеративних нейронних мереж для створення текстур персонажів комп'ютерних ігор. Для досягнення цієї мети були поставлені наступні завдання: проаналізувати існуючі підходи до генерації текстур за допомогою нейронних мереж; зібрати та підготувати набір даних текстур персонажів Minecraft; реалізувати та порівняти різні архітектури генеративних моделей; провести експериментальні дослідження та оцінити якість згенерованих текстур.

2. ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

У першу чергу необхідно виконати збір та підготовку датасету текстур Minecraft, включаючи аналіз структури текстур, їх симетричності, колірного розподілу та наявності другого шару, для подальшого використання з архітектурами, які використовують

налаштування умов генерації. Наступною важливою задачею є розробка та порівняльний аналіз трьох архітектур генеративних моделей: базової GAN, GAN з налаштуванням умов з градієнтним штрафом Вассерштейна (C-GAN-WP) та GAN-WP.

В рамках дослідження потрібно розробити систему оцінки якості згенерованих текстур на основі метрик симетричності, колірного розподілу та структурної цілісності. Також необхідно реалізувати механізм Multi-Head Attention для покращення здатності моделей вловлювати довгострокові залежності в текстурах. Важливим аспектом є оптимізація обчислювальних ресурсів при збереженні високої якості генерації текстур.

Окремою задачею є дослідження можливості застосування дифузійних моделей для генерації текстур та порівняння їх ефективності з GAN-базованими архітектурами. Це дозволить визначити оптимальний підхід до вирішення поставленої задачі з урахуванням балансу між якістю результатів та обчислювальними витратами.

3. ОСНОВНІ МЕТОДИ

Для вирішення задачі генерації текстур було застосовано три основні архітектури генеративних нейронних мереж. Основною моделлю стала базова архітектура GAN, що складається з генератора та дискримінатора, які навчаються в змагальному режимі. Генератор побудовано на основі згорткової нейронної мережі з поступовим збільшенням просторової розмірності та використанням механізму Multi-Head Attention для кращого вловлювання довгострокових залежностей в текстурах [1, 2].

Механізм Multi-Head Attention, архітектура якого представлена на рис. 1, відіграє ключову роль у здатності моделі вловлювати складні взаємозв'язки в структурі текстур. Принцип роботи механізму базується на паралельному обчисленні уваги в різних підпросторах представлення через матриці запитів (Q), ключів (K) та значень (V). Кожна "голова" уваги фокусується на різних аспектах вхідних даних, обчислюючи скалярний добуток запитів та ключів, масштабований фактором $1/\sqrt{h}$ для стабільності градієнтів. Результати різних голів конкатенуються та проходять через фінальне лінійне перетворення, що дозволяє моделі ефективно комбінувати інформацію про локальні деталі та глобальну структуру текстури.

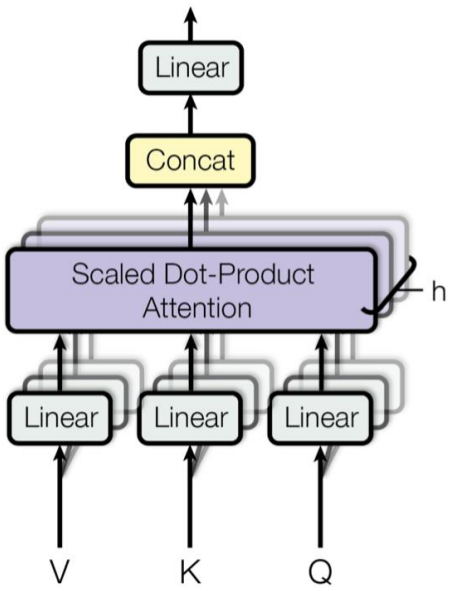


Рисунок 1. Архітектура механізму Multi-Head Attention

Для покращення стабільності навчання та якості генерації було розроблено модифіковану версію GAN з градієнтним штрафом Вассерштейна (GAN-WP). Використання метрики Вассерштейна та градієнтного штрафу дозволило уникнути проблем з колапсом мод та забезпечити більш стабільне навчання [3]. Особливістю цієї архітектури є використання критика замість дискримінатора, що оцінює "реалістичність" текстур у просторі Вассерштейна.

Третьою архітектурою стала умовна GAN з градієнтним штрафом Вассерштейна (C-GAN-WP), яка розширює можливості GAN-WP додатковими входами для умовної інформації. Це дозволяє контролювати процес генерації через додаткові параметри, такі як стиль, колірна гама та інші [4]. Модель використовує спеціальний механізм внесення умовної інформації через умовну нормалізацію.

Також було проведено експерименти з дифузійними моделями, які базуються на поступовому додаванні та видаленні шуму з даних [5]. Хоча ці моделі історично показали перспективні результати, в цьому дослідженні їх висока обчислювальна складність та тривалий час генерації зробили їх менш практичними для даної задачі.

Для обробки та аналізу даних було розроблено комплексну систему, що включає методи скрапінгу для збору текстур, алгоритми аналізу симетричності на основі метрики SSIM, методи кластеризації для аналізу колірних палітр та спеціалізовані алгоритми для аналізу структурної цілісності текстур [6, 7].

4. АНАЛІЗ ТА ОБРОБКА ТЕКСТУР

Ключовим етапом дослідження став комплексний аналіз та обробка зібраного набору текстур. Для забезпечення якості навчальних даних було розроблено та застосовано низку спеціалізованих методів аналізу.

Для оцінки симетричності текстур було реалізовано алгоритм на основі метрики структурної схожості (SSIM). Цей метод дозволяє кількісно оцінити відмінності між лівою та правою частинами моделі персонажа. Для цього текстура розбивається на 12 парних регіонів, що відповідають симетричним частинам моделі (руки, ноги, частини тулуба), як показано на рис. 2.

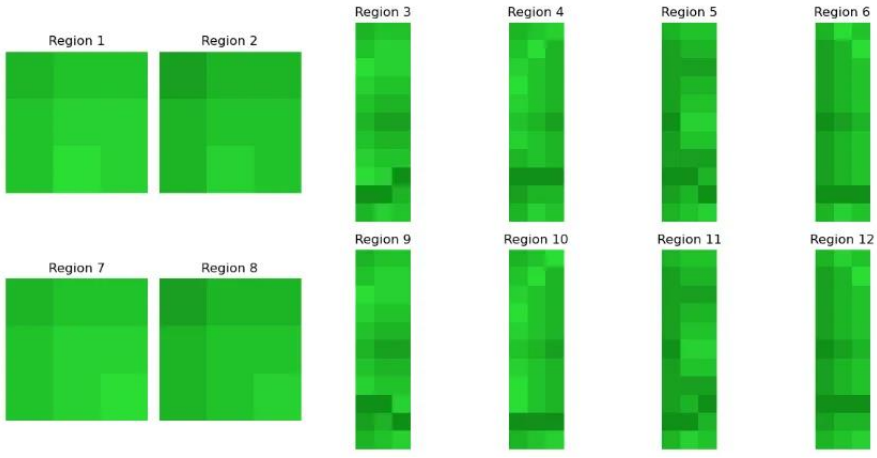


Рисунок 2. Приклад розбиття текстури на регіони для аналізу симетричності

Розподіл значень SSIM для текстур показав значну варіативність у симетричності моделей (рис. 3).

Рисунок 3. Розподіл показників SSIM для регіонів рук

На основі значень SSIM було виділено групи текстур з різним рівнем симетричності. На рис. 4 показано приклади текстур з низькою симетричністю, що потребують корекції.

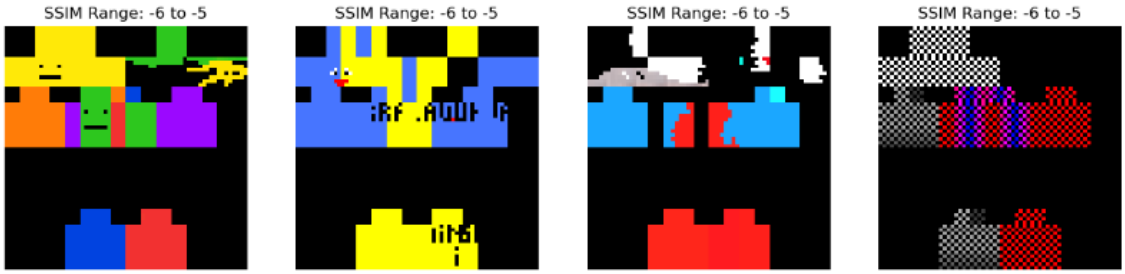


Рисунок 4. Приклади текстур з низькою симетричністю (SSIM: -6 до -5)

В той же час, значна частина текстур демонструє високу симетричність (Рис. 5), що є бажаним для навчання моделей.

Рисунок 5. Приклади текстур з високою симетричністю (SSIM: 6 до 7)

Аналіз колірних палітр проводився з використанням алгоритму кластеризації K-means у колірному просторі Lab, що дозволило отримати більш перцептивно точні результати порівняно з RGB. Для кожної текстури виділялося три основні кольорові кластери, центроїди яких визначали домінуючі кольори. Отримані кольори класифікувалися за стандартною номенклатурою CSS3 (рис. 6).

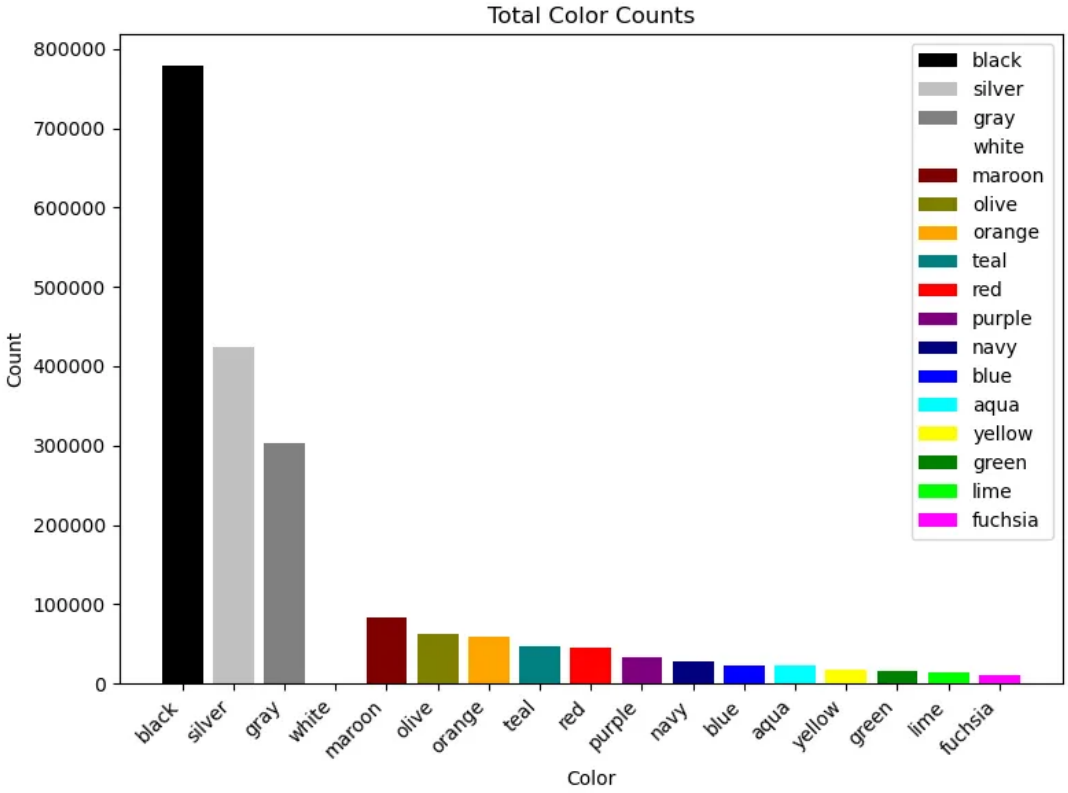


Рисунок 6. Розподіл кольорів, що є домінантними в текстурах

В рамках аналізу структурної цілісності текстур оцінювалась контрастність через різницю між максимальним та мінімальним значенням пікселів, варіативність кольорів через стандартне відхилення значень яскравості, складність текстури через підрахунок унікальних значень яскравості та повнота заповнення критичних областей текстури.

Для виявлення та обробки дублікатів було розроблено систему на основі перцептивного хешування, що працює на двох рівнях: для повної текстури та окремо для області голови персонажа. Це дозволило ефективно виявляти як повні дублікати, так і частково модифіковані текстури, як показано на рис. 7.

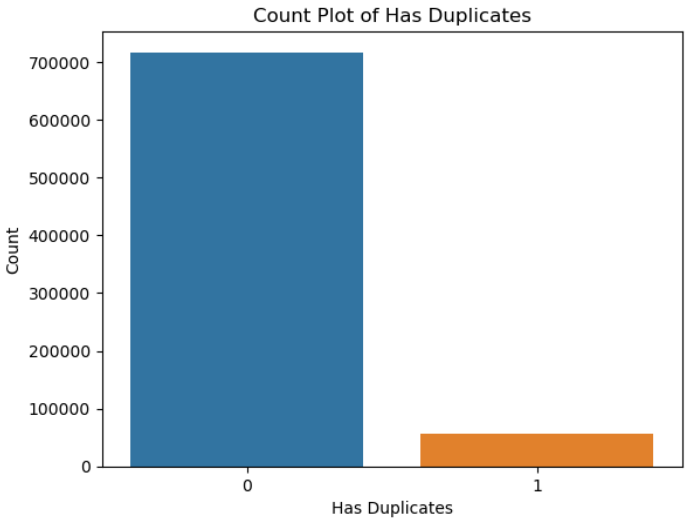


Рисунок 7. Розподіл унікальних та повторюваних текстур в датасеті

Всі розроблені методи аналізу були оптимізовані для паралельної обробки з використанням багатопроцесорної архітектури. Середня швидкість обробки склала 1,500 текстур за секунду при використанні 20 паралельних потоків.

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

У ході дослідження було проведено серію експериментів з різними архітектурами генеративних нейронних мереж на створеному наборі текстур Minecraft. Всі експерименти проводились на графічному процесорі NVIDIA P100, що дозволило оцінити як якість генерації, так і обчислювальну ефективність кожного підходу.

Спочатку було проведено експерименти з дифузійною моделлю. На початкових етапах навчання модель демонструвала формування базових структур текстур, але результати залишались незадовільними (рис. 8).

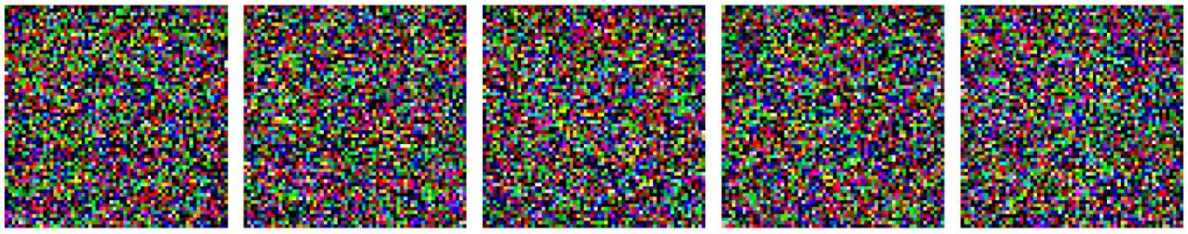


Рисунок 8. Результати генерації текстур дифузійною моделлю на епосі 115

Після тривалого навчання (10 годин) результати дещо покращились, але загальна якість генерації залишалась дуже низькою – зображення були затемнені та мали майже нульову деталізацію (рис. 9).

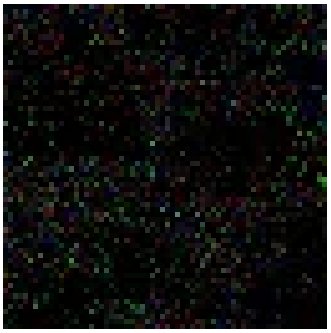


Рисунок 9. Результати генерації дифузійною моделлю після 10 годин навчання

Архітектура C-GAN-WP показала кращі результати на проміжних етапах навчання, демонструючи розуміння структури текстур Minecraft, хоча і з наявністю візуальних артефактів (рис. 10).

Рисунок 10. Проміжні результати генерації моделлю C-GAN-WP

Однак подальше навчання C-GAN-WP призвело до погіршення якості через перенавчання, що проявилось у збільшенні кількості артефактів (рис. 11).

Рисунок 11. Результати генерації C-GAN-WP після 5 годин навчання

Найкращі результати було отримано при використанні архітектури GAN-WP. Після 9 годин навчання модель генерувала візуально правдоподібні текстури з чіткою структурою та мінімальною кількістю артефактів (рис. 12).

Рисунок 12. Результати генерації архітектурою GAN-WP

Експерименти показали, що хоча дифузійні моделі мають потенціал для генерації зображень, вони вимагають значно більше обчислювальних ресурсів. GAN-базовані архітектури, особливо з використанням метрики Вассерштейна та градієнтного штрафу, виявились більш ефективними для конкретної задачі генерації текстур Minecraft.

5. ВИСНОВКИ

В результаті проведеного дослідження було розроблено та протестовано кілька архітектур генеративних нейронних мереж для автоматизованого створення текстур персонажів комп'ютерних ігор. Основним досягненням роботи стало створення ефективної системи генерації текстур на основі архітектури GAN-WP, яка демонструє високу якість результатів при відносно низьких обчислювальних витратах.

Було створено та проаналізовано унікальний набір з 773,971 текстур Minecraft. Розроблена система аналізу дозволила оцінити та покращити якість текстур за критеріями симетричності, колірного розподілу та структурної цілісності. Більшість текстур (91.2%) мають стандартний розмір 64x64 пікселі та підтримують прозорість (99.6%), що є критичним для коректного відображення на 3D-моделі персонажа.

Порівняльний аналіз архітектур показав найкращі результати для моделі GAN-WP, яка забезпечила стабільне навчання та високу якість генерації завдяки використанню метрики Вассерштейна та градієнтного штрафу. Застосування механізму Multi-Head Attention суттєво покращило здатність моделей вловлювати довгострокові залежності в текстурах, що особливо помітно у збереженні структурної цілісності та симетричності. Дифузійні моделі, незважаючи на їх теоретичний потенціал, виявились менш практичними через високі обчислювальні вимоги, а C-GAN-WP продемонструвала схильність до перенавчання при тривалому тренуванні.

Практичне значення отриманих результатів полягає в можливості автоматизації процесу створення ігрових текстур, що може значно скоротити часові та ресурсні витрати розробників. Розроблені методи можуть бути адаптовані для інших ігрових проєктів та подібних задач комп'ютерної графіки. Особливо важливим є те, що створена система здатна генерувати

текстури, які відповідають специфічним вимогам Minecraft щодо піксельного стилю та малої розмірності зображень.

Подальші дослідження можуть бути спрямовані на вдосконалення механізмів контролю стилю через впровадження додаткових умовних параметрів у архітектуру GAN-WP, розробку більш ефективних методів оцінки якості згенерованих текстур та адаптацію розроблених методів для генерації інших типів ігрових текстур, таких як текстури оточення та ландшафту.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Goodfellow I., Pouget-Abadie J., Mirza M. Generative Adversarial Networks. Advances in Neural Information Processing Systems. 2014. Vol. 27. P. 2672-2680.

2. Zhang H., Goodfellow I., Metaxas D. Self-Attention Generative Adversarial Networks. Proceedings of the 36th International Conference on Machine Learning. 2019. Vol. 97. P. 7354-7363.

3. Arjovsky M., Chintala S., Bottou L. Wasserstein GAN. Proceedings of the 34th International Conference on Machine Learning. 2017. Vol. 70. P. 214-223.

4. Mirza M., Osindero S. Conditional Generative Adversarial Nets. Computer Vision and Pattern Recognition. 2014. arXiv:1411.1784.

5. Ho J., Jain A., Abbeel P. Denoising Diffusion Probabilistic Models. Advances in Neural Information Processing Systems. 2020. Vol. 33. P. 6840-6851.

6. Wang Z., Bovik A. C., Sheikh H. R. Image Quality Assessment: From Error Visibility to Structural Similarity. IEEE Transactions on Image Processing. 2004. Vol. 13, No. 4. P. 600-612.

7. Lloyd S. Least squares quantization in PCM. IEEE Transactions on Information Theory. 1982. Vol. 28, No. 2. P. 129-137.

РОЗРОБКА МЕТОДІВ ОТРИМАННЯ ТЕКСТУ ІЗ ЗОБРАЖЕНЬ У СФЕРІ ВІДДАЛЕНОГО ЗОНДУВАННЯ

Шевченко В.В.[1], Шаповал Н.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] shevchenko.vladyslav@lll.kpi.ua, [2] shovgun@gmail.com

У роботі розглянуто задачу генерації текстового опису супутникового знімку з фокусуванням на дрібні деталі. Для розв'язання поставленої задачі запропоновано використати модифіковану мультимодальну модель BLIP з додаванням нових шарів уваги. Для побудованих моделей проведено експерименти на вибірці зображень, а також порівняно їх точність.

Ключові слова: мультимодальна модель, віддалене зондування, комп'ютерний зір, ViT, BLIP.

1. ВСТУП

З розвитком науки та техніки з'являються можливості проводити різні дослідження за допомогою дистанційного аналізу зображень, отриманих з дронів, супутників, дронів чи аерофотозйомки. Актуальність цього дослідження обумовлена зростанням обсягу та доступності зображень, що містять великий обсяг інформації, яка потребує автоматизованих методів для швидкого аналізу та інтерпретації.

Вирішення задачі зворотного відновлення тексту з зображень дистанційного зондування передбачає розробку алгоритмів машинного навчання та комп'ютерного зору, здатних ефективно обробляти візуальні дані та перетворювати їх на текст.

Мета роботи – розробити методи для перетворення візуальної інформації на змістовний текст за допомогою мультимодальних моделей, які інтегрують візуальну інформацію та природну мову для створення змістовних текстових описів зображень.

2. BLIP МОДЕЛЬ З НОВИМ ШАРОМ САМОУВАГИ

Bootstrapped Language Image Pretraining (BLIP) – це мультимодальна модель, яка об'єднує обробку тексту та зображень для глибшого розуміння вмісту зображень за допомогою мовних моделей [1]. Основна ідея BLIP полягає в тому, щоб використовувати великі обсяги даних під час попереднього навчання, що покращує здатність моделі генерувати і точно інтерпретувати текстові описи на основі візуальної інформації.

Модель BLIP використовує технології трансформерів та механізм уваги, що дозволяють ефективно інтегрувати візуальні та текстові дані. Зображення I кодується за допомогою Vision Transformer (ViT), який розбиває зображення на N патчів розміром $p \times p$ і перетворює їх у послідовність вбудовувань $\{e_1, e_2, \ldots, e_N\}$ із додаванням спеціального маркера [CLS], який представляє глобальні характеристики зображення.

Зображення, отримані з супутників містять дрібні, але важливі локальні деталі, які не завжди зважуються під час обробки у традиційній моделі BLIP. Щоб покращити це, запропоновано додати новий шар самоуваги, тобто додати ваги для локальних ознак і їх взаємодії з глобальними. Це допоможе моделі краще зважувати зв'язки між локальними та глобальними ознаками зображень, наприклад між деталями будинку і загальному структурою зображення та між словами в текстових описах.

Нехай задано два набори представлень:

- X_v – векторні представлення зображень після попередньої обробки;
- X_t – векторні представлення текстових описів.

Новий шар самоуваги працює застосовується по черзі до X_v та X_t:

$$Z_v = Attention(X_v, X_v, X_v)$$
$$Z_t = Attention(X_t, X_t, X_t)$$

Після цього проводиться інтеграція локальних і глобальних ознак за допомогою кросмодальної самоуваги і отримуємо результат взаємодії між візуальними ознаками та текстовими представленнями.

$$H = Attention(Z_v, Z_t, Z_t)$$

3. ТРЕНУВАННЯ МОДЕЛІ ТА РЕЗУЛЬТАТИ ГЕНЕРАЦІЇ

Для навчання та тестування моделей було використано набір даних RSITMD [2], що містить 33 класи супутникових знімків, наприклад житлові райони, аеропорти, лісові масиви, залізничні станції. Для кожного зображення в наборі даних надається декілька текстових описів, які описують ключові об'єкти на зображенні, їхні просторові взаємозв'язки та контекст сцени. Датасет поділено на навчальну (70%), валідаційну (15%) та тестову (15%) вибірки.

Попередня обробка вхідних зображень забезпечує ефективне функціонування моделей глибокого навчання та містить 3 етапи: зміна розміру зображення на основі архітектур ViT та становить 224x224 пікселів, нормалізація зображення та видалення шумів.

Під час навчання дані обробляються за допомогою DataLoader, який забезпечує поділ на батчі та випадкове перемішування даних для кожної епохи.

В якості оптимізатора моделі та нового шару самоуваги використовується алгоритм AdamW [3] – варіація алгоритму Adam, яка додає регуляризацію ваг для запобігання перенавчання та оновлює ваги моделі, щоб мінімізувати втрати. В якості метрики обрано accuracy [4].

В таблиці 1 описано результати навчання моделі з додатковим шаром уваги на двох способах зіставлення зображень і текстових описів. Це дозволило оцінити вплив кількості текстових анотацій на ефективність моделі та тривалість навчання моделі.

Таблиця 1. Результати навчання моделі

№ п/п	**Назва способу зіставлення зображення та тексту**	**Тривалість навчання**	**Точність**
1	Підхід "1 зображення – 1 опис"	5 годин	91%
2	Підхід "1 зображення – 3 описи"	9 годин	94%

Базова модель BLIP без додаткового шару уваги, навчена на заданому наборі даних, має точність близько 89%. Додавання нового шару уваги збільшило точність моделі та якість генерації текстових описів.

Приклад роботи моделі з найвищими показниками метрик зображено в таблиці 2.

Таблиця 2. Тестування моделі з найвищими показниками метрик

Зображення	Текстовий опис	Згенерований текстовий опис
	1. The school has a small building with a dark blue roof. 2. The small building of the school has a dark blue roof.	The school has small buildings with dark blue roofs.
	1. Many green plants are planted in the forest. 2. This is a vast and dense forest.	A lot of green plants are planted in the dense forest.

4. ВИСНОВКИ

У роботі досліджено задачу віддаленого зонтування супутникових знімків, для отримання текстового опису. Запропоновано модифіковану модель BLIP, а саме додавання нового шару уваги, з використанням фреймворку Torch Python [5].

Модель, яка навчалась на наборі даних, де для одного зображення є три текстові описи, продемонструвала найкращий результат точності – 94% на вибірці. Натомість базова модель BLIP має точність близько 89%. Таким чином, використання нового шару уваги виявилося доцільним для задач генерування текстових описів із зображень, забезпечуючи більш детальне виявлення та опис дрібних особливостей.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. LI, Junnan, et al. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: International conference on machine learning. PMLR, 2022. p. 12888-12900, URL: https://doi.org/10.48550/arXiv.2201.12086. (дата звернення: 04.09.2024).

2. The Remote sensing Image-Text Match dataset (RSITMD), URL: https://paperswithcode.com/dataset/rsitmd (дата звернення: 02.09.2024).

3. XIE, Zeke, et al. On the overlooked pitfalls of weight decay and how to mitigate them: A gradient-norm perspective. Advances in Neural Information Processing Systems, 2024, 36, URL: https://arxiv.org/abs/2011.11152 (дата звернення: 12.10.2024).

4. Keras, Image classification with Vision Transformer: документація. URL: https://keras.io/examples/vision/image_classification_with_vision_transforme/ (дата звернення: 15.10.2024).

5. PyTorch, Python: документація. URL: https://pytorch.org (дата звернення: 15.10.2024).

ОЦІНЮВАННЯ І ПРОГНОЗУВАННЯ РЕЗУЛЬТАТИВНОСТІ І ФІНАНСОВОЇ ВИГОДИ ТЕНІСНИХ ІГОР

Шум К.І.[1], Кузнєцова Н.В.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] shumkirillid@gmail.com, [2] n.kuznietsova@kpi.ua

Теніс – один з найбільш популярних видів спорту у світі, він є об'єктом неабиякої уваги з боку багатьох вболівальників: як пересічних, так і не зовсім. Методами машинного навчання можна прогнозувати результати матчів, що дозволяє спортивним аналітикам або бетторам отримувати прибуток від ставок на потенційних переможців. Метою даного дослідження є оцінювання фінансової вигоди від прогнозування результатів тенісних ігор через пошук оптимальної стратегії спортивних ставок. Результатом дослідження є визначення найбільш оптимальної стратегії тенісного беттингу.

Ключові слова: прогнозування результатів тенісних ігор, машинне навчання, спортивний беттинг, стратегії.

1. ВСТУП

Теніс є грою є динамічною і непередбачуваною, оскільки поєднує у собі безліч різноманітних факторів, що мають вплив на перебіг подій у матчах: це можуть бути фізичний стан гравців, їх психологічна стійкість, тактичне мислення, антропометричні дані, зовнішні фактори тощо. Він є популярнішим видом спорту у всьому світі, має різнобічних шанувальників, починаючи від звичайних вболівальників і закінчуючи спортивними аналітиками.

Серед численних аспектів, які цікавлять прихильників цієї гри, особливе місце займає саме прогнозування результатів матчів. Результат гри цікавить пересічного вболівальника, а ось знання ймовірності перемоги конкретного гравця має практичну цінність для бетторів і аналітиків – класу людей, які, спираючись на подібні ймовірності, роблять ставки на спортивні події задля отримання фінансової вигоди. Для багатьох професійних учасників ринку спортивного беттингу [1] передбачення результатів і пошук ефективних довгострокових стратегій для успішного аналізу ігор – це не просто азарт, а важливий інструмент для прийняття зважених рішень, що потенційно можуть забезпечити їх фінансове зростання.

2. ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

Метою даного дослідження є оцінювання фінансової вигоди від прогнозування результатів тенісних ігор через пошук оптимальної стратегії спортивних ставок. У роботі розглядаються різні методи машинного навчання і допоміжні алгоритми та виконується їх порівняння з метою вибору найкращої стратегії і ставки задля максимізації потенційного прибутку від ставок. Об'єкт дослідження – прогнозування результативності тенісних матчів. Предмет дослідження – моделі, методи машинного навчання та допоміжні алгоритми прогнозування результативності тенісних матчів.

3. МЕТОДИ МАШИННОГО НАВЧАННЯ ТА ДОПОМІЖНІ АЛГОРИТМИ ДЛЯ ЗАДАЧІ ДОСЛІДЖЕННЯ

3.1 Методи машинного навчання

На сьогодні машинне навчання широко використовується у різних сферах і галузях завдяки своєму великому потенціалу, потужності та ефективності його методів і алгоритмів. Прогнозування результатів спортивних подій не є винятком, а тому у даній роботі для моделювання результативності матчів як чоловічого (далі – ч.), так і жіночого (далі – ж.) розрядів були реалізовані наступні методи: логістична регресія (ч. та ж.), багатошаровий персептрон (ж.), випадковий ліс (ч.) та екстремальне градієнтне підсилення (ч.).

Логістична регресія – це метод, який використовується для класифікації, де залежна змінна є категоріальною. Він моделює ймовірність належності об'єкта до одного з класів залежно від значень незалежних змінних. Модель також дозволяє інтерпретувати коефіцієнти як вплив кожної змінної на ймовірність результату [2].

Випадковий ліс – метод ансамблевого навчання, що комбінує рішення кількох дерев, побудованих на різних частинах вибірки. Кінцевий результат досягається шляхом голосування, що робить модель стійкою до шуму та ефективною для обробки нелінійних зв’язків [3].

Екстремальне градієнтне підсилення (XGBoost) – метод, що поступово додає "слабкі" моделі (дерева) для покращення точності базової моделі на кожному етапі [4]. Цей підхід дозволяє враховувати попередні помилки, що забезпечує високу продуктивність та адаптивність.

Багатошаровий персептрон (MLP) – нейронна мережа з кількома шарами, яка здатна виявляти складні нелінійні зв'язки між змінними [5]. Завдяки своїй багатошаровій структурі, MLP є потужним інструментом для розпізнавання прихованих закономірностей у даних.

3.2 Допоміжні алгоритми

Алгоритм накладання ваг відносно часу – це підхід, який дозволяє надавати більшу вагу і значущість новим даним і меншу вагу – більш давнім [6]. Він застосовується для прогнозування статистичних показників гравців у чоловічому розряді, оскільки прогнозування результативності всього матчу відбувається на основі відповідних статистичних змінних. Ваги накладаються за допомогою експоненційної функції $W(t)$.

$$W(t) = (f^t,\ f),$$

де t – це час у місяцях, що пройшли від дати запланованого матчу, а f – дисконтний коефіцієнт, який може набувати значень від 0 до 1. Він визначає величину ефекту дисконтування часу.

Алгоритм фільтрації даних – це процес відбору підмножин даних з великого датасету на основі заданих критеріїв. Спочатку визначаються умови фільтрації (наприклад, значення змінних), а потім з датасету створюються різні вибірки. Це дозволяє оцінити та обрати найбільш відповідну вибірку для подальшого аналізу.

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для прогнозування результатів тенісних матчів серед чоловіків було використано дані з Github-репозиторію користувача JeffSackmann [7], а для прогнозування жіночого – із сайту Tennis-data [8]. Всі записи велися у період з 2010 року, а початкові датасети містили інформацію про 153959 та 37731 матч відповідно.

Для обох датасетів першочергово відбулася їх обробка: досліджено властивості та специфіку кожної змінної, оброблено пропущені значення, видалено нерелевантні записи та

трансформовано дані у зручний для розуміння формат майбутніми моделями. Також за допомогою статистичних методів виділено лише важливі для прогнозування ознаки.

Наступним етапом була побудова моделей машинного навчання із застосуванням згаданих раніше методів. Для кожної моделі виконано пошук по сітці Grid search і визначено параметри, при яких оцінки моделі є найвищими. У таблицях 1–2 наведено результати обчислення різник метрик на навчальній та перевірочній вибірках для моделей, що прогнозують результатів гри в жіночому тенісі, а у таблицях 3–5 – для прогнозування результатів в чоловічому тенісі.

Таблиця 1. Оцінка якості моделі логістичної регресії (жінки)

Вибірка	Назва метрики					
	Accuracy	Precision	Recall	F1 Score	Roc Auc	Loss
Навчальна	0.68966	0.68897	0.70124	0.69496	0.76102	0.57987
Перевірочна	0.68341	0.68355	0.68202	0.68277	0.75474	0.58704

Таблиця 2. Оцінка якості моделі багатошарового персептрону (жінки)

Вибірка	Назва метрики					
	Accuracy	Precision	Recall	F1 Score	Roc Auc	Loss
Навчальна	0.69175	0.68903	0.70954	0.69894	0.7599	0.5803
Перевірочна	0.68290	0.68239	0.68323	0.68274	0.7535	0.58745

Таблиця 3. Оцінка якості моделі логістичної регресії (чоловіки)

Вибірка	Назва метрики					
	Accuracy	Precision	Recall	F1 Score	Roc Auc	Loss
Навчальна	0.98278	0.98292	0.98253	0.98272	0.99790	0.05599
Перевірочна	0.98145	0.98146	0.98144	0.98145	0.99789	0.05618

Таблиця 4. Оцінка якості моделі випадкового лісу (чоловіки)

Вибірка	Назва метрики					
	Accuracy	Precision	Recall	F1 Score	Roc Auc	Loss
Навчальна	0.98345	0.98345	0.98376	0.98381	0.99763	0.07793
Перевірочна	0.98363	0.98362	0.98366	0.98349	0.99789	0.07484

Таблиця 5. Оцінка якості моделі екстремального градієнтного підсилення (чоловіки)

Вибірка	Назва метрики					
	Accuracy	Precision	Recall	F1 Score	Roc Auc	Loss
Навчальна	0.98396	0.98501	0.98293	0.98396	0.99834	0.04733
Перевірочна	0.98359	0.98267	0.98455	0.98361	0.99842	0.04601

З огляду на представлені результати можна зауважити, що моделі для прогнозування результатів матчів жіночого розряду є прийнятними, а чоловічого прогнозують результат дуже добре, однак точність останніх дещо погіршиться через необхідність передбачати статистичні змінні гравців у матчі ще до його початку. Реалізація відповідного алгоритму, що має на меті накладання ваг на показники гравців у попередніх матчах, і була наступним кроком даного дослідження.

Було розроблено веб-інтерфейс, який дозволяє користувачу зручно вводити необхідну інформацію про гравців з клавіатури, а також у разі необхідності мати можливість її змінювати і отримувати легко інтерпретовані результати роботи побудованих моделей. Інтерфейс програми представлено на рисунку 1.

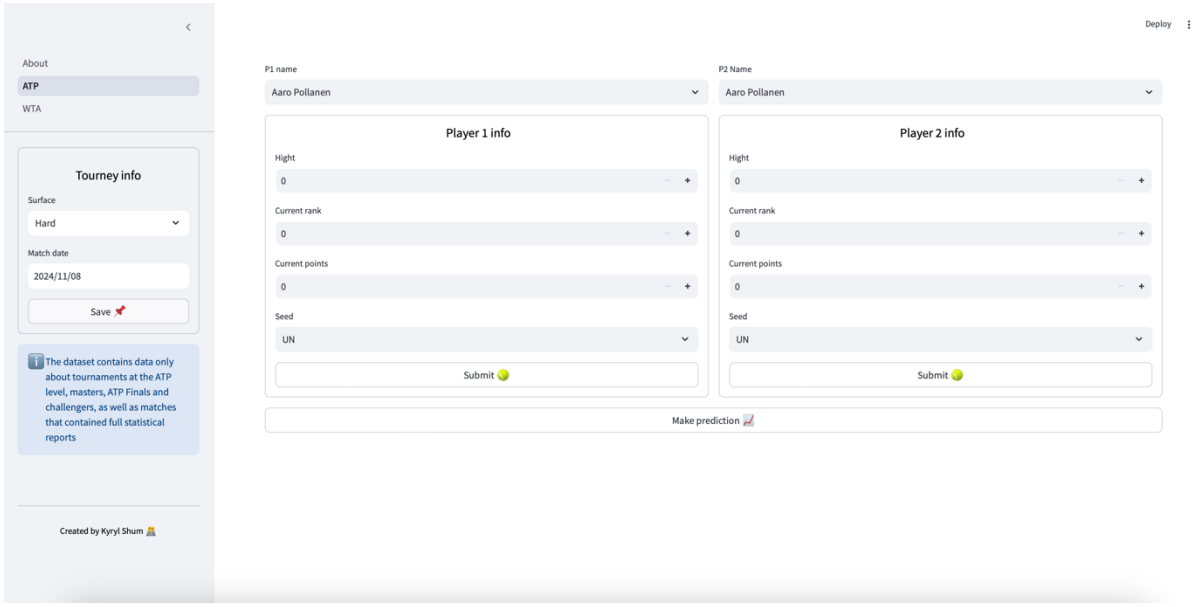


Рисунок 1. Веб-інтерфейс для прогнозування матчів

Далі для визначення потенційних стратегій ставок було сформовано нові набори даних з прогнозних матчів за допомогою побудованих раніше моделей і алгоритмів. Було сформовано вибірку зі 112 прогнозних матчів для жінок та 346 для чоловіків. Прогнозування статистичних показників для чоловіків відбувалося у двох варіаціях алгоритму: на всіх покриттях корту і тільки на обраному покритті, тобто загалом одержано 6 прогнозів на кожен матч. Вибір оптимальної стратегії робився за трьома основними показниками якості:

- прибуток, отриманий у результаті зроблених ставок. Для зручності вважалося, що сума ставок розподілена рівномірно і дорівнює 100 умовним одиницям на кожен матч;
- відсоток правильних прогнозів;
- загальний приріст до початкового банку (початковий банк дорівнює кількості ставок, помноженим на 100).

До отриманих даних було застосовано алгоритм фільтрації, який дозволяє відсікати певні категорії матчів і таким чином збільшує площу пошуку оптимальних умов прибутковості. У таблицях 6–7 представлено найбільш успішні стратегії за кожною моделлю.

Таблиця 6. Оптимальні стратегії для жіночого розряду

Модель	Поріг мінімальної ймовірності	Поріг мінімального коефіцієнту	Співвідношення прогнозів (правильні прогнози / всі прогнози)	Відсоток правильних прогнозів	Прибуток	Приріст до початкового банку (%)
Логістична регресія	0.65	1.35	14/14	100	542	39
Багатошаровий персептрон	0.65	1.35	17/18	94	588	33

Таблиця 7. Оптимальні стратегії для чоловічого розряду

Модель	Поріг мінімальної кількості зіграних матчів	Поріг мінімального коефіцієнту	Співвідношення прогнозів (правильні прогнози / всі прогнози)	Відсоток правильних прогнозів	Прибуток	Приріст до початкового банку (%)
Логістична регресія (всі покриття)	12	1.5	14/24	57	287	12
Випадковий ліс (всі покриття)	12	1.25	19/30	63	793	26
XGBoost (всі покриття)	успішні стратегії відсутні					
Логістична регресія (обране покриття)	8	1.6	12/19	63	410	22
Випадковий ліс (обране покриття)	успішні стратегії відсутні					
XGBoost (обране покрииття)	8	1.55	14/23	61	517	22

З огляду на результати можна сформувати висновок, що двома найкращими стратегіями отримання фінансової вигоди від ставок на результати тенісних матчів будуть:

- для жінок: модель логістичної регресії, оскільки її стратегія має вищий показник правильних прогнозів та більший загальний приріст;
- для чоловіків: модель випадкового лісу з алгоритмом прогнозування статистичних показників по всіх типах корту, оскільки її стратегія має найбільший загальний приріст та прибуток.

Зобразивши зміну прибутку у результаті ставок за найбільш оптимальними стратегіями отримаємо графіки, представлені на рисунках 2 та 3 для ілюстрування результативності нашої стратегії і обраної моделі для тенісних жіночих і чоловічих матчів відповідно.

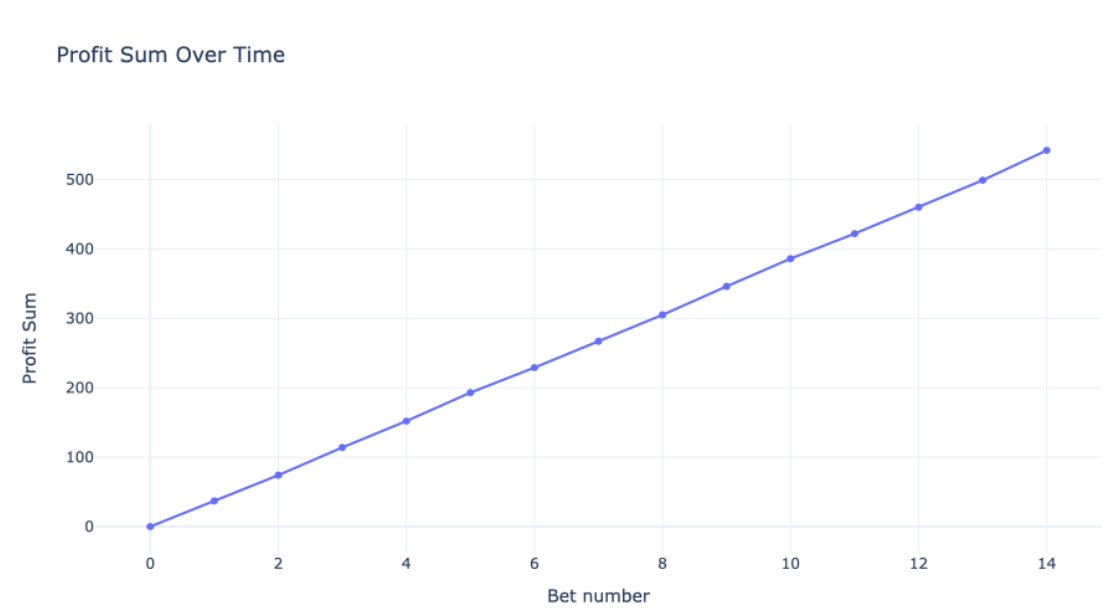


Рисунок 2. Зміна прибутку за оптимальною стратегією на основі моделі логістичної регресії для жіночих тенісних матчів

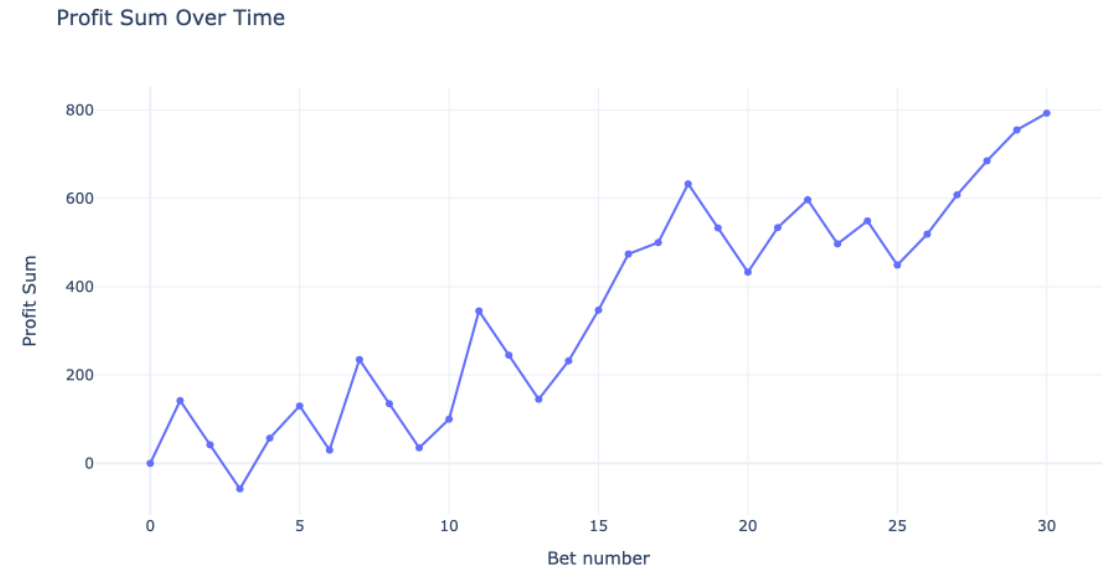


Рисунок 3. Зміна прибутку за оптимальною стратегією на основі моделі випадкового лісу з алгоритмом прогнозування статистичних показників по всіх типах корту для чоловічих тенісних матчів

5. ВИСНОВКИ

У проведеному дослідженні було розглянуто методи машинного навчання і допоміжні алгоритми прогнозування результатів тенісних матчів для визначення найкращих стратегій отримання фінансової вигоди у результаті спортивного беттингу. У процесі написання роботи було досліджено як чоловічий, так і жіночий тенісні розряди, визначено найкращі моделі для прогнозування результативності відповідних тенісних матчів.

У результаті реалізації методів машинного навчання, обробки даних і аналізу отриманих результатів було виконано пошук і визначено найефективніші стратегії ставок задля

потенційної максимізації прибутку користувача. Було розроблено програмний продукт, який дозволяє користувачу вибрати відповідний розряд матчу, переглянути параметри гравців, особливості тенісного корту і зробити ставку за результатом отриманого прогнозу.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Wikipedia, the free encyclopedia, 2004. Sports betting. [online] (Last edited 28 October 2024). Available at: <https://en.wikipedia.org/wiki/Sports_betting>. [Accessed 11 November 2024].
2. AWS. What is Logistic Regression?. [online] Available at: <https://aws.amazon.com/what-is/logistic-regression/?nc1=h_ls>. [Accessed 11 November 2024].
3. IBM. What is random forest?. [online] Available at: <https://www.ibm.com/topics/random-forest>. [Accessed 11 November 2024].
4. Nvidia. XGBoost. [online] Available at: <https://www.nvidia.com/en-us/glossary/xgboost/>. [Accessed 11 November 2024].
5. Carolina Bento 2021, Multilayer Perceptron Explained with a Real-Life Example and Python Code: Sentiment Analysis. [online] Available at: <https://towardsdatascience.com/multilayer-perceptron-explained-with-a-real-life-example-and-python-code-sentiment-analysis-cb408ee93141>. [Accessed 11 November 2024].
6. Sipko, M. 2015. Machine Learning for the Prediction of Professional Tennis Matches, Imperial College London, viewed 16 September 2024.
7. JeffSackmann. tennis_atp. [online] Available at: <https://github.com/JeffSackmann/tennis_atp>. [Accessed 18 September 2024].
8. Tennis-data. Data Files: All Competitions [online] (Last edited 9 November 2024). Available at: <http://www.tennis-data.co.uk/alldata.php>. [Accessed 4 October 2024].

ENHANCING SECURITY AND SCALABILITY IN CRITICAL INFRASTRUCTURE USING AWS CLOUD SERVICES

Lysov B.S.[1], Huskova V.H.[2]

Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine

[1] bogukraine@gmail.com, [2] guskovavera2009@gmail.com [0000-0001-7637-201X]

The article examines the potential of Amazon Web Services (AWS) for building secure and scalable infrastructure aimed at protecting critical sectors such as energy, healthcare, transportation, and finance. The proposed deployment model incorporates AWS tools, including IAM, Amazon GuardDuty, AWS Shield, Auto Scaling, and CloudWatch, to enable effective access management, real-time threat detection, and performance optimization. The analysis demonstrates the proposed approach's capability to address challenges related to cybersecurity, adaptability to fluctuating workloads, and high availability. The study's findings confirm the feasibility of leveraging AWS to establish threat-resilient infrastructure with the potential for further integration of machine learning models to enhance threat detection and response efficiency.

Keywords: Critical Infrastructure, AWS, Security Solutions, Cloud Infrastructure, Real-time Monitoring.

1. INTRODUCTION

With the increasing reliance on digital systems, the protection of critical infrastructure — spanning sectors such as energy, healthcare, transportation, and finance — has become paramount. As threats to these infrastructures grow, both in frequency and sophistication, there is a pressing need for scalable, resilient, and secure solutions that can handle and preempt unauthorized access, ensure seamless performance, and facilitate swift scalability. This need has driven the exploration of cloud services, particularly Amazon Web Services (AWS), for secure deployments of critical infrastructure. AWS offers robust security mechanisms, access control, and scalability solutions ideal for protecting critical infrastructure from malicious actors. By leveraging the AWS cloud, infrastructure systems can adapt to demand, maintain high performance under stress, and enforce strict cybersecurity protocols to prevent breaches.

1.1. Current State

Traditional on-premises solutions often lack the scalability and rapid adaptability required in modern, dynamic threat landscapes. Consequently, there is a need for a cloud-based approach that can dynamically adjust to varying loads, respond to access threats in real-time, and ensure high availability. AWS capabilities such as its security features, elastic scaling, and performance optimization tools can be applied to protect and support critical infrastructure services.

1.2. Critical infrastructure architecture issues

Critical infrastructure, which supports the core functioning of essential sectors such as energy, healthcare, transportation, and finance, often faces a variety of challenges that can significantly impact its efficiency and security. These problems may arise at different times and under different circumstances, and if not addressed properly, they can lead to severe consequences for security, operational effectiveness, and financial stability.

Scalability: Scalability issues often arise during sudden changes in load, such as increased traffic or unpredictable spikes in resource demand. In such cases, traditional on-premises solutions may not respond quickly enough, leading to reduced performance and system failures. This not only results in business losses but also impacts consumers and clients, as critical services may become unavailable.

Access Control and Access Management: Improper access management or inadequate control can lead to unauthorized access to sensitive data or critical systems. This is especially relevant for critical infrastructure, which deals with large volumes of sensitive data. Lack of adequate protection can result in cyberattacks, data breaches, financial losses, and severe reputational damage.

Cyberattack Protection: Critical infrastructure is an attractive target for cybercriminals, particularly for DDoS attacks that can lead to temporary shutdowns or decreased performance. Protection issues arise when outdated or inadequate security systems fail to respond to emerging threats. This jeopardizes data security and can lead to substantial financial losses and a decline in trust from clients and partners.

Reliability and High Availability: Systems that fail to provide high availability and fault tolerance are highly vulnerable. Failures in critical infrastructure can have catastrophic effects, including not just technical problems but also significant financial losses, operational disruptions, and damage to reputation. Availability issues typically occur due to adverse weather conditions, technical failures, or cyberattacks, underscoring the need for reliable and adaptive solutions.

Monitoring and Performance Optimization: The absence of effective monitoring can delay problem detection, negatively affecting performance and potentially increasing the cost of resolving issues. Without a proper monitoring system, it becomes difficult to adjust resources in a timely manner or to respond effectively to changing loads, which may lead to a decline in service quality.

Cost Optimization: Cost optimization issues arise when infrastructure operates with excessive resources or, conversely, insufficient resources to maintain optimal service levels. This can occur due to unforeseen load changes or inefficient resource management. Leaving unused resources operational is costly, and the lack of proper optimization can lead to overspending and unnecessary costs.

Integration and Flexibility: Integrating new technologies and adapting to changes can be challenging in traditional infrastructures due to their limited flexibility. This can become problematic when implementing new solutions or adapting quickly to evolving requirements or threats. The inability to quickly adopt innovations can delay progress and harm competitiveness.

2. APPROACH

The primary approach of this study is to design a framework for deploying critical infrastructure on AWS, emphasizing secure access management, efficient scaling, and performance optimization. This framework will integrate AWS-specific security features such as Identity and Access Management (IAM), Amazon GuardDuty, and AWS Shield to fortify access control and threat detection. Additionally, services like AWS Auto Scaling, Elastic Load Balancing, and Amazon CloudWatch will be employed to ensure that the system can scale and respond to varying demands while maintaining robust performance metrics. Through this framework, we aim to deliver a scalable, resilient, and secure infrastructure solution suitable for modern critical applications.

The main stage is to identify the access and security needs unique to critical infrastructure. This stage involves assessing the sensitivity of data, the number of user roles, and the types of devices interacting with the system. Based on these factors, AWS Identity and Access Management (IAM) is configured to enforce strict access policies, while Amazon GuardDuty and AWS Shield provide continuous threat detection and protection against distributed denial-of-service (DDoS) attacks.

Subsequent stages focus on scaling and performance. Using AWS Auto Scaling, the infrastructure can dynamically adjust resources based on demand, ensuring both efficient resource utilization and high availability. Elastic Load Balancing is employed to distribute incoming traffic across multiple instances, preventing bottlenecks and enabling seamless performance during peak usage. Amazon CloudWatch monitors system performance, allowing for real-time adjustments to maintain optimal operation.

Key Provisions. The following provisions outline the core components of the proposed framework:

1. Access Control and Threat Detection: The framework utilizes AWS IAM to establish strict access controls, defining user roles and permissions. Amazon GuardDuty continuously monitors for threats, while AWS Shield provides DDoS protection.

2. Elastic Scaling: AWS Auto Scaling is implemented to handle fluctuating workloads. This ensures the infrastructure can meet increased demand without sacrificing performance or incurring unnecessary costs.

3. Load Balancing and Performance Monitoring: Elastic Load Balancing distributes incoming requests, ensuring smooth system performance. Amazon CloudWatch enables real-time performance tracking, allowing for proactive adjustments to avoid potential issues.

4. Diagram for Process Flow: Figure 1 illustrates the process flow of deploying critical infrastructure on AWS, highlighting access control, scaling, and monitoring mechanisms.

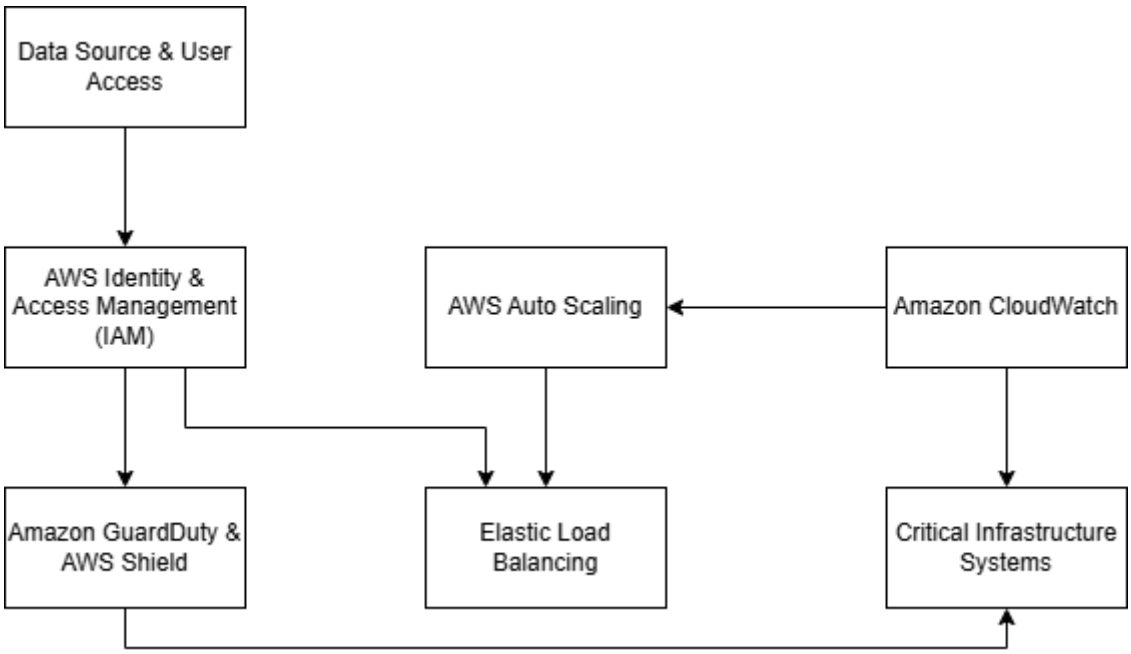


Figure 1. AWS-based deployment framework for critical infrastructure

This diagram (Figure 1) represents the AWS-based deployment framework for critical infrastructure, focusing on the interplay between access management, scaling, and performance monitoring.

Implementation and Evaluation. To validate the effectiveness of the proposed framework, the implementation will be tested under various simulated conditions to assess its response to different threat scenarios, scaling requirements, and performance benchmarks. AWS GuardDuty's threat detection accuracy will be evaluated against baseline security metrics, while Auto Scaling and CloudWatch's efficiency in handling peak loads will be monitored. Key performance indicators,

including Mean Time to Recovery (MTTR), latency, and throughput, will be measured to determine the framework's overall resilience and operational efficiency.

3. RESULTS

Results and Discussion. Initial tests suggest that the integration of AWS's security and performance tools offers a robust solution for critical infrastructure protection. GuardDuty and Shield significantly reduce unauthorized access risks and detect anomalies in real time. Auto Scaling and Elastic Load Balancing enhance system availability and performance under stress, ensuring seamless service even during high-demand periods. AWS CloudWatch provides a granular view of system health, enabling proactive issue resolution and optimization.

This study presents a practical framework for deploying critical infrastructure systems on AWS, emphasizing access security, scalability, and high performance. By integrating AWS's suite of security, scaling, and monitoring tools, critical infrastructure can achieve enhanced resilience against cyber threats and unauthorized access while efficiently managing varying workloads. Future research will explore the integration of machine learning models to further improve threat detection and response capabilities within this framework. The findings underscore AWS's potential as a reliable platform for safeguarding essential services, contributing to a fortified infrastructure capable of withstanding modern cyber threats.

REFERENCES

1. Alter S. L. Decision support systems : current practice and continuing challenges. / Alter S. L. - Reading, Mass.: Addison-Wesley Pub., 1980.
2. Bonczek R.H. Foundations of Decision Support Systems / Bonczek R.H., Holsapple C., Whinston A.B. - New York: Academic Press, 1981.
3. Cooper G.F. The computational complexity of probabilistic inference using Bayesian belief networks / Cooper G.F. // Artificial Intelligence, 1990, vol. 42, No. 2-3, p. 393-405.

РЕАЛІЗАЦІЯ МОНІТОРИНГУ МІКРОСЕРВІСНИХ СИСТЕМ

Ахмедов М.Ш.[1], Булах Б.В.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] akhmedovmagomed2001@gmail.com, [2] bogdan.bulakh@gmail.com

Стаття зосереджена на розгляді та впровадженні існуючих інструментів для моніторингу мікросервісних систем. Впродовж розгляду таких інструментів моніторингу як Zabbix, Grafan та Graphite будуть визначені ключові переваги та недоліки цих інструментів. Розглядаються практичні аспекти застосування функціоналу наведених інструментів. Функціонал включає у себе збирання різноманітних метрик операційної системи та окремих додатків, візуалізація зібраних метрик, сповіщення адміністраторів системи про несправності. У роботі було використано теоретичні та емпіричні методи дослідження.

Ключові слова: моніторинг, мікросервісна архітектура, Zabbix, Grafana, Graphite.

1. ВСТУП

У поточній ситуації розробки програмного забезпечення мікросервісна архітектура стала домінуючою парадигмою, що дозволяє створювати масштабовані, стійкі та підтримувані додатки. Цей архітектурний стиль розкладає складні системи на набір слабко пов'язаних між собою сервісів, кожен з яких відповідає за певну сферу бізнесу. Хоча мікросервіси пропонують численні переваги, вони також створюють нові виклики, особливо у сфері моніторингу.

Розбиття відповідальності між окремими сервісами, динамічний характер інфраструктури і потенціал складних взаємозалежностей роблять необхідним прийняття спеціалізованих стратегій моніторингу, які можуть ефективно відстежувати стан, продуктивність і поведінку мікросервісів.

Інструменти, які будуть розглянуті у статті включають Zabbix, Graphite та Grafana. Наведені інструменти надають такі можливості, як: централізація, підтримка серверів різних типів операційних систем, моніторинг SNMP і IPMI-пристроїв, моніторинг Java додатків, сповіщення на основі умови, графічне відображення отриманих метрик та інші.

Метою цієї статті є вивчення та оцінка існуючих рішень для моніторингу, які добре підходять для мікросервісних архітектур. Вивчаючи різні інструменти та фреймворки, потрібно визначити найбільш перспективні підходи для збору, аналізу та візуалізації критичних показників продуктивності, індикаторів стану системи та окремих додатків. Стаття починається з обґрунтування основних тверджень що відносяться до моніторингу та обґрунтовують його важливість. Далі буде представлено огляд існуючих рішень для моніторингу, висвітлено їхні переваги та основний функціонал. Також будуть висвітлені практичні аспекти використання функціоналу існуючих інструментів. У висновку будуть обговорені перспективи застосування наведених інструментів та оцінено їх вплив.

2. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

2.1. Обґрунтування основних тверджень та їх порівняння

У рамках статті будуть фігурувати наступні твердження: моніторинг (monitoring) та спостережуваність (observability).

Моніторинг – це безперервний процес збору та аналізу системних даних для виявлення, попередження та реагування на заздалегідь визначені проблеми або аномалії. [1] Зазвичай він зосереджується на відомих умовах, які є критично важливими для продуктивності системи, таких як завантаження процесора, використання пам'яті та час відгуку. Моніторинг спирається на заздалегідь визначені метрики та порогові значення для сповіщення операційних команд, коли система відхиляється від очікуваної поведінки, що дозволяє швидко реагувати для забезпечення доступності та надійності системи. Моніторинг дає змогу відстежувати історичні показники, виявляти тенденції та управляти інцидентами, часто слугуючи основою для дотримання нормативних вимог та аудиту в системах з жорсткими стандартами продуктивності.

Спостережуваність – це властивість системи, яка дозволяє зрозуміти її внутрішній стан на основі зовнішніх даних, що дає змогу виявити першопричину складних проблем. Походячи з теорії управління, спостережуваність у програмному забезпеченні та системній інженерії означає здатність робити висновки про внутрішнє функціонування системи за допомогою таких даних, як логи та метрики. Ключовими елементами спостережуваності є метрики (кількісні дані в часі) та логи (записи дискретних подій з позначкою часу) Інструменти спостережуваності об'єднують і аналізують ці точки даних, щоб забезпечити реальну інформацію в режимі реального часу, що дозволяє інженерам діагностувати проблеми без необхідності використання заздалегідь визначених сценаріїв помилок. [1]

Хоча і спостереження, і моніторинг передбачають збір даних для розуміння системи, вони суттєво відрізняються за обсягом, фокусом і цілями. [2] Моніторинг - це цілеспрямований, часто статичний процес, який фокусується на виявленні відомих проблем за допомогою певних метрик і заздалегідь визначених порогових значень. Він дозволяє оперативним командам швидко виявляти і вирішувати передбачувані проблеми, такі як вузькі місця в роботі або перенасиченість ресурсами. Однак моніторинг обмежується відомими умовами, що робить його недостатнім для діагностики нових або складних проблем без попередньої конфігурації.

З іншого боку, спостережуваність - це ширша концепція, яка прагне зрозуміти і дослідити стан системи, навіть коли проблеми невідомі або невизначені. У той час як моніторинг часто є реактивним, спостережуваність за своєю суттю є проактивною і діагностичною, використовуючи комбінацію метрик та логів для забезпечення цілісного уявлення про поведінку системи. [3] Це дозволяє інженерам вирішувати не лише рутинні інциденти, але й складні та непередбачувані проблеми, що робить спостережливість особливо цінною в динамічних, розподілених системах, таких як мікросервіси. [2] Таким чином, моніторинг має важливе значення для підтримки стабільної роботи, тоді як спостережуваність є критично важливою для діагностики та розуміння складної поведінки в сучасних архітектурах.

2.2. Огляд існуючих рішень у сфері моніторингу

На даний момент, існує безліч інструментів для реалізації концепцій спостережуваності та моніторингу. Розглянемо наступні Zabbix, Graphite та Grafana.

2.2.1. Zabbix

Zabbix – це інструмент моніторингу з відкритим вихідним кодом, призначений для відстеження, аналізу та візуалізації продуктивності та доступності IT-інфраструктури,

включаючи сервери, віртуальні машини, програми та мережеві пристрої. Він пропонує повний набір функцій для моніторингу системних показників, включаючи використання процесора, пам'яті, пропускної здатності мережі та продуктивності додатків, а також потужну систему оповіщення для сповіщення користувачів про критичні проблеми. Zabbix використовує гнучку, засновану на агентах і без агентів архітектуру, що робить його придатним для моніторингу як великомасштабних, так і гетерогенних середовищ. Інструмент також підтримує збір даних через різні протоколи, такі як SNMP, ICMP і JMX, і може бути інтегрований з API для створення власних конфігурацій моніторингу [4].

Основні переваги:

- Широкі можливості налаштування та гнучкість. Zabbix дозволяє налаштовувати конфігурації моніторингу та підтримує широкий спектр методів збору даних. Він включає в себе великий набір шаблонів, макросів і тригерів, які спрощують процес адаптації інструменту до конкретних умов.
- Детальна система оповіщення та сповіщення. Zabbix має надійну систему оповіщення, яка підтримує широкий спектр каналів оповіщення, таких як електронна пошта, SMS та інтеграція зі сторонніми програмами обміну повідомленнями. Сповіщення можуть бути налаштовані з точними умовами, щоб зменшити втому від сповіщень і визначити пріоритетність критичних подій.
- Візуалізація даних та звітність. Zabbix надає різноманітні варіанти візуалізації даних, включаючи графіки, карти та інформаційні панелі, щоб допомогти користувачам аналізувати тенденції продуктивності. Ці інструменти покращують спостережливість, дозволяючи чітко бачити історичні дані та дані в реальному часі.

2.2.2. Graphite

Graphite – це інструмент з відкритим вихідним кодом для зберігання, пошуку та візуалізації часових рядів даних у реальному часі. В основному використовується для моніторингу системних показників, Graphite збирає числові дані часових рядів з різних джерел і надає потужну платформу для візуалізації тенденцій та аналізу поведінки системи. За своєю суттю Graphite складається з трьох основних компонентів: Carbon, який отримує та зберігає вхідні дані; Whisper, база даних часових рядів для ефективного зберігання даних; та Graphite Web, користувацький інтерфейс для запитів та візуалізації збережених метрик. Завдяки своїй легкій та гнучкій архітектурі Graphite широко використовується для моніторингу інфраструктури та додатків, дозволяючи користувачам відстежувати широкий спектр показників продуктивності, від завантаження системи та використання пам'яті до специфічних показників для конкретних додатків. [5]

Основні переваги:

- Ефективне зберігання часових рядів. База даних Whisper від Graphite оптимізована для зберігання часових рядів, що робить її придатною для середовищ з великими обсягами метрик. Система зберігання використовує формат бази даних фіксованого розміру, що забезпечує передбачуване використання диска та ефективний пошук даних.
- Гнучка та масштабована архітектура. Модульна конструкція Graphite дозволяє користувачам самостійно масштабувати її компоненти відповідно до зростаючих потреб моніторингу. Ця гнучкість дозволяє змінювати частоту збору даних і глибину зберігання без шкоди для продуктивності.
- Підтримка спільноти та розширюваність. Як інструмент з відкритим вихідним кодом та активною спільнотою, Graphite має велику документацію, плагіни та інструменти сторонніх розробників. Ця мережа підтримки спрощує усунення несправностей, розширює функціональність і дозволяє налаштовувати програму під конкретні потреби.

2.2.3. Grafana

Grafana – це популярна платформа з відкритим вихідним кодом для візуалізації даних, моніторингу та оповіщення, пристосована для роботи з різними джерелами даних, включаючи Prometheus, Graphite, Elasticsearch, InfluxDB, Zabbix та інші. Grafana надає інтуїтивно зрозумілий веб-інтерфейс, який дозволяє користувачам створювати, керувати та ділитися настроюваними дашбордами. Його універсальність у поєднанні з потужними функціями оповіщення та розширеними можливостями візуалізації робить його ідеальним для моніторингу цілого ряду джерел даних у режимі реального часу, від метрик інфраструктури до продуктивності додатків та бізнес KPI. Розширюваний дизайн Grafana дозволяє користувачам додавати плагіни, створювати власні панелі та інтегруватися з різними системами, що робить його одним з найбільш поширених інструментів для спостереження та моніторингу. [6]

Основні переваги:

- Сумісність даних з різних джерел. Grafana підтримує широкий спектр джерел даних, дозволяючи користувачам агрегувати та візуалізувати показники з різних систем на одній інформаційній панелі. Ця гнучкість забезпечує крос-платформну аналітику та спрощує моніторинг в середовищах з гетерогенним зберіганням даних.
- Насичені та настроювані візуалізації. Grafana надає різноманітні варіанти візуалізації, включаючи графіки, теплові карти, таблиці та геомапи, що дозволяє детально аналізувати патерни даних. Панелі інформаційної панелі, що налаштовуються, дозволяють користувачам пристосовувати візуалізації до конкретних потреб, підвищуючи зручність використання для різних ролей і команд.
- Розширені можливості оповіщення. Система сповіщень Grafana підтримує складні сповіщення на основі умов з гнучкими каналами сповіщення (наприклад, електронна пошта, Slack, PagerDuty). Сповіщення можуть бути налаштовані з пороговими значеннями, функціями агрегації та інтервалами оцінки, пропонуючи проактивний моніторинг та своєчасне реагування на інциденти.

3. ВИКОРИСТАННЯ ФУНКЦІОНАЛУ ІСНУЮЧИХ ІНСТРУМЕНТІВ

Для демонстрації функціоналу реалізуємо моніторинг черг повідомлень. Моніторинг черг повідомлень, таких як RabbitMQ або Kafka, має вирішальне значення для забезпечення надійності системи, пропускної здатності повідомлень та затримок у розподілених системах. Використання Zabbix та Grafana разом забезпечує як комплексний збір даних, так і розширену візуалізацію для моніторингу та аналізу продуктивності черги повідомлень.

Для моніторингу черг повідомлень за допомогою Zabbix, як правило, потрібно встановити і налаштувати Zabbix Agent на серверах, де працює служба черг повідомлень. Для ілюстрації ми будемо використовувати RabbitMQ, хоча подібні принципи застосовуються і до інших систем керування чергами:

1. Налаштуйте Zabbix Agent для метрик черг:

- На сервері RabbitMQ встановіть Zabbix Agent і налаштуйте плагіни RabbitMQ для передачі метрик через HTTP.
- Плагін rabbitmq_management можна використовувати для надання метрик RabbitMQ (таких як розмір черги, швидкість передачі повідомлень, використання користувачами) через HTTP API.
- Zabbix може використовувати тип елемента агента HTTP для збору цих показників. Крім того, метрики RabbitMQ можуть бути представлені у форматі JSON і оброблені Zabbix для створення відповідних елементів.

2. Створення елементів Zabbix для метрик черги:

- В інтерфейсі Zabbix створіть елементи для моніторингу ключових метрик RabbitMQ, таких як:
  - Розмір черги (кількість повідомлень у кожній черзі).
  - Опубліковані та спожиті повідомлення за секунду.
  - Кількість і статус споживачів.
  - Швидкість повідомлень (вхідних та вихідних) та час обробки повідомлень.
- Налаштуйте тригери Zabbix, щоб сповіщати, коли ці метрики перевищують задані пороги, наприклад, глибина черги вище певного рівня, що вказує на потенційні відставання.

3. Оповіщення та сповіщення:

- Налаштуйте тригери та порогові значення для кожної метрики, яку ви відстежуєте. Наприклад, якщо швидкість обробки повідомлень значно падає або глибина черги перевищує певний поріг, Zabbix може надсилати сповіщення електронною поштою, SMS або через інші інтегровані сервіси.
- Налаштуйте ескалацію для критичних проблем, які не вирішуються протягом тривалого часу, щоб забезпечити швидке вирішення.

Використовуючи Grafana, зібрані метрики Zabbix можна візуалізувати за допомогою детальних дашбордів, щоб забезпечити розуміння в реальному часі та аналіз історичних даних.

4. Інтегруйте Zabbix з Grafana:

- Встановіть плагін Grafana Zabbix, який дозволяє Grafana отримувати дані безпосередньо з Zabbix.
- Налаштуйте плагін на підключення до сервера Zabbix за допомогою Zabbix API, що дозволить Grafana отримати доступ до метрик черги, зібраних Zabbix.

5. Створіть інформаційну панель черги повідомлень (рис. 1):

- У Grafana створіть новий дашборд з панелями для ключових метрик:
- Глибина черги з часом: Використовуйте лінійні або гістограми, щоб показати глибину кожної черги, виділяючи періоди високого навантаження або відставання.
- Швидкість обробки повідомлень: Налаштуйте панелі часових рядів для відображення кількості опублікованих і спожитих повідомлень за секунду, що дозволить командам контролювати пропускну здатність в режимі реального часу.
- Статус споживача: Використовуйте панелі стану або теплові карти для відображення стану споживачів у черзі (активні/неактивні) та стану їхнього з'єднання.
- Затримка і час обробки: відстежуйте затримку повідомлення від моменту його розміщення в черзі до споживання, допомагаючи виявити вузькі місця в обробці повідомлень.

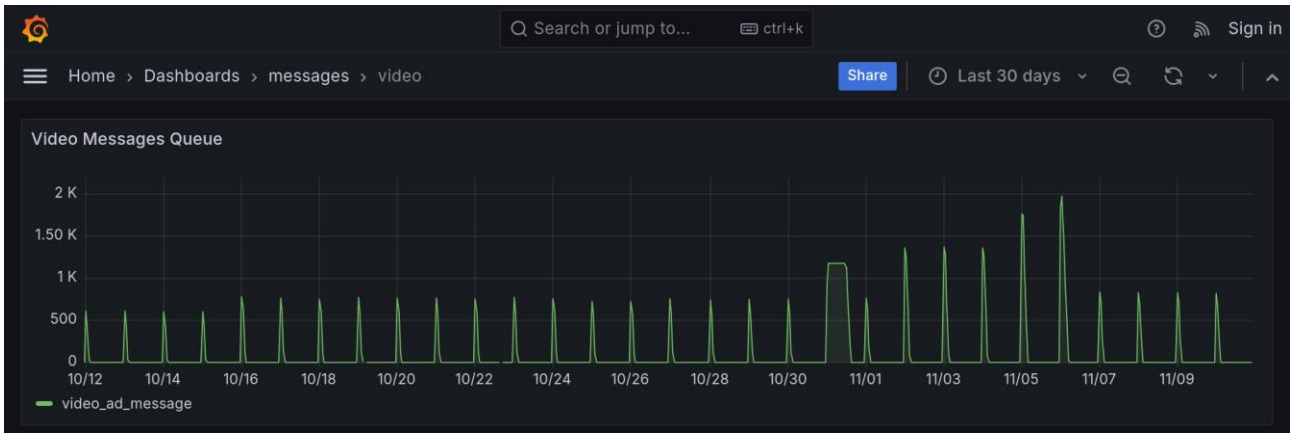


Рисунок 1. Візуалізація динаміки повідомлень черги video_messages_queue у середовищі Grafana

4. ВИСНОВКИ

Реалізація моніторингу в мікросервісних архітектурах була розглянута через аналіз існуючих інструментів і методологій. Висвітлено проблеми, пов'язані з моніторингом розподілених, контейнерних середовищ, підкреслено потребу в рішеннях, здатних обробляти динамічні робочі навантаження і залежності сервісів. Було проведено порівняння та оцінку різних інструментів, кожен з яких продемонстрував унікальні сильні сторони та обмеження при застосуванні до екосистем мікросервісів.

У статті показало, що Zabbix, Graphite і Grafana мають відмінні риси, які відповідають конкретним потребам моніторингу. Zabbix, з його широкими можливостями оповіщення та методами збору даних, виявився дуже придатним для середовищ, що потребують детального моніторингу та оповіщення інфраструктури, хоча йому може не вистачати прямої підтримки контейнерних мікросервісів без додаткового налаштування. Ефективність Graphite у зберіганні часових рядів даних та його масштабованість для високочастотних даних роблять його цінним для ресурсномістких установок, хоча його обмеження в нативних оповіщеннях та аналітиці вимагають додаткових інструментів для досягнення всебічної спостережливості. З іншого боку, Grafana довела свою ефективність завдяки розширеним можливостям візуалізації та підтримці інтеграції, що дозволяє їй слугувати централізованою інформаційною панеллю, яка розширює уявлення про різні джерела даних, хоча і покладається на зовнішні джерела для збору даних.

Функціональність кожного інструменту була оцінена, продемонструвавши, що ефективний моніторинг у мікросервісах часто вимагає комбінації інструментів. Така інтеграція може запропонувати цілісне уявлення про стан системи, коли кожен інструмент доповнює обмеження інших. Потреба в таких комбінованих системах моніторингу ілюструє складність спостереження за мікросервісами, де компоненти системи часто розосереджені по декількох вузлах, контейнерах і сервісах, створюючи великий обсяг телеметричних даних, які необхідно ефективно збирати, зберігати і аналізувати.

Загалом, було визначено, що індивідуальний підхід до моніторингу, який використовує можливості різних інструментів, є важливим для вирішення унікальних проблем мікросервісних архітектур. Рекомендується, щоб майбутні дослідження і розробки були зосереджені на вдосконаленні функцій інтеграції та масштабування інструментів, а також на вивченні аналітики, керованої штучним інтелектом, для оптимізації часу відгуку і подальшого підвищення надійності системи.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Observability Engineering: Achieving Production Excellence / Charity Majors, Liz Fong-Jones, George Miranda / O'Reilly Media 2022.

2. Distributed Systems Observability / Cindy Sridharan / O'Reilly Media 2018.

3. Practical Monitoring: Effective Strategies for the Real World 1st Edition / Mike Julian / O'Reilly Media 2017.

4. Zabbix Solution in IT & Telecommunications. Офіційний сайт Zabbix: веб-сайт. URL: https://www.zabbix.com/it_and_telecommunications (дата звернення: 10.11.2024).

5. Overview — Graphite 1.2.0 documentation. Документація Graphite: веб-сайт. URL: https://graphite.readthedocs.io/en/latest/overview.html (дата звернення: 10.11.2024).

6. Grafana OSS and Enterprise | Grafana documentation. Офіційний сайт Grafana: веб-сайт. URL: https://grafana.com/docs/grafana/latest (дата звернення: 10.11.2024).

ОГЛЯД АРХІТЕКТУРИ НЕЙРОМОРФНИХ ЧИПІВ: БАЗОВІ ПРИНЦИПИ ФУНКЦІОНУВАННЯ, ЇХ ВІДМІННІСТЬ ВІД ТРАДИЦІЙНИХ CPU ТА GPU

Бабічев І.О.[1], Письменний І.О.[2]

Національний технічний університет України
«Київський політехнічний інститутімені Ігоря Сікорського», Київ, Україна

[1] babichev.ilya25@gmail.com, [2] ihor.pismennyy@gmail.com [0000-0001-7648-2593]

Мета даної роботи – дослідити архітектуру нейроморфних чипів, проаналізувати їх основні принципи функціонування, переваги та недоліки порівняно з традиційними процесорами (CPU) та графічними процесорами (GPU) у задачах штучного інтелекту, а також оцінити їх потенціал для застосування в енергоефективних автономних системах. Наукова новизна дослідження полягає у демонстрації практичної доцільності нейроморфних чипів для застосування в автономних системах, а також у висвітленні актуальних викликів, пов'язаних із розробкою нейронних мереж для них. Практична значимість полягає в можливості подальшого використання нейроморфних технологій для побудови систем зі зниженим енергоспоживанням, що є критичним для автономних застосувань.

Ключові слова: нейроморфні чипи, штучний інтелект, енергоефективність, автономні системи, Akida AKD1000, комп'ютерний зір, процесор, GPU, класифікація зображень.

1. ВСТУП

У сучасних автономних системах зростає потреба в ефективному використанні ресурсів для обробки великих обсягів даних в реальному часі, особливо у сфері штучного інтелекту (ШІ). Традиційні процесори та GPU, хоч і забезпечують високу продуктивність, часто споживають значні обсяги енергії, що стає критичним у контексті автономних пристроїв, таких як дрони, роботи та IoT. Саме тому дослідження нових підходів до обчислень, таких як нейроморфні чіпи, набувають дедалі більшого значення.

Нейроморфні чипи імітують роботу біологічного мозку, що дозволяє значно зменшити енергоспоживання та покращити ефективність при виконанні складних задач, таких як розпізнавання образів або обробка сигналів. Завдяки використанню подійно-орієнтованих архітектур, ці чіпи споживають енергію лише при передачі інформації, що робить їх особливо привабливими для енергоефективних автономних систем. Одним з таких чіпів є Akida AKD1000, який розроблено для використання в обчисленнях, де критичними є швидкість і ефективність. Він пропонує нові можливості для реалізації моделей штучного інтелекту на малопотужних пристроях, що працюють автономно.

2. НЕЙРОМОРФНІ ОБЧИСЛЮВАЛЬНІ СИСТЕМИ

2.1. Загальна схема роботи

Нейроморфні обчислювальні системи черпають своє натхнення з біології мозку, який демонструє надзвичайну ефективність та продуктивність у вирішенні складних завдань при низькому рівні енергоспоживання. Щоб зрозуміти суть нейроморфних чипів, важливо

розглянути, як функціонують біологічні нейрони, і яким чином ці механізми стали основою для розробки інноваційних апаратних рішень.

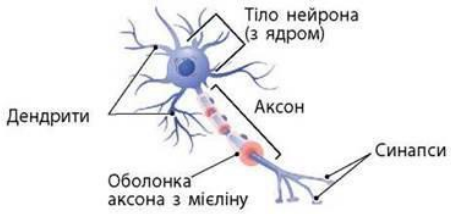


Рисунок 1. Будова біологічного нейрона [1]

У біологічних системах нейрон складається з кількох ключових елементів: дендритів, соми та аксона [2]. Дендрити приймають сигнали від інших нейронів і передають їх до соматичної частини нейрона, де ці сигнали обробляються. Коли сигнал досягає певного порогу, нейрон активується і передає імпульс вздовж аксона до інших нейронів через синапси. Ця подієво-орієнтована передача сигналів є фундаментальним принципом біологічної нейронної мережі, що дозволяє зберігати та передавати інформацію лише тоді, коли це необхідно, мінімізуючи енергетичні витрати.

Нейроморфні чипи використовують цей принцип у своїй архітектурі. Вони створюють штучні нейрони та синапси, які імітують роботу біологічних нейронів. Основною перевагою таких чіпів є їх подієво-орієнтована обробка даних: сигнали (спайки) передаються лише тоді, коли досягається певний поріг активації, що знижує енергоспоживання в порівнянні з традиційними підходами, де всі дані обробляються одночасно незалежно від їх значущості [3].

Таким чином, нейроморфні чипи реалізують спайкові нейронні мережі (SNN), де в будь-який момент часу один або кілька нейронів можуть відправляти імпульси (спайки) своїм сусідам через направлені зв'язки, які діють як синапси. Кожен нейрон має свій локальний стан і набір правил, які визначають час генерації імпульсу. Взаємодія між нейронами в мережі є повністю асинхронною, тобто нейрони передають інформацію лише тоді, коли їх локальні умови досягають певного порогу [2]. Це суттєво відрізняється від традиційних нейронних мереж, де інформація обробляється на кожному кроці для всіх нейронів одночасно.

Нейроморфні чипи реалізують архітектуру, що базується на мережі з багатьох ядер, організованих у вигляді сітки, а також позачіповий інтерфейс, що дозволяє масштабувати систему в чотирьох напрямках як показано на рисунку 2. Ця архітектура дозволяє об'єднувати кілька чипів у більш складні структури для вирішення великих задач, наприклад, обробки в реальному часі у великих автономних системах [2].

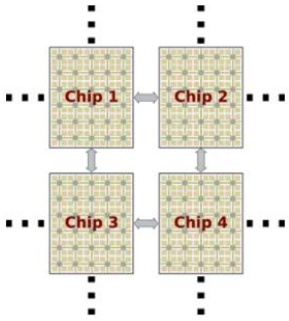


Рисунок 2. Принцип масштабування системи на основі нейроморфних чіпів [2]

Архітектура нейроморфних чипів зазвичай базується на синаптичних вагах, що визначають силу зв’язків між штучними нейронами, і на здатності цих чипів ефективно адаптуватися до нових умов завдяки пластичності — механізму, що дозволяє їм змінювати

свої параметри в залежності від вхідних даних.

Унікальною особливістю нейроморфних ядер є інтегрований модуль, який дозволяє здійснювати навчання безпосередньо на чипі за допомогою вищезгаданого механізму пластичності. Це реалізується за допомогою програмованих правил і дозволяє модифікувати синаптичні ваги нейронів у реальному часі на основі отриманих сигналів(спайків).

Нейрони в таких чипах зазвичай групуються в деревоподібні структури, щоб спростити реалізацію архітектури. Кожна група нейронів має однакові вхідні та вихідні зв'язки, конфігурації та змінні стану, що дозволяє спрощувати управління нейронними мережами та забезпечує високу ефективність роботи чипа [2].

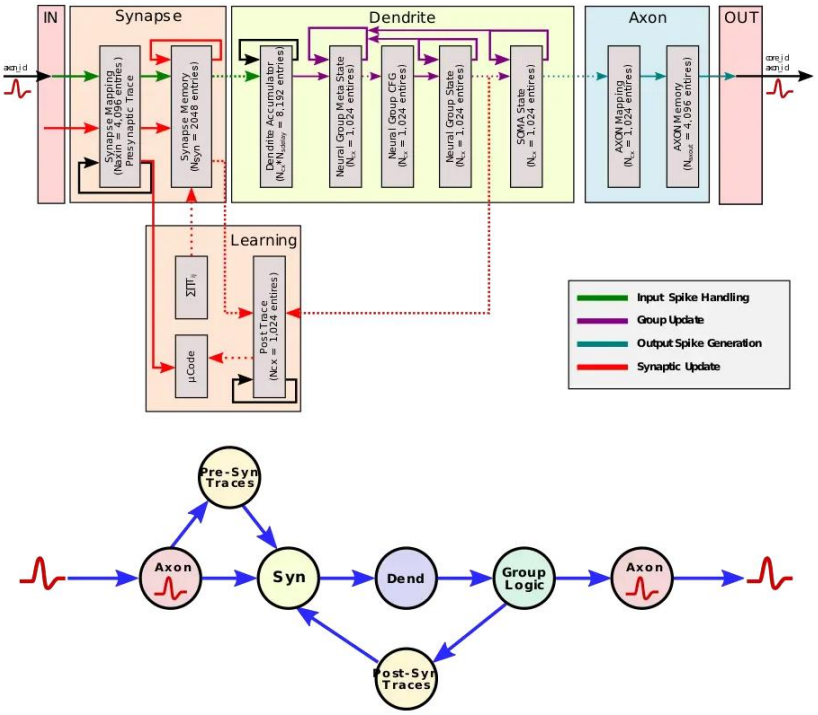


Рисунок 3. Блок-схема нейроморфного ядра [2]

Вище наведено блок-схему основних блоків пам'яті, що показує різні зв'язки, конфігурації та динамічні стани всіх нейронів, які відображаються на нейроморфне ядро. Сигнали надходять на вхід, обробляються внутрішньо (синаптичне/групове оновлення), і, за умови виконання певних умов, на виході може бути згенерований новий сигнал. Більшість цих блоків працюють незалежно з мінімальною синхронізацією керування. Дуже часто апаратне забезпечення паралелізується таким чином, що одна велика подія обробляється ефективніше, розбиваючись на кілька менших [3]. Це зображено у вигляді пунктирної лінії.

Більшість нейроморфних чипів реалізують модель нейронів "витікання, інтеграція та активація" (leaky integrate-and-fire) з двома внутрішніми змінними стану:

- Синаптичний струм — зважена сума вхідних сигналів(спайків) і постійного зміщення.
- Мембранний потенціал — витікаючий (тобто з часом слабшає) потенціал сигналу, який запускає генерацію вихідного спайку при перевищенні певного порогу активації.

Варто зазначити, що оскільки ці чипи є цифровими архітектурами, наведені вище безперервні функції апроксимуються за допомогою дискретного кроку часу, коли всі нейрони зберігають часову позначку, синхронізовану по всьому чіпу. Це необхідно для забезпечення чітко визначеної поведінки системи [3].

2.2. Переваги та недоліки нейроморфних чипів порівняно із традиційними архітектурами

З особливостей архітектури нейроморфних чипів випливає ряд переваг порівняно з традиційними архітектурами, такими як CPU, чи GPU:

- Паралельна обробка інформації: Нейроморфні чипи імітують роботу мозку, що дозволяє їм обробляти інформацію паралельно. Це значно покращує їх здатність виконувати кілька завдань одночасно, особливо ефективно підходячи для задач розпізнавання патернів та навчання, в той час як CPU виконують завдання в лінійній послідовності.
- Адаптивність: Нейроморфні чипи побудовані за принципом нейронної мережі, що дозволяє їм адаптуватися до середовища і навчатися, з часом стаючи ефективнішими у виконанні певних завдань. З іншого боку, традиційні CPU працюють за визначеними інструкціями.
- Ефективність у розпізнаванні патернів: Нейроморфні чипи спеціально розроблені для завдань, пов'язаних із навчанням. Хоча CPU і GPU теж можуть виконувати ці задачі, нейроморфні чипи роблять це з більшою ефективністю.
- Енергоефективність: Нейроморфні чипи оптимізовані для низького енергоспоживання, зберігаючи високу ефективність при виконанні складних завдань, особливо в галузі штучного інтелекту та машинного навчання. У той час як CPU і GPU споживають значно більше енергії.

З іншого боку, нейроморфні чипи також мають ряд недоліків порівняно із CPU, чи GPU:

- Обмеженість у використанні для загальних завдань: Нейроморфні чипи спеціалізовані для конкретних типів задач, таких як розпізнавання патернів або навчання, що робить їх менш ефективними для широкого спектра завдань, які виконує CPU або GPU. Традиційні процесори (CPU) є універсальними і можуть виконувати будь-які завдання, включаючи управління операційними системами та запуск додатків, що робить їх незамінними для широкого кола застосувань. GPU, спочатку розроблені для рендерингу графіки, також виявилися дуже корисними для задач, де потрібно обробляти великі обсяги даних одночасно.
- Відсутність лінійної послідовності: Через нелінійність архітектури нейроморфних чипів, вони можуть бути менш ефективними для задач, що потребують суворої лінійної послідовності обчислень, яку забезпечує архітектура CPU.
- Менша продуктивність у масових обчисленнях: GPU мають вищу продуктивність у виконанні великих обчислень і обробці масивних наборів даних, завдяки їх здатності обробляти одночасно безліч даних на великій швидкості, а тому нейроморфні чипи поступаються GPU у показниках чистої обчислювальної потужності.

2.3. Нейроморфний чип Akida AKD1000

В межах даної роботи було детально розглянуто нейроморфні чипи на прикладі Akida AKD1000 від компанії BrainChip. Akida AKD1000 призначений для забезпечення високопродуктивних можливостей штучного інтелекту на периферії (edge computing) з акцентом на енергоефективність. Дане апаратне забезпечення оптимізоване для прискорення згорткових (CNN) та глибоких (DNN) нейронних мереж, але переважно використовується саме для задач комп’ютерного зору [4]. Так, згідно з особливостями нейроморфної архітектури даний чип використовує підхід до обробки зображень на основі подій, перетворюючи кожен піксель на окрему подію за допомогою методу Rank Order Coding (ROC) [5].

Спосіб роботи чипа сам по собі досить простий: як тільки накопичується достатня кількість імпульсів, що перевищують визначений поріг, створюється повідомлення про імпульс і відправляється до різних груп на різних кінцевих ядрах. Комунікація здійснюється

таким чином, що кожне з ядер незалежно перебирає кожну зі своїх груп нейронів, і для кожної групи, яка досягає стану активації, ядро генерує повідомлення про імпульс, яке розповсюджується до всіх інших ядер у сітці. Ітерації всіх груп на всіх ядрах повинні виконуватися в межах одного і того ж дискретного часового кроку. Щоб переконатися, що всі імпульси досягли своїх цілей до повторення операції, нейроморфний чип відправляє повідомлення синхронізації, яке спочатку скидає всі імпульси, які знаходяться в дорозі, а потім відправляє всім ядрам повідомлення про перехід на наступний часовий крок t+1, що дозволяє їм продовжити оновлення своїх внутрішніх груп.

Незважаючи на численні переваги нейроморфних чипів, Akida AKD1000 має низку обмежень, що створюють виклики при розробці [6]. Зокрема, обмежує використання шарів нейронних мереж. Це означає, що більш складні архітектури, як-от сегментація, механізми уваги та скіп-з'єднання, наразі не підтримуються. При розробці кастомної моделі на основі Keras необхідно дотримуватися суворих правил щодо порядку шарів моделі, а також враховувати низку інших обмежень. Основні виклики включають наступне:

- Типи шарів нейронної мережі: лише певні шари Keras підтримуються для Akida-сумісних моделей, зокрема обробники (Conv2D, SeparableConv2D, Dense), активації (лише ReLU) та пулінгові шари (MaxPool2D, GlobalAvgPool2D, BatchNormalization, Dropout, Flatten, Input, Reshape).
- Порядок шарів: моделі повинні дотримуватися чіткого шаблону, де кожен блок шарів обов’язково починається з обробника і закінчується активацією, за наявності, BatchNormalization/GlobalAvgPool2D повинні бути розміщені перед активацією, тощо.
- Квантування: кількість бітів для активацій та ваг обмежена 4, 2 або 1-бітними значеннями, з 8-бітною точністю, що дозволяється лише для ваг у першому шарі нейронної мережі. Такі архітектурні особливості можуть потенційно призвести до значної втрати точності.

Проте незважаючи на усі виклики та недоліки архітектури, нейроморфні чипи мають значний потенціал у різних галузях завдяки своїй здатності імітувати роботу людського мозку, забезпечуючи адаптивне навчання та обробку даних у реальному часі.

3. РОЗПІЗНАВАННЯ ТИПІВ ПОВЕРХНІ НА НЕЙРОМОРФНОМУ ЧИПІ AKIDA AKD1000

3.1. Розробка Akida-сумісної нейронної мережі

Проєкт "Autonomous-Seed-Dispensing-Quadcopter" був створений для автоматизації процесу висаджування насіння на великих площах шляхом використання квадрокоптера, оснащеного нейронною мережею, здатною в реальному часі розпізнавати ділянки ґрунту, придатні для висадки. Ця модель навчається на спеціальному датасеті, який включає зображення різних типів ландшафтів, поділених на придатні та непридатні для висадки. Таким чином, завдання розпізнавання формулюється як задача класифікації зображень, що визначає найбільш сприятливі зони для висадки насіння.

Для прискорення процесу навчання використовувався підхід Transfer Learning, який дозволяє адаптувати вже попередньо навчені ваги під специфіку конкретної задачі. Було обрано стандартний розмір зображень 224x224 пікселів, який відповідає вимогам до моделей, натренованих на ImageNet. Датасет розподілено на тренувальну та валідаційну частини (80/20), а також використано такі аугментації, як RandomHorizontalFlip, RandomRotation та RandomTranslation, для імітації змін у перспективі та положенні камери квадрокоптера. Це дозволяє моделі краще адаптуватись до варіативних умов під час зйомки поверхні. Для узгодженості з навчанням на ImageNet значення пікселів було масштабовано до діапазону [0; 1].

Враховуючи обмеження нейроморфного чипа BrainChip Akida AKD1000, було розроблено оптимальну архітектуру для класифікації типів поверхні, яка складається з 14 основних будівельних блоків. Основними компонентами цих блоків є шари Conv2D, SeparableConv2D, BatchNorm2D та ReLU, з чітко визначеною послідовністю: Conv2D/SeparableConv2D -> BatchNorm2D -> ReLU. Використання SeparableConv2D на пізніх етапах мережі сприяє збереженню обчислювальної ефективності, хоча Conv2D залишається основним інструментом на початкових етапах для максимальної точності витягування ознак. Також додано кілька повнозв'язних шарів для покращення загальної точності класифікації як показано на рисунку 4. В результаті було отримано нейронну мережу, яка має 1.1 мільйон параметрів.

```
x = base_model.output
x = dense_block(x,
                units=512,
                name="fc1",
                add_batchnorm=True,
                relu_activation="ReLU7.5")
x = Dropout(0.5, name="dropout_1")(x)
x = dense_block(x,
                units=N_CLASSES,
                name="predictions",
                add_batchnorm=False,
                relu_activation=False)
x = Activation("softmax", name="act_softmax")(x)
x = Reshape((N_CLASSES,), name="reshape")(x)
```

Рисунок 4. Фінальні модифікації нейронної мережі для задачі класифікації типів поверхонь

Для даної задачі застосовується лосс-функція SparseCategoricalCrossentropy, яка добре підходить для мультикласової класифікації і мінімізує різницю між прогнозованими ймовірностями та фактичними класами. Оскільки датасет є збалансованим, то немає потреби додатково зважувати класи, що спрощує процес оптимізації. Формульно лосс-функція визначається як $Loss = -\sum_i^C y_i log\hat{y}_i$, де C – кількість класів, y_i – клас і-го зображення перетворений у бінарний вектор(вектор нулів довжини C, де одиниця встановлюється в індексі, який відповідає номеру класу), $\hat{y}_i$ – вектор ймовірностей, отриманий з нейронної мережі.

В якості оптимізаційного алгоритму обрано AdamW, який в поєднанні з експоненційним learning rate decay дозволяє швидко навчати модель на початкових етапах та плавно знижувати темп навчання для точнішого налаштування в кінці. Початковий темп навчання встановлюється на 1×10^{-3}, а кінцевий на 1×10^{-5}. Це допомагає збалансувати швидкість та стабільність навчання. Зміну темпу навчання в процесі тренування нейронної мережі проілюстровано на рисунку 5.

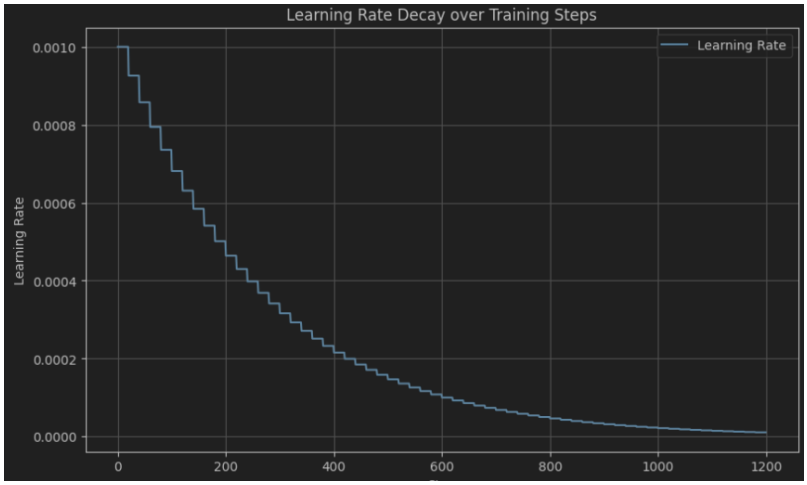


Рисунок 5. Зміна темпу навчання з 1×10^{-3} до 1×10^{-5} в процесі тренування

Після завершення етапу навчання наступним кроком є квантування моделі. Оскільки для роботи на чипі Akida AKD1000 підтримуються лише 4-, 2- та 1-бітні значення вагів і активацій (за винятком першого шару, де дозволяється 8-бітне значення вагів), застосовується схема 8/4/4. Перший шар нейронної мережі квантується у 8-бітний формат, а решта – до 4 біт для ваг та активацій. Квантування виконується за допомогою функції cnn2snn.quantize, яка замінює стандартні шари Keras на кастомні з підтримкою квантування, зокрема QuantizedReLU та QuantizedDense.

Зазвичай квантування стандартної Keras-моделі призводить до деякої втрати продуктивності. Якщо вона є надмірною, можливим рішенням є використання Quantization Aware Training (QAT) [7]. Оскільки отримана квантована модель залишається Keras-моделлю, її можна донавчати за допомогою стандартного Keras API. Таким чином, для досягнення Quantization Aware Training достатньо запустити попередньо розроблений пайплайн навчання на квантованій моделі.

Нарешті, останнім етапом є конвертація квантованої моделі в спайкову нейронну мережу, сумісну з чипом Akida AKD1000. Для цього використовується функція convert з бібліотеки CNN2SNN, яка адаптує модель для роботи з нейроморфним чипом [6]. Після цієї конвертації модель готова до запуску на апаратному забезпеченні Akida AKD1000, де вона буде використовувати нейроморфні принципи для ефективного розпізнавання типів поверхні.

3.2. Збір та аналіз цільових метрик

Завдяки послідовності розробленого пайплайну навчання нейронної мережі можна відслідковувати зміну її точності на датасеті. Результати подано у таблиці 1.

Таблиця 1. Зміна точності на кожному етапі розробки нейронної мережі

Етап	Точність, %
Keras навчання	99.17
Квантування(8/4/4)	31.67
Quantization-Aware Training(QAT)	96.66
Конвертація(SNN)	96.66

Висока точність (99,17%) свідчить про те, що модель добре навчилася розпізнавати класи в збалансованому датасеті. Це слугує базовою точкою для подальшого аналізу впливу квантування та конвертації в спайкову нейронну мережу (SNN). Статичне квантування включає зменшення розрядності ваг і активацій, що дозволяє суттєво знизити обсяг пам'яті та енергоспоживання. Однак цей етап призвів до значного падіння точності, оскільки зменшення формату ваг і активацій до 4 біт у більшості шарів призвело до втрати інформації. QAT дозволяє адаптувати модель до умов обмеженої розрядності під час навчання, змінюючи параметри мережі з урахуванням квантування. Таким чином, завдяки даному етапу практично вдалося відновити початкову точність моделі після її стискання у більш компактне представлення. Перехід до спайкової нейронної мережі не вплинув на точність, яка залишилася на рівні 96,66%. Це свідчить про успішну конвертацію з мінімальними втратами продуктивності. Як результат, хоч і кінцева спайкова нейронна мережа має дещо нижчу точність, різниця з моделлю, яка оперує дробовими числами, є незначною і складає 2,51%. З іншого боку, було отримано оптимізоване та компактніше представлення нейронної мережі сумісне з апаратним забезпеченням Akida AKD1000.

Наступним кроком в аналізі ефективності роботи нейроморфного чипа є заміри швидкості обробки зображень. Експеримент проводився з метою оцінки швидкодії спайкової нейронної мережі (SNN) на апаратному забезпеченні Akida AKD1000. Основним критерієм була кількість оброблених зображень за секунду (FPS) для різних розмірів батчу (Batch Size), які збільшувалися кратно степеню двійки (2^k, де k=0,…,7). Результати подано у вигляді графіка на рисунку 6. Така оцінка є критичною для перевірки здатності чипа забезпечувати обробку в режимі реального часу, що важливо для додатків із жорсткими часовими обмеженнями, таких як автономні системи чи обробка потокових даних.

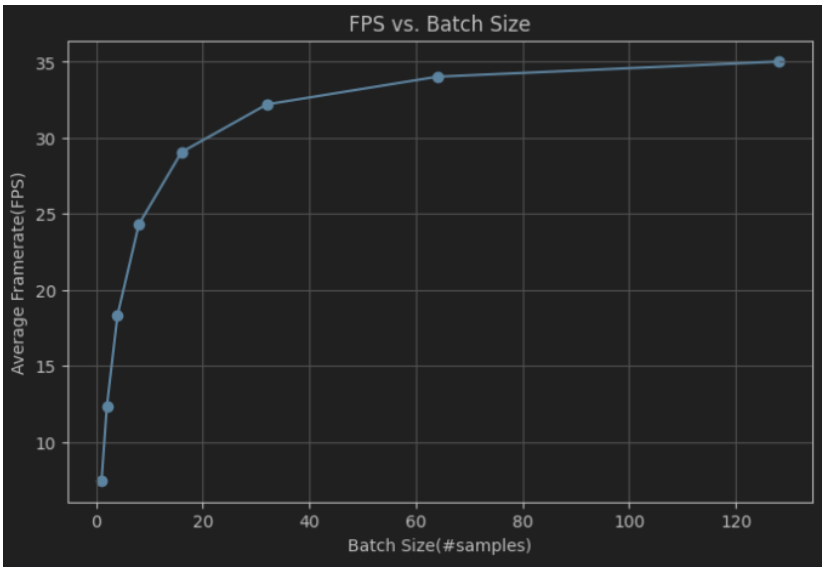


Рисунок 6. Залежність FPS від розміру батчу

В отриманих даних прослідковується явний тренд: зі збільшенням розміру батчу FPS також збільшується, але темп зростання знижується при перетині певного порогового значення Batch Size:

- Для малої кількості зображень (1–16): Спостерігається швидке зростання FPS. Це вказує на те, що чип ефективно використовує паралелізм для обробки невеликих розмірів батчу.
- Для великої кількості зображень (16–128): Зростання FPS сповільнюється, що свідчить про досягнення апаратних обмежень продуктивності.

Графік FPS від Batch Size має експоненційний характер з поступовим насиченням. На

початкових етапах продуктивність чипа різко зростає, але після певного порогу (Batch Size ≈ 16) збільшення продуктивності стає менш помітним. Це типово для систем, де апаратні ресурси використовуються максимально ефективно до певного рівня, після чого додавання нових задач не призводить до пропорційного зростання швидкодії.

Для багатьох додатків у режимі реального часу, таких як відеоаналітика або автономні системи, потрібне значення FPS становить 30. Навіть при невеликому Batch Size (16), чип досягає ~29 FPS, що вже є близьким до межі реального часу. При збільшенні Batch Size до 64 і 128, швидкість досягає відповідно 34 і 35 FPS, що забезпечує стабільну роботу в режимі реального часу.

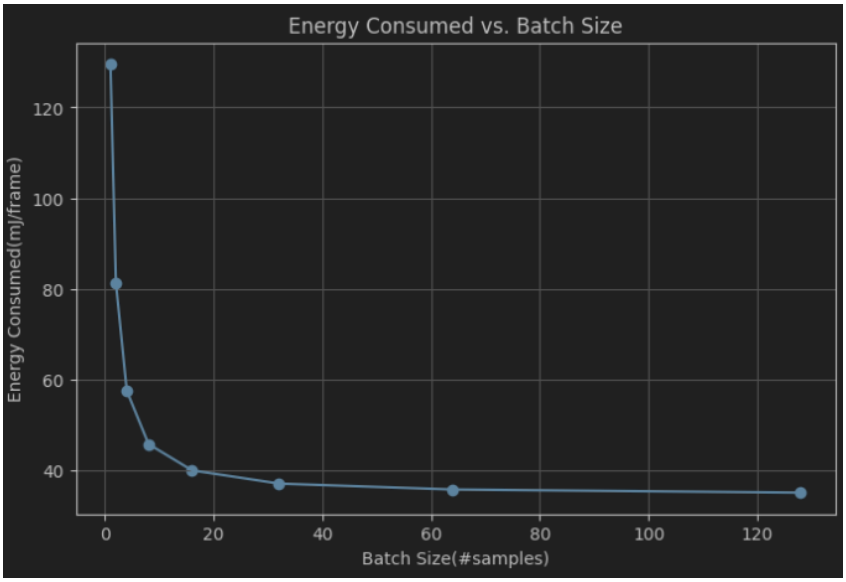


Рисунок 7. Залежність використання електроенергії від розміру батчу

Ключовим етапом при тестуванні продуктивності роботи нейроморфного чипа є заміри використання електроенергії (рис. 7). Метою даного експерименту було оцінити енергоефективність Akida AKD1000 під час обробки зображень у залежності від розміру батчу (Batch Size). Ключовим показником була витрата енергії на обробку одного зображення, виміряна в міліджоулях на кадр (mJ/frame). Аналогічно до попереднього експерименту, розмір батчу змінювався кратно степеню двійки (2^k, де k=0,…,7), щоб оцінити, як енергоефективність змінюється при збільшенні кількості оброблюваних зображень одночасно. Для отримання вищезазначеної метрики спершу було розраховано середню потужність пристрою в mW: для кожного розміру батчу було виконано по 100 замірів за допомогою програмного забезпечення від BrainChip, а результати усереднено. Для переходу в одиниці вимірювання mJ/frame, середню потужність чипа для кожного розміру батчу було поділено на відповідні значення FPS, отримані протягом попереднього експерименту.

Зібрані результати свідчать про те, що зі збільшенням Batch Size спостерігається експоненційне зниження витрат енергії на кадр.

- Для малої кількості зображень (1–16): Витрати енергії зменшуються досить різко, що свідчить про ефективну оптимізацію розподілу ресурсів чипа при збільшенні кількості оброблюваних зображень.
- Для великої кількості зображень (16–128): Темп зниження енергоспоживання сповільнюється, і метрика наближається до точки насичення (~35 mJ/frame).

В свою чергу, експоненційне зменшення енергоспоживання на один кадр пояснюється тим, що при обробці більшої кількості зображень значно зменшуються витрати на передачу

даних та ініціалізацію обчислень за рахунок паралельного виконання операцій і зниження простоїв.

Згідно з отриманими даними, при Batch Size ≥ 64 витрати енергії наближаються до стабільного рівня ~35 mJ/frame. Така продуктивність дозволяє чипу працювати довгий час при обмеженому енергозабезпеченні, що є критично важливим для автономних систем, таких як роботи, дрони або сенсорні мережі.

4. ВИСНОВКИ

Проведене дослідження демонструє значний потенціал нейроморфного чипа Akida AKD1000 для використання в автономних системах. Аналіз продуктивності показав, що чип здатний обробляти дані в реальному часі, досягаючи стабільного FPS навіть при збільшенні розміру батчу. При Batch Size ≥ 64 швидкість обробки стабілізується на рівні ~35 FPS, що перевищує мінімальні вимоги для додатків із часовими обмеженнями, таких як відеоаналітика чи робототехніка.

Енергоефективність чипа, виміряна в mJ/frame, показала експоненційне зменшення енергоспоживання на один кадр зі збільшенням Batch Size. Це підтверджує, що Akida AKD1000 ефективно розподіляє обчислювальні ресурси, мінімізуючи витрати енергії, особливо при великих обсягах даних. Показник енергоспоживання наближається до стабільного рівня ~35 mJ/frame при Batch Size ≥ 64, що є важливим для систем із жорсткими енергетичними обмеженнями.

Окрім переваг у швидкодії та енергоефективності, нейроморфний чип демонструє конкурентоспроможну точність після конвертації у спайкову нейронну мережу (SNN). Навіть із впровадженням агресивного квантування точність моделі зменшилась лише на 2,51%, що вказує на її здатність зберігати продуктивність за умов обмежених апаратних ресурсів.

Таким чином, Akida AKD1000 має високу ефективність для завдань реального часу, забезпечуючи оптимальний баланс між продуктивністю, енергоспоживанням та точністю. Це підтверджує доцільність його використання в автономних системах, де критично важливі енергоефективність та швидкодія.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Задорожний. Біологія з поглибленим вивченням. 8 клас. С. 82
2. Loihi - Intel. WikiChip. URL: https://en.wikichip.org/wiki/intel/loihi (дата звернення 15.10.2024)
3. Davies M. Loihi: A Neuromorphic Manycore Processor with On-Chip Learning. 2018. URL: https://redwood.berkeley.edu/wp-content/uploads/2021/08/Davies2018.pdf (дата звернення 15.10.2024)
4. A Look at Akida - BrainChip. Open Neuromorphic. URL: https://open-neuromorphic.org/neuromorphic-computing/hardware/akida-brainchip/ (дата звернення 15.10.2024)
5. Thorpe S., Gautrais J. Rank Order Coding. Computational Neuroscience. 1998. С. 113-118. URL: https://link.springer.com/chapter/10.1007/978-1-4615-4831-7_19 (дата звернення 18.10.2024)
6. CNN2SNN toolkit API. Akida Docs. URL: https://doc.brainchipinc.com/api_reference/cnn2snn_apis.html#cnn2snn.compatibility_checks.check_model_compatibility (дата звернення 18.10.2024)
7. Quantization for Neural Networks. Lei Mao. URL: https://leimao.github.io/article/Neural-Networks-Quantization/ (дата звернення 18.10.2024)

ХМАРНА ГЕНЕРАЦІЯ ТА ЛОКАЛЬНИЙ РЕНДЕРИНГ НЕСКІНЧЕННИХ ЛАНДШАФТІВ

Білан В.А.[1], Безносик О.Ю.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] bilan.vadim@lll.kpi.ua, [2] beznosyk.oleksandr@lll.kpi.ua [0000-0003-2775-6070]

Метою даного дослідження є створення кластера для хмарної генерації та клієнта для локального рендерингу нескінченних ландшафтів. Наукова новизна роботи полягає у новому підході компонування візуальних даних і організації віддаленого доступу до них у реальному часі для рендерингу за допомогою сервісної архітектури. Це усуває недоліки додатків, що працюють із 3D сценами локально. Практична цінність дослідження полягає у застосуванні реалізованої системи для задач візуалізації моніторингу на основі просторових даних, спільного редагування нескінченних ландшафтів у 3D-редакторах, для синхронізації стану в багатокористувацьких комп'ютерних іграх.

Ключові слова: кластер, нескінченні ландшафти, компонування візуальних даних, chunk, конвеєр генерації.

1. ВСТУП

Досягнення в області телекомунікацій, хмарних обчислень, сервісів, комп'ютерного обладнання та штучного інтелекту, накопиченні за останні 25 років, дозволяють виконати перехід від програмних систем із періодичним оновленням до оновлення у реальному часі.

Програми із періодичним оновленням контенту мають обмеження, що не дозволяють їм виконувати нові задачі. Їхній підхід до архітектури є застарілим і не задовольняє усе більш персоналізовані вимоги користувачів. Є цілий спектр задач, де оновлення у реальному часі є необхідним: моніторинг вбудованих систем, віртуальні світи, колективне редагування 3D сцен, безперервні ігрові оновлення.

У даному досліджені розглядається проблема компонування 3D сцен із декількох джерел даних і їхнє відображення у реальному часі. Компонування 3D сцен довільного розміру потребує великих обчислювальних ресурсів, пам'яті та часу, що не дозволяє робити оновлення у реальному часі на стороні клієнта. Щоб вирішити дану проблему пропонується такий підхід – розділити проблему на дві складові: виконувати компонування 3D сцен у масштабованому кластері із кешуванням результату; виконувати порціонне отримання композиції 3D сцен та рендеринг на стороні клієнта за запитом користувача.

2. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

2.1. Формулювання проблематики дослідження

Хмарні сервіси використовуються для широкого спектру прикладних задач, як-от моніторинг вбудованих систем, управління ресурсами, спільна робота над проектами, зберігання та доступ до даних, хостинг медіа, організація спільних відеоконференцій, навчання та використання нейронних мереж, надання послуг тощо. Ці задачі не потребують високих вимог до пропускної спроможності і затримки мережі.

Проте є ряд задач, для яких й досі відсутні стандартні рішення або ж вони не набули

широкої популярності для переходу від локальних програм до хмарних сервісів. До цієї категорії можна віднести: спільне редагування 3D сцен, віртуальні світи із багатьма користувачами, моніторинг у реальному часі 3D простору, запуск ігрових сесій на сервері, виконання запитів до нейронних мереж із генерацією великих об'ємів даних. Ці задачі потребують наднизької затримки, великої пропускної здатності, що не може забезпечити на великі відстані більшість сучасної мережевої інфраструктури.

У якості проблеми для дослідження було взято компонування 3D сцен із декількох джерел даних і їхнє відображення у реальному часі. Компонування 3D сцен довільного розміру потребує великих обчислювальних ресурсів, пам'яті та часу, що не дозволяє робити оновлення у реальному часі на стороні клієнта, тому рішення має бути пов'язаним із хмарними сервісами.

2.2. Існуючі підходи для вирішення обраної проблеми

На даний момент для вирішення задач, що потребують наднизької затримки і великої пропускної здатності, існує декілька підходів, які пов'язані з: обмеженням географії користувачів, зменшенням відстані від серверів до користувачів, використанням стандартів зв'язку із великою пропускною здатністю, меншою затримкою; розподілом обробки даних між вузлами кластера, кешуванням даних; зберіганням даних якомога ближче до клієнтів і зменшенням кількості операцій обміну даними у середині кластеру.

Існуючі підходи не набули широкої популярності через незадовільний рівень затримки, коштовність і недостатню поширеність зв'язку із великою пропускною здатністю. До того ж, щоб зберігати великі масиви даних, потрібно використовувати хостинг баз даних, що при таких розмірах даних не є дешевим.

Більшість сучасних додатків із 3D графікою використовують готові візуальні файли, що завантажуються разом з додатком, завантажуються перед самим запуском додатка або зберігаються локально. Це зумовлено необхідністю якомога швидше виконати рендеринг, щоб покращити досвід користування додатком. Якість цього досвіду формується частотою кадрів і швидкістю реагування, тобто рівнем затримки. Як наслідок, вимоги до організації даних і системи пам'яті збільшуються порівняно із використанням неграфічних додатків.

Але такий підхід обмежує спектр задач, який можуть виконувати додатки для користувачів. Додаток потрібно регулярно оновлювати, щоб отримати новий контент. Незручно переглядати дані, які не можуть зберігатися у повному об'ємі на стороні клієнта. Є обмеження на використання створених власноруч графічних даних, а також даних, згенерованих штучним інтелектом або симуляціями.

Є системи, що частково вирішують згадану проблему: Google Earth Engine, Mixer Blender Addon, Photon Quantum, Roblox, FAST3DIS, GullsEye TOS. Проте, якщо розглядати їх окремо, вони не можуть вирішувати одночасно декілька задач:

- обробку необмежених у просторі візуальних даних;
- можливість масштабування під різні потреби;
- доступність для довільної кількості клієнтів, що хочуть отримати доступ до оброблених візуальних даних;
- можливість зміни процесу компонування даних і вибору вхідних даних;
- відсутність обмежень у рамках одної практичної області, універсальність.

2.3. Постановка задач

Обробка необмежених за обсягом візуальних даних із розширенням доступу для багатьох клієнтів не може бути виконаною на персональних пристроях через обмеження у обчислювальних ресурсах, складність збереження і синхронізації даних. Тому пропонується розділити рішення на дві задачі: створення кластеру для обробки, синхронізації стану,

збереження візуальних даних; та створення клієнта, який отримує доступ до скомпонованих візуальних даних і виконує їхнє відображення.

Щоб надати практичного значення вирішуваній проблемі, у якості об’єкту генерації обрано ландшафт, необмежений у просторі. У результаті має бути створена система, в якій кластер повинен:

- зберігати світи і оновлювати їхній стан;
- за запитами клієнтів компонувати і кешувати візуальні дані для окремих фрагментів (chunk) світів [1].

Клієнти:

- визначають, як повинні компонуватися візуальні дані;
- відправляють запити для отримання фрагментів залежно від власного розташування.

Метою даної роботи є створення системи для оновлення візуальних даних у реальному часі шляхом хмарної генерації і локального рендерингу нескінченних ландшафтів.

Щоб досягнути поставленої мети, необхідно виконати такі задачі: створити модель генерації ландшафтів; створити кластер, що обробляє запити користувачів щодо надання графічних даних для локального рендерингу нескінченних ландшафтів; створити клієнт, який використовує модель локального рендерингу ландшафтів; перевірити працездатність системи шляхом збору метрик і аналізу результатів.

3. ПРОЕКТУВАННЯ КЛАСТЕРУ ДЛЯ ГЕНЕРАЦІЇ ЛАНДШАФТІВ

3.1. Визначення об'єкту генерації та його конфігурації

Щоб надати прикладного значення розроблюваному кластеру, необхідно визначити, що буде об’єктом генерації. Об’єкт генерації – світ, що не обмежений у просторі, у якому розміщуються візуальні дані і який формується за допомогою конвеєру генерації.

Конвеєр генерації – це ациклічний граф (рис. 1), на вхід якого подаються дані, а на виході скомпоновані фрагменти нескінченного ландшафту. Граф складається із вхідних, проміжних і вихідних вузлів. У якості вхідних джерел даних можуть використовуватися зовнішні джерела, генератори даних. На виході графу скомпоновані фрагменти ландшафтів кешуються і можуть бути використані клієнтами. Проміжні вузли є етапами обробки. Кожен вузол має власний інтерфейс, завдяки цьому можна формувати нескінченну кількість комбінацій і утворювати найрізноманітніші результати, що робить систему достатньо гнучкою. Ациклічний граф може мати декілька виходів, які можна назвати шарами. Шари можна вмикати і вимикати за потреби.

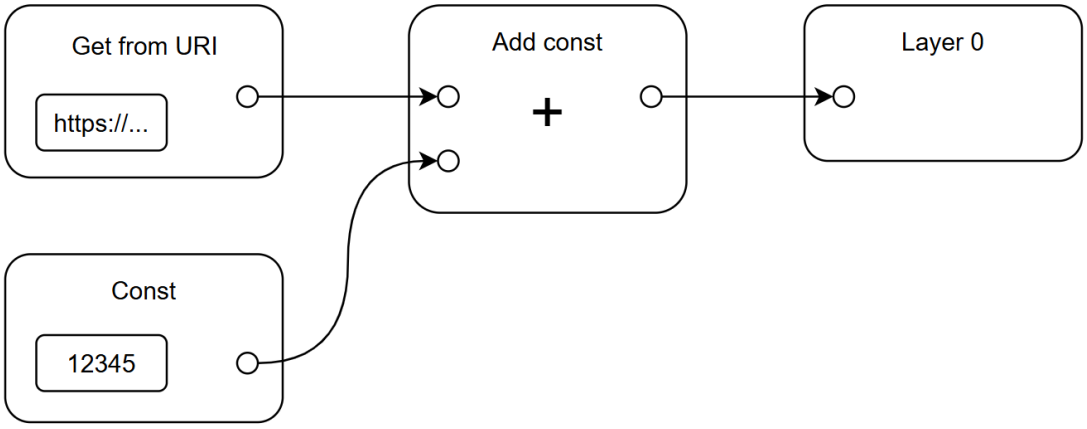


Рисунок 1. Приклад конвеєру генерації для налаштування створення фрагментів власних світів

У якості вузла конвеєру генерації може бути:

- алгоритм – функція, яка на вході отримує координати простору і інші параметри, щоб на виході згенерувати частину простору;

- існуюча модель, сцена або група моделей – візуальні дані, у поширеному форматі розповсюдження 3D моделей, які містять дані про геометрію, текстури, матеріали, анімації тощо.

У даній роботі будуть підтримуватися вузли на основі таких алгоритмів і даних: wave function collapse [2] (рис. 2, а [3]), безкоштовні моделі, отримані за допомогою Sketchfab API [4] (рис. 2, б [5]), map tiles (рис. 2, в [6]).

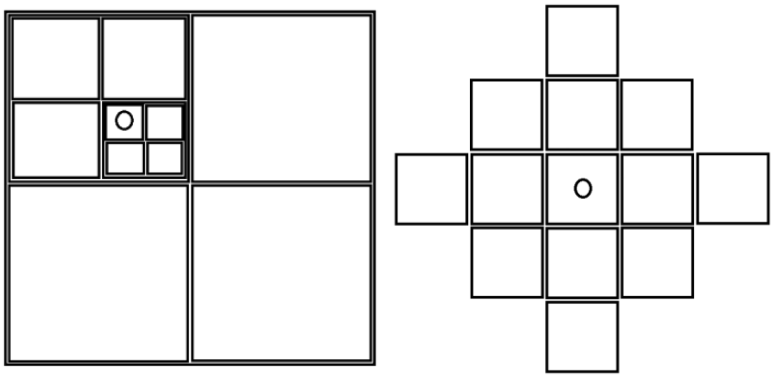
а) б) в)

Рисунок 2. Приклади об’єктів, які можуть бути використані для композиції світу

3.2. Поділ нескінченного ландшафту на частини

Мета даної роботи передбачає доступ клієнтів через мережу до візуальних даних, сформованих за допомогою конвеєра генерації світів. Світи можуть бути необмежені у просторі, тому не можуть бути розміщенні повністю у пам’яті клієнтів. Клієнт одночасно потребує лише частини світу, яка буде відображатися у нього на екрані. Це означає, що світи мають якимось чином розділятися у просторі на частини. Поділ світу має відбуватися на частини однакового розміру, які ідентифікуються унікально за глобальними координатами. Фрагмент світу представлений як форматовані візуальні дані, що обмежені у просторі.

Коли клієнт хоче візуалізувати у себе локально світ, йому потрібно задати ділянку простору, у якій буде відображатися частина світу. Клієнт надсилає запит до сервера із параметрами ділянки, де має відбуватися візуалізація світу, а сервер повертає набір фрагментів для ділянки відображення. Організація фрагментів відбуватиметься гібридним чином із поєднанням двох моделей, що зображені на рис. 3. Сервер, отримавши запит від клієнта із параметрами ділянки відображення, формує набір фрагментів, що визначається за допомогою структури даних quadtree [7] (рис. 3, а). Квадранти, у яких не перебуває клієнт, не будуть ділитися на частини, і навпаки, якщо квадрант містить позицію клієнта, він буде ділитися на 4 рівні частини до самих листків. До отриманих фрагментів також додаються фрагменти, відстань яких до клієнта менше за вказаний радіус, за моделлю на рис. 3, б. Вони перекривають фрагменти більшого масштабу.



а) б)

Рисунок 3. Варіанти кешування фрагментів навколо поточної позиції клієнта:
а) Quadtree; б) обмеження радіусом рендерингу

3.3. Архітектура кластера

Нижче коротко описано функції та інтерфейс кожного сервісу створеної архітектури [8].

Database – база даних для зберігання глобального стану. **Landscape Generator** зв'язується із **Database**, щоб оновити глобальний стан. **Database** передає оновлення стану іншим **Landscape Generator**-ам із застарілим кешом. Для реалізації обрано CouchDB.

Landscape Generator – генератор фрагментів (частин глобального стану), що приймає і узгоджує зміни стану від **Request Handler-**ів та **Database**. Кожна репліка **Landscape Generator-**а обробляє унікальну частину згенерованого глобального стану (фрагмент). Рідко можна зустріти ситуацію, коли усі клієнти знаходилися б в одному місці світу. Зазвичай вони розподілені нерівномірно. Тому це дозволяє знизити гіпотетичне навантаження на одну репліку. **Landscape Generator** використовує зовнішні API для отримання візуальних даних. Сповіщає інші сервіси через канали про делегування обробки конкретних фрагментів. Відправляє за запитами клієнтів візуальні дані фрагментів із власного локального кешу.

Request Handler – обробляє запити клієнтів, відправляє **Landscape Generator**-ам події від клієнтів, як-от запити про генерацію фрагментів, повертає клієнтам посилання на візуальні дані фрагментів.

Як **Request Handler** дізнається про те, які фрагменти зберігають репліки **Landscape Generator-**а? Він отримує цю інформацію за допомогою програмних каналів подій, використовуючи **Event Broker**-и [9]. Кожна репліка **Landscape Generator**-а надсилає через відповідний канал подію, що інформує **Event Broker-**и та інші **Landscape Generator-**и про делегування конкретного фрагменту саме йому. Таким чином, **Event Broker**-и надсилають запити за фрагментами конкретним **Landscape Generator**-ам, і інші **Landscape Generator**-и не генерують існуючі фрагменти. Канали можна використати також для оновлення глобального стану.

Event Broker – забезпечує роботу програмних каналів, якими користуються інші сервіси. Зменшує навантаження на мережу шляхом зберігання буферу подій, що може бути прочитаним будь-яким сервісом, без необхідності регулярного посилання подій усім сервісам. Для реалізації обрано Kafka.

Використання реплік створює нові проблеми. Репліки потребують управління трафіком, балансування навантаження [10], виявлення збоїв і відновлення після них [11]. Цю задачу можна делегувати таким системам як **Service Mesh** [12]. Для реалізації обрано Istio.

На рис. 4 показано схему архітектури, яка використовується для реалізації кластера для генерації нескінченних ландшафтів.

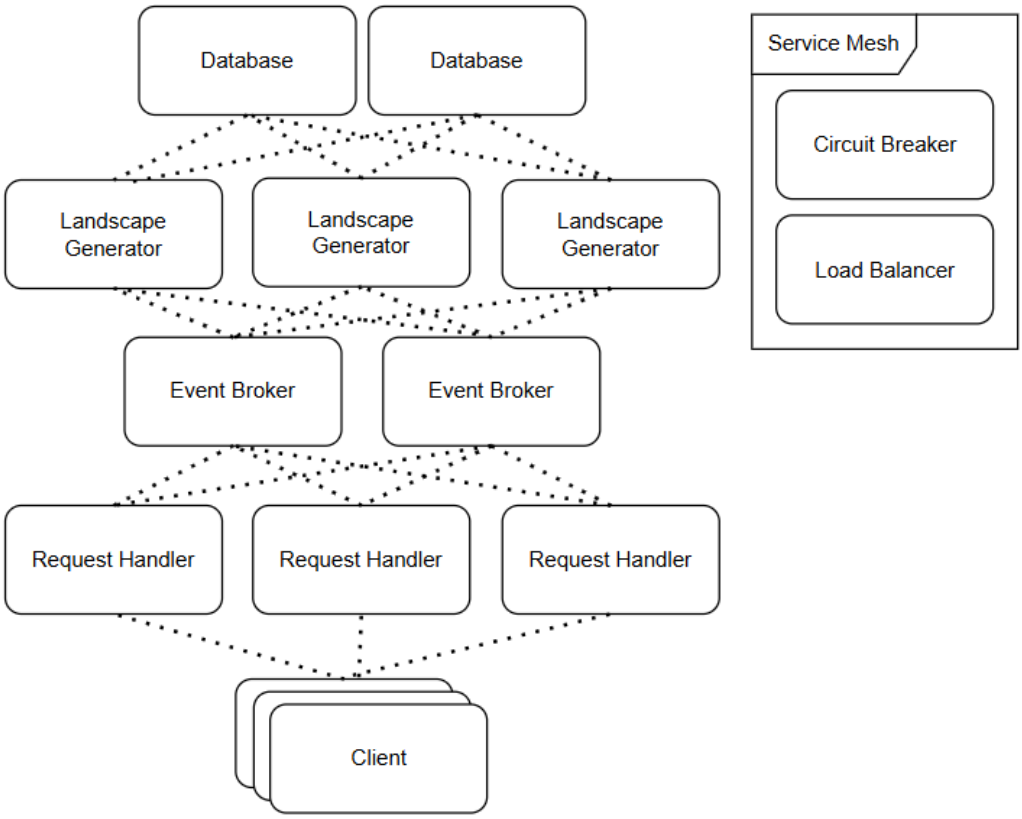


Рисунок 4. Архітектура розроблюваного кластера

4. ПРОЕКТУВАННЯ КЛІЄНТА ДЛЯ НАДСИЛАННЯ ЗАПИТІВ КЛАСТЕРУ ТА РЕНДЕРИНГУ ОТРИМАНИХ ЛАНДШАФТІВ

Процес візуалізації клієнтом світів, що зберігаються у хмарі, складається із таких етапів: формування і відправка запиту до кластеру на отримання фрагментів; отримання списку посилань на візуальні дані для фрагментів; формування і відправка запитів щодо отримання візуальних даних фрагментів із кластеру; завантаження візуальних даних фрагментів; рендеринг фрагментів. Запит до кластера може супроводжуватися зміною конвеєра генерації світів.

Клієнт складається із таких частин:

- **App** – містить усі компоненти клієнта, має метод loop для циклічного завантаження, кешування, оновлення і відображення фрагментів, відправки запитів на сервер;
- **Client** – представляє собою інтерфейс для паралельної обробки запитів до сервера із боку **App**;
- **Renderer** – відображає фрагменти;
- **InputHandler** – обробляє введення користувача для оновлення **LocalState**;
- **World** – зберігає стан світу, а саме конвеєр генерації для редагування, кеш фрагментів, локальний стан (**LocalState**) (наприклад, позицію користувача у просторі і ділянку для відображення). Має можливість фільтрувати шари.

5. ВИСНОВКИ

В роботі було проаналізовано існуючі підходи та системи для вирішення поставленої проблеми, приведено аргументи, чому проблема ще досі не має завершеного рішення, запропоновано власне рішення, визначено об’єкт генерації, спосіб його внутрішнього

представлення, методи генерації ландшафтів, спосіб поділу світу на окремі частини (фрагменти), їхнє кешування для зменшення навантаження на клієнт, представлено архітектуру кластеру та описано його сервіси, описано структуру клієнта, принци роботи кластера та клієнта.

Результатом роботи є розробка системи, що має вирішити поставлену задачу, а саме компонування 3D сцен із декількох джерел даних і їхнє відображення у реальному часі. Здійснено це було шляхом створення системи для оновлення візуальних даних у реальному часі шляхом хмарної генерації у кластері і локального рендерингу на клієнті нескінченних ландшафтів.

Розроблена система усуває недоліки локальної роботи із візуальними даними і дозволяє довільній кількості клієнтів працювати із даними необмежених розмірів завдяки перевагам хмарної архітектури. У подальшому створене рішення можна розширяти для виконання прикладних задач, як-от моніторинг на основі просторових даних, спільне редагування ландшафтів, організація ігрових сесій для мультиплеєрних ігор або виконання оновлень контенту у реальному часі.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Чанк. Minecraft Wiki Fandom: веб-сайт. URL: https://minecraft.fandom.com/uk/wiki/%D0%A7%D0%B0%D0%BD%D0%BA (дата звернення 18.10.2024).
2. Wauthier Tristan. Using Wave Function Collapse algorithm for 2D and 3D level generation. Belgium, Bruges/Kortrijk : Howest.be, 2023, 46 p.
3. Thesis: Markov-Chain based Wave Function Collapse. Pronay Peddiraju: веб-сайт. URL: https://www.pronay.me/wp-content/uploads/2020/03/WFCBanner-825x510.png (дата звернення 18.10.2024).
4. Download glTF samples. Sketchfab: веб-сайт. URL: https://sketchfab.com/features/gltf (дата звернення 18.10.2024).
5. FREE 1975 Porsche 911 (930) Turbo. Sketchfab: веб-сайт. URL: https://sketchfab.com/3d-models/free-1975-porsche-911-930-turbo-8568d9d14a994b9cae59499f0dbed21e (дата звернення 18.10.2024).
6. Google Map Tiles. Google Map Tiles: веб-сайт. URL: https://mt1.google.com/vt/lyrs=s&x=1321&y=3143&z=13 (дата звернення 18.10.2024).
7. Дерево квадрантів. Wikipedia: веб-сайт. URL: https://uk.wikipedia.org/wiki/%D0%94%D0%B5%D1%80%D0%B5%D0%B2%D0%BE_%D0%BA%D0%B2%D0%B0%D0%B4%D1%80%D0%B0%D0%BD%D1%82%D1%96%D0%B2 (дата звернення 18.10.2024).
8. R. Shea, J. Liu, E. C.-H. Ngai and Y. Cui, "Cloud gaming: architecture and performance". In: *IEEE Network*, том 27, № 4, с. 16-21.
9. What is an event broker? Solace: веб-сайт. URL: https://solace.com/what-is-an-event-broker/ (дата звернення 18.10.2024).
10. Load balancing (computing). Wikipedia: веб-сайт. URL: https://en.wikipedia.org/wiki/Load_balancing_(computing) (дата звернення 18.10.2024).
11. Service Mesh. Istio: веб-сайт. URL: https://istio.io/latest/about/service-mesh/ (дата звернення 18.10.2024).
12. Circuit Breaking. Istio: веб-сайт. URL: https://istio.io/latest/docs/tasks/traffic-management/circuit-breaking/ (дата звернення 18.10.2024).

ЗАСТОСУВАННЯ ПАТЕРНІВ ПРОЄКТУВАННЯ МІКРОСЕРВІСНИХ АРХІТЕКТУР ДЛЯ РОЗРОБКИ СИСТЕМИ ВІДСТЕЖЕННЯ ТРАНСПОРТУ

Бут І.О.[1], Булах Б.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] but.ihor@lll.kpi.ua, [2] bogdan.bulakh@gmail.com [0000-0001-5880-6101]

Метою даного дослідження є огляд різних патернів, які використовуються при розробці мікросервісних додатків з метою їх застосування для системи виставлення рахунку користувачам платних автомобільних доріг. Наукова новизна роботи полягає у застосуванні патернів проектування для оптимізації процесів в системі автоматичного виставлення рахунків за користування платними дорогами. Практична цінність дослідження полягає у розробці адаптивної системи, яка може інтегруватися, масштабуватися та оновлюватися в залежності від країни розгортання та типів пропускних пунктів на автобанах, забезпечуючи автоматизацію та високу продуктивність при обслуговуванні користувачів.

Ключові слова: мікросервісна архітектура, патерни проєктування, платні автомобільні дороги, розпізнавання.

1. ВСТУП

Постійний розвиток технологій значно спростив життя людей, автоматизуючи багато повсякденних завдань. Автоматизовані системи дозволяють швидше виконувати роботу і витрачати менше зусиль та дій на неї, а також надавати послуги без використання додаткових пристроїв, тощо. Автоматизація зменшує ризики, які пов’язані з людським фактором або механічними пристроями.

Покращення якості надання послуг є одним з пріоритетів як компаній, так і держав, адже саме це забезпечує позитивний користувацький досвід і сприяє сталому розвитку. Платні автомагістралі не стали виключенням, і нові підходи для покращення сервісу оплати за використання доріг впроваджуються досить рідко через масштабність проектів та потрібних фінансів на пункти пропуску. Тому нові підходи мають бути довготривалими, зручними та безвідмовними.

Поєднання штучного інтелекту та мікросервісної архітектури дозволяє створити систему, для оновлення якої достатньо покращити програмні компоненти, або швидко додати обчислювальної потужності без потреби модернізувати кожен наявний пункт пропуску з метою покращень. Використання комп’ютерного зору дозволить класифікувати транспортний засіб та ідентифікувати його за номером, в той час як мікросервісна система проведе розрахунок подорожі, надасть користувачу зручні способи заплатити гроші за користування дорогою а адміністрації вчасно відслідковувати стан системи та дорожнього руху. А нові ідеї легко додаються шляхом модифікації якихось сервісів в системі або додаванням нових.

2. ПАТЕРНИ ПРОЄКТУВАННЯ МІКРОСЕРВІСНИХ СИСТЕМ

При розробці та розвитку будь якої системи, в тому числі якщо вона побудована на мікросервісній архітектурі, постають виклики та задачі, які є однаковими практично для всіх систем. Таким чином, при розробці системи розробники використовують патерни проєктування – стандартизовані рішення повторюваних проблем у розробці програмного забезпечення, що надають адаптивні методи для вирішення конкретних завдань у додатку. Застосовуючи ці патерни, розробники спрощують процес розробки, оминаючи необхідність ретельного дослідження кожного питання і натомість зосереджуючись на реалізації унікальних аспектів системи. Такий підхід економить час завдяки використанню перевірених рішень для загальних компонентів[1].

Кожен з патернів може належати до своєї групи патернів в залежності від їх ролей та задач. В залежності від області застосування, обираються потірбні патерни реалізації. Патерни цих груп діляться на ті, що забезпечують міжсервісну комунікацію, управління даними, забезпечення відмовостійкості, масштабованості, безпеки.

Після дослідження патернів проєктування мікросервісних систем та їх особливостей, було вирішено сконцентруватися на реалізації таких важливих патернів як api-шлюз[2], патерну міжсервісної комунікації event-driven[3], CQRS[4] та Circuit Breaker[5].

API-шлюз – централізована точка входу для всіх запитів до мікросервісів, яка діє як проксі-сервер для перенаправлення запитів до відповідних сервісів. API-шлюз спрощує взаємодію з клієнтом, об’єднуючи кілька викликів API в один запит, таким чином зменшуючи складність на стороні клієнта. Шлюз API налаштовується для виконання автентифікації, авторизації, балансування навантажень, кешування та обмеження швидкості.

Event-Driven архітектура – шаблон проєктування, який дозволяє відокремленим сервісам взаємодіяти асинхронно, публікуючи події та підписуючись на них через брокер подій. Ця архітектура забезпечує потік даних у реальному часі між сервісами та підключеними пристроями в міру того, як події відбуваються в системі, що дозволяє швидко реагувати на зміни. Впроваджуючи брокер подій, архітектура керована подіями сприяє вільному зв’язку, а отже додаткам не потрібно знати джерело або пункт призначення даних, які вони обробляють. Такий підхід підвищує масштабованість, відмовостійкість і гнучкість, оскільки компоненти можуть незалежно масштабуватися, виходити з ладу або оновлюватися, не впливаючи на всю систему.

Command Query Responsibility Segregation (CQRS) – розділяє операції читання і запису в системі, вирішуючи такі проблеми, як конфлікт даних і покращує продуктивність системи в складних середовищах. У CQRS сторона запису обробляє команди, які змінюють стан програми, і зберігає ці зміни. Сторона читання, з іншого боку, фокусується на створенні оптимізованих, часто сильно денормалізованих представлень даних для запитів, що дозволяє отримувати швидкі, адаптовані відповіді. Такий поділ дозволяє оптимізувати кожну сторону незалежно. Це означає, що сторону читання можна масштабувати для обробки частих запитів на отримання даних, не впливаючи на продуктивність сторони запису.

Circuit Breaker – запобігає перевантаженню сервісів, відслідковуючи їх працездатність. Він автоматично відключає запити до сервісу, якщо той стає нестабільним, даючи йому час на перезавантаження та стабілізацію, і знову дозволяє запити після перевірки його стану. Цей підхід захищає систему від каскадних збоїв, забезпечуючи стабільну роботу інших компонентів і запобігає збільшенню навантаження в умовах збоїв.

3. ОПИС ЧАСТИН СИСТЕМИ ТА АРХІТЕКТУРИ МІКРОСЕРВІСНОЇ ЧАСТИНИ

3.1 Опис системи

Система складається з двох умовних складових. Першою складовою є пункти пропуску транспорту, обладнаних камерами відеоспостереження з гарною роздільною здатністю та комп'ютерами, достатньої потужності для обробки відеоряду в режимі реального часу. Основною задачею цих пунктів є відстеження на в'їзд чи виїзд, класифікація транспортного засобу та зчитування номеру. Далі отримана інформація формується повідомлення, яке надсилається на головний сервер.

Другою складовою є сервер, на якому запущений мікросервісний додаток. Основною ціллю системи є прийом повідомлень від пунктів пропуску, проведення розрахунку вартості проїзду для транспортного засобу за номером, збереження цієї інформації. Для користувачів є розподіл по ролям, де водії транспортних засобів можуть переглянути історію своїх подорожей та сплатити за проїзд, а адміни зможуть розглядати та реагувати на помилки, які надходять від сервісів. Обидві частини роботи написані мовою програмування Python.

3.2 Пропускні пункти

Для реалізації програмного забезпечення для пропускного пункту використовується нейронні мережі для виконання задач комп'ютерного зору.

Для відстеження типу транспортного засобу та його номерного знаку скористаємося моделлю сімейства YOLO. YOLO зарекомендував себе як надійна модель комп'ютерного зору для роботи у реальному часі, з можливістю вибору моделі в залежності від наданих ресурсів. Yolo11 (рис. 1) [6] є самою новою випущеною моделлю, яка має кращі результати точності, швидкості та розміру моделі.

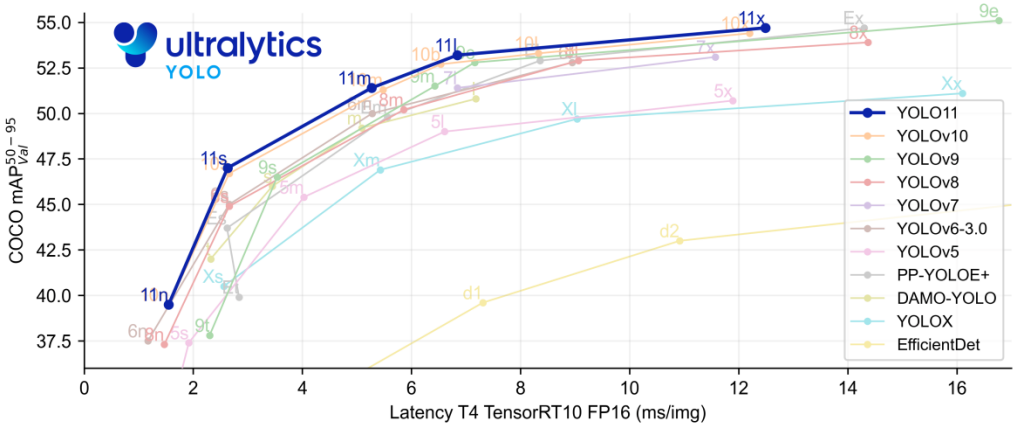


Рисунок 1. Порівняння різних версій моделей сімейства YOLO

Для виконання обробки номеру з номерного знаку буде використовуватися OCR бібліотека для зчитування символів у Python. В результатів пунктом пропуску буде сформовано повідомлення для головної системи, що буде містити інформацію про транспортний засіб та пункт пропуску, необхідних для відстеження його руху дорогою.

3.3 Архітектура системи

Розроблена архітектура містить в собі як і сервіси, керовані подіями, так і сервіси які виконують синхронний запит – відповідь. Асинхронно сервіси опрацьовують інформацію від камер, записується інформація про подорож до бази даних, записується інформація про

можливі нештатні ситуації, пов’язані з тим, що транспортний засіб не був виявлений при в’їзді чи виїзді, через що неможливо розрахувати вартість користування дорогою транспортному засобу.

Синхронна комунікація відбувається у сервісах, пов’язаних з користувачем для операцій з профілем, переглядом історії транспортних подорожей, а також для збереження факту сплати за дорогу, що є дуже критичним для негайної доступності цієї інформації.

Розглянемо детальніше запропоновану архітектуру системи автоматизації оплати за користування платними дорогами та використані патерни на рисунку 2.

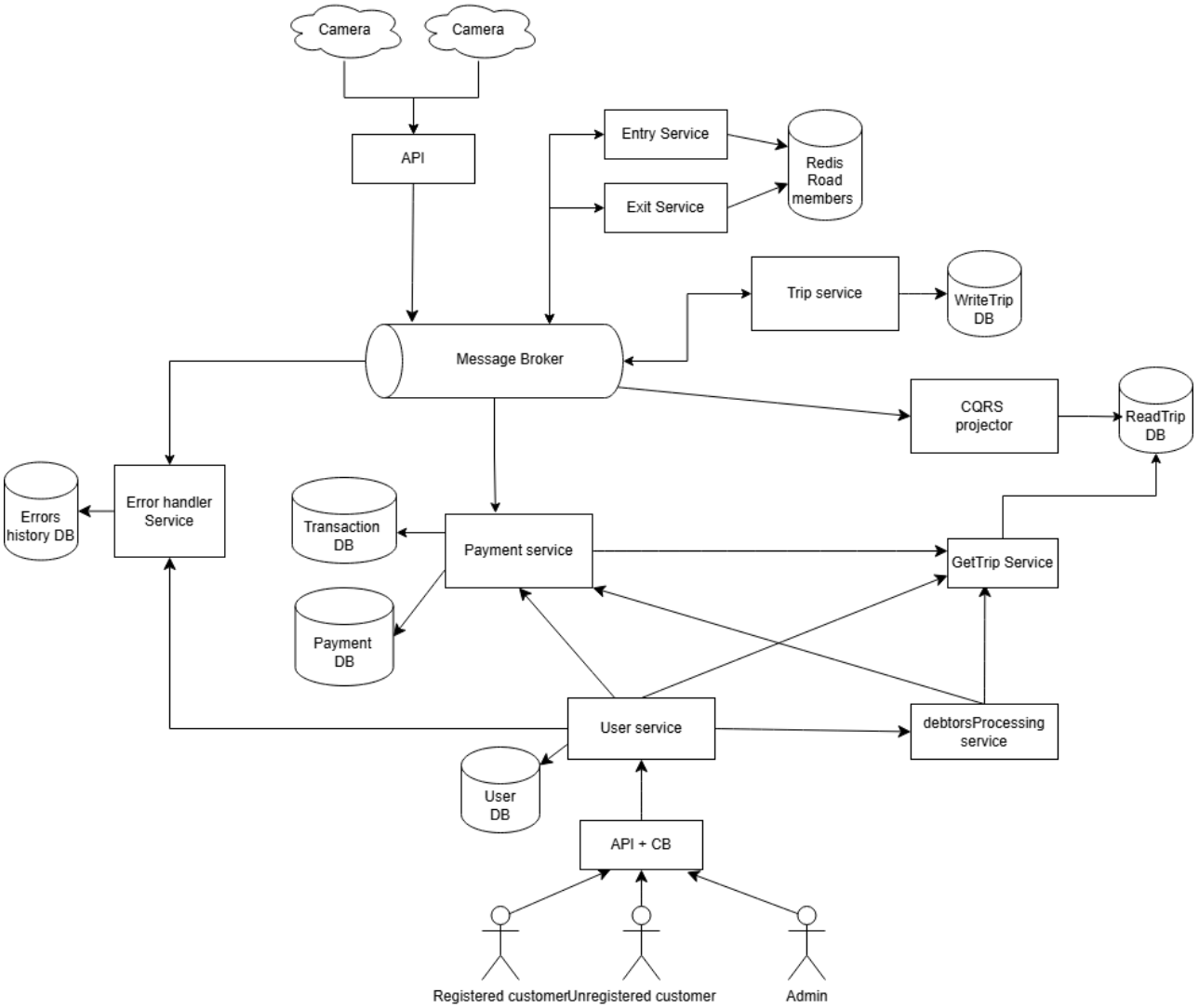


Рисунок 2. Архітектура розробленої системи

Розглянемо кожен сервіс та його задачу у системі. Камери – це зовнішні комп’ютери на пунктах пропуску, мета яких є передача даних до головної системи.

API шлюз (для віддалених камер) – відповідає за прийом повідомлень від камер та передачі їх брокеру повідомлень. Попередньо він підготовлює повідомлення до відправки, а також забезпечує захищене та безпечне з’єднання з системою.

Entry service – підписаний на брокер повідомлень і читає топік присвячений транспортним засобам, які заїхали на платні автомагістралі. Він виконує запис у кеш Redis з інформацією про транспорт, його тип, пункт в’їзду , дату та час в’їзду. Також перевіряє, чи

не був цей же номер транспортного засобу раніше записаний як той, що на дорозі. У випадку проблем, надсилає повідомлення про неточність в брокер повідомлень до відповідного топіку пропущених транспортних засобів пунктами пропуску.

Redis Road members – кеш, який зберігає інформацію про учасників дорожнього руху, які наразі перебувають на платних дорогах.

Exit service – підписаний на брокер повідомлень і читає топік присвячений транспортним засобам, які виїхали з платних автомагістралей. Він виконує запит до кешу Redis для перевірки, чи був цей транспортний засіб записаний при в'їзді на платні дороги. Якщо ні, він генерує відповідне повідомлення до брокера повідомлень для топіку пропущених транспортних засобів пунктами пропуску. Якщо транспортний засіб присутній, він видаляє його з кешу, формує повідомлення з повною інформацію про транспортний засіб, пункти пропуску та час, коли він їх проїхав та надсилає його у відповідний топік брокера повідомлень.

Message broker – брокер повідомлень, який відповідає за виконання асинхронної комунікації між сервісами.

Error handler – сервіс, який підписується на відповідний топік в брокері повідомлень для перегляду та зберігання конфліктої інформації, яка надходить з Entry та Exit сервісів.

Trip service – підписаний на брокер повідомлень і читає топік від Exit service. Отримана інформація використовується для розрахунку вартості користуванням транспортним засобом.

Для зберігання даних про подорож і її вартість реалізований патерн CQRS. Trip service записує інформацію до бази даних Write DB, призначеної для запису. Разом з цим публікується подія цього запису. CQRS projector реагує на подію і оновлює базу даних Read DB, яка використовується для читання сервісом GetTrip за запитами інших сервісів.

Payment service також підписаний на топік від Exit service. Він зберігає id подорожі та статус про його оплату до Payment DB. Інформація про транзакцію зберігається у базі Transaction DB.

debtorsProcessing використовується для виявлення подорожей, які не були оплачені. Цей сервіс може реалізуватися як Function-as-a-Service (FaaS), і викликатися раз на день чи тиждень. В залежності від країни, в якій розгортається система, можна редагувати функціонування сервісу на передавання інформації стосовно боржників до структур, які відповідають за штрафування осіб-боржників.

Для роботи з користувачами та їх даними використовуються нижня частина сервісів.

API шлюз з використанням патерну Circuit Breaker для відстеження працездатності user services. Він виконує перенаправлення до сервісів з використанням балансування навантаження.

User service надає різний функціонал в залежності від ролі. Для незареєстрованих користувачів він підтягує інформацію стосовно якогось конкретно введеного транспортного номеру. Для зареєстрованих користувачів буде доступна детальні інформація стосовно їх поїздок а також підтримка декількох номерів. Для адміна буде надано можливість переглядати конфлікті ситуації.

4. ВИСНОВКИ

Для підвищення швидкості розробки та надійності системи важливо використовувати патерни проєктування як невід'ємний елемент процесу. Єдиний концепт допомагає новим розробникам набагато швидше зрозуміти процес роботи системи, а використані патерни можуть бути взаємозамінні в залежності від змін цілей проєкту та задач.

Метою розробленої системи було об’єднання новітніх підходів для автоматизації буденних задач. Такою задачею стала автоматизація проходження пропускних пунктів на платних дорогах з метою покращення користувацького досвіду та проблем, які могли виникнути при старих підходах. Об’єднання штучного інтелекту та мікросервісної системи дозволяє створити швидке та автоматизоване рішення, яке має набагато менше шансів вийти з ладу.

Після дослідження особливостей та цілей патернів проєктування, була розроблена структура системи. Використання патернів забезпечує високу модульність, масштабованість і гнучкість. Використано патерни Event-Driven для асинхронної обробки подій, API Gateway як єдиний шлюз для запитів, Circuit Breaker для захисту сервісів від перевантаження та CQRS для розділення операцій запису і читання, що забезпечує високу продуктивність і гнучкість системи.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Richardson, C. Microservices Patterns: With Examples in Java. New York: Manning Publications, 2018. 520 p.

2. Kong. What is an API Gateway? Core Fundamentals and Use Cases. Kong Inc. URL: https://konghq.com/blog/learning-center/what-is-an-api-gateway (дата звернення: 08.10.2024).

3. Choosing the best communication type for your microservices. URL: https://www.opslevel.com/resources/choosing-the-best-communication-type-for-your-microservices (дата звернення: 11.10.2024).

4. Tech, C. O. 10 Microservices Design Patterns for Developers | Capital One | Capital One Tech. Medium. URL: https://medium.com/capital-one-tech/10-microservices-design-patterns-for-better-architecture-befa810ca44e (дата звернення: 14.10.2024).

5. Subhashana, H. Circuit Breaker pattern (Design Patterns for microservices). Medium. URL: https://medium.com/geekculture/design-patterns-for-microservices-circuit-breaker-pattern-276249ffab33 (дата звернення: 20.10.2024).

6. Ultralytics. YOLO11. Ultralytics YOLO Docs. URL: https://docs.ultralytics.com/models/yolo11 (дата звернення: 01.11.2024).

РОЗПІЗНАВАННЯ АНОМАЛІЙ ТА ПЕРЕДОБРОБКА ДАНИХ ЧАСОВИХ РЯДІВ НА ПРИКІНЦЕВИХ ПЛАТФОРМАХ

Васін М.С.[1], Крайнік М.В.[2], Письменний І.О.[3]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] vasin.maksym@lll.kpi.ua, [2] krainik.mykyta@lll.kpi.ua,
[3] ihor.pismennyy@gmail.com [0000-0001-7648-2593]

У цьому дослідженні розглядаються методи розпізнавання аномалій у часових рядах для медичних даних, зокрема звуків серцебиття і дихання. Метою є визначення оптимальних підходів для використання на прикінцевих платформах. Використано статистичні методи, алгоритм XGBoost і глибокі нейронні мережі (RNN, CNN). Отримані результати дозволяють обрати оптимальний метод для виявлення аномалій при обмежених ресурсах, що є важливим для медичних додатків.

Ключові слова: розпізнавання аномалій, часові ряди, медичні дані, глибокі нейронні мережі, статистичні методи, прикінцеві платформи.

1. ВСТУП

У сучасному світі штучний інтелект стає невід'ємною частиною багатьох галузей, у тому числі й охорони здоров'я, де його застосування може суттєво підвищити ефективність діагностики та лікування. Зокрема, аналіз часових рядів, що містять життєво важливі показники пацієнтів (як-от звуки серцебиття та дихання), дозволяє своєчасно виявляти патологічні стани та реагувати на них. Однак, використання таких технологій супроводжується певними труднощами, як-от висока обчислювальна складність алгоритмів глибокого навчання, що потребують значних ресурсів.

Ця роботи спрямована на пошук оптимального методу (наприклад, статистичні алгоритми і глибокі нейронні мережі) для аналізу аномалій у часових рядах при обмежених ресурсах. Особлива увага приділяється можливості застосування даних методів на прикінцевих платформах, таких як мікроконтролери, де обчислювальні ресурси є обмеженими. Досягнення в цій галузі можуть значно сприяти впровадженню штучного інтелекту в медичні системи, що використовуються в реальних умовах, забезпечуючи більш доступну та оперативну допомогу.

2. ПОСТАНОВКА ЗАДАЧІ

Сучасні системи охорони здоров'я все частіше використовують аналітику даних для підвищення ефективності діагностики та лікування пацієнтів. Однією з ключових складових таких систем є моніторинг стану пацієнтів через збори даних у вигляді часових рядів. Ці дані можуть включати показники життєдіяльності, такі як звук биття серця, записаний за допомогою просто мікрофону чи спеціального обладнання, аудіофайл дихання, результати лабораторних досліджень, дані про фізичну активність і інші медичні показники. Для початку діагностування пацієнта потрібно розуміти в чому проблема. Одним з показників може слугувати наявність аномалій в показниках життєдіяльності.

Аномалії в часових рядах можуть бути викликані різними факторами, такими як помилки вимірювань, зміни в поведінці пацієнта, або ж реальні патологічні стани. Аномалії викликані помилками зазвичай називають шумом і вони не є цікавими для розгляду, тому їх відкидають. Виявлення реальних аномалій є критично важливим для своєчасного реагування медичних працівників.

Враховуючи масовість застосовуваної технології варто враховувати її вартість і доступність. Існуючі методи аналізу даних часто вимагають значних обчислювальних ресурсів, що може бути недоступно на кінцевих платформах, таких як мікроконтролери або вбудовані системи.

Метою даної роботи є розгляд існуючих методів розпізнавання аномалій в часових рядах, порівняння методів на прикінцевих платформах та вибір методу, який задовольнить всі критерії.

3. ВИКОРИСТАНІ ДАНІ

По-перше, дані медичного характеру зазвичай є чутливими та особистими. Збір такої інформації повинен відбуватися з дотриманням етичних норм і законодавчих вимог, включаючи дотримання принципів конфіденційності та захисту особистої інформації. У більшості країн збір медичних даних вимагає отримання інформованої згоди від пацієнтів, що може обмежити доступ до великого обсягу даних. Крім того, недостатня кількість погоджень може призвести до неповних або неструктурованих наборів даних, що ускладнює подальший аналіз.

По-друге, аномалії, які є об'єктом дослідження, можуть бути рідкісними, що також ускладнює збір достатньої кількості даних для тренування ефективних моделей. Дослідникам потрібно враховувати як кількісні, так і якісні аспекти даних, адже наявність нетипових ситуацій може призвести до значних викривлень у моделі.

Через вищезгадані причини було прийнято рішення використовувати вже готові набори даних. У свою чергу ми їх розділили на дві окремі групи.

Перша група містить аудіофайли звуків серцебиття, включаючи нормальні й патологічні зразки, та класифікацію на хвороби [1, 2].

Друга група включає аудіозаписи звуків респіраторної системи з класифікацією на хвороби та метаданими про процес запису даних [3–6].

4. ІСНУЮЧІ МЕТОДИ АНАЛІЗУ АНОМАЛІЙ ЧАСОВИХ РЯДІВ

На даний момент найсучаснішими технологіями, які показують найкращий результат є глибокі нейронні мережі. Проте вони потребують надзвичайно великих наборів даних, аугментації (штучного розширення набору за допомогою технік на кшталт зашумлення, масштабування чи генерування нових зразків на основі існуючих) та самі по собі є доволі громіздкими. Більші моделі налічують таку кількість ваг і параметрів, що пересічна людина не має можливості запустити модель на особистому комп’ютері. Це стосується і організацій: не кожна може дозволити собі підтримання дата-центру.

До попередників глибоких нейронних мереж можна віднести статистичні методи, які в свою чергу не мають таких обмежень. Простота статистичних моделей і відносно невелика кількість параметрів і проста структура дозволяє перш за все запускати дані моделі на значно гіршому обладнанні. На відміну від глибоких нейронних мереж, які зачасту представляють собою чорний ящик, статистичні моделі можливо зрозуміти та вивести «процес мислення».

4.1. Статистичні методи

Статистичні методи є основою для аналізу аномалій у часових рядах, оскільки вони дозволяють виявляти ненормативні патерни в даних за допомогою кількісних підходів.

В даний час доступно багато статистичних програм, таких як SPSS, R, Stata і SAS, і за допомогою цих програм можна легко виконати статистичний аналіз, але вибір відповідного статистичного тесту все ще залишається складним завданням для біомедичних дослідників, особливо для тих, хто не має математичної освіти [7].

Наведена таблиця 1 може бути використана для правильного підбору статистичного тесту, якщо дані є незалежними.

Таблиця 1. Вибір статистичного тесту для незалежних спостережень

	Вихідна змінна						
Вхідна змінна		Номінальна	Категоріальна (>2 категорій)	Порядкова	Кількісна дискретна	Кількісний не нормальний розподіл	Кількісна нормальний розподіл
	Номінальна	χ^2 або тест Фішера	χ^2	χ^2-тренд або Манна-Уітні	Манна-Уітні	Манна-Уітні або Логранк (log-rank) (a)	t-критерій Стьюдента
	Категоріальна (>2 категорій)	χ^2	χ^2	Краскал-Уоліс (b)	Краскал-Уоліс (b)	Краскал-Уоліс (b)	Дисперсійний аналіз (c)
	Порядкова	χ^2-тренд або Манна-Уітні	e	Коефіцієнт кореляції рангу Спірмена	Коефіцієнт кореляції рангу Спірмена	Коефіцієнт кореляції рангу Спірмена	Коефіцієнт кореляції рангу Спірмена або Лінійна регресія (d)
	Кількісна дискретна	Логістична регресія	e	e	Коефіцієнт кореляції рангу Спірмена	Коефіцієнт кореляції рангу Спірмена	Коефіцієнт кореляції рангу Спірмена або Лінійна регресія (d)
	Кількісний не нормальний розподіл	Логістична регресія	e	e	e	Граф з Пірсоном або Коефіцієнт кореляції рангу Спірмена	Граф з Пірсоном або Коефіцієнт кореляції рангу Спірмена і Лінійна регресія
	Кількісний нормальний розподіл	Логістична регресія	e	e	e	Лінійна регресія (d)	Пірсон і Лінійна регресія

a – якщо дані піддаються цензурі.

b – критерій Краскала-Уолліса використовується для порівняння порядкових або ненормальних змінних для більш ніж двох груп і є узагальненням U-критерію Манна-Уітні.

c – дисперсійний аналіз є загальним методом, і одна з його версій (односторонній дисперсійний аналіз) використовується для порівняння нормально розподілених змінних для більш ніж двох груп і є параметричним еквівалентом тесту Краскала-Уолліса.

d – якщо залежною змінною є змінна результату, то за умови, що залишки (різниця між спостережуваними значеннями і прогнозованими відповідями на основі регресії) розподілені за нормальним законом, розподіл незалежної змінної не є важливим.

e – існує низка більш досконалих методів, таких як регресія Пуассона, для вирішення цих ситуацій. Однак вони вимагають певних припущень, і часто простіше або дихотомізувати результативну змінну, або розглядати її як неперервну [8].

XGBoost. XGBoost (екстремальне градієнтне підсилювання) – це програмна бібліотека з відкритим кодом, яка пропонує рішення для градієнтного підсилювання для C++, Java, Python, R, Julia, Perl, та Scala.

Градієнтне підсилювання це техніка машинного навчання, заснована на підсиленні у функціональному просторі, де метою є псевдозалишки, а не залишки, як у традиційному підсиленні. Це створює модель прогнозування у вигляді ансамблю слабких моделей прогнозування, тобто тих, які роблять дуже мало припущень про дані, які зазвичай є простими деревами рішень [9]. Коли дерево рішень є слабким учнем, отриманий алгоритм називається градієнтно-підсиленими деревами; він зазвичай перевершує Random Forest[9]. Як і інші методи підсилення, модель градієнтно-підсилених дерев будується поетапно, але воно узагальнює інші методи, дозволяючи оптимізувати функцію втрат, яка може бути диференційована довільно.

4.2. Глибокі нейронні мережі

Глибокі нейронні мережі стали основним інструментом для аналізу аномалій у часових рядах завдяки своїй здатності моделювати складні, нелінійні залежності у даних. Вони показують вражаючі результати у багатьох застосуваннях, таких як фінансова аналітика, моніторинг здоров'я та прогнозування.

Рекурентні нейронні мережі (RNN). Рекурентні нейронні мережі є особливим типом нейронних мереж, які призначені для роботи з послідовними даними. RNN здатні зберігати інформацію про попередні стани через свої рекурентні зв'язки, що робить їх ідеальними для аналізу часових рядів.

Основна перевага RNN полягає в їхній здатності враховувати контекст, що дозволяє виявляти аномалії на основі інформації з попередніх часових точок. Однак, класичні RNN мають обмеження, пов'язані з проблемами затухання градієнта, що ускладнює їхнє навчання на тривалих послідовностях.

Для подолання обмежень класичних RNN були розроблені LSTM (Long Short-Term Memory) та GRU (Gated Recurrent Units). Ці архітектури використовують механізми гейтування, які дозволяють моделі зберігати та забувати інформацію за необхідності. LSTM і GRU демонструють високу ефективність у завданнях, пов'язаних з часовими рядами, таких як прогнозування медичних показників, де важливо враховувати тривалі патерни.

Згорткові нейронні мережі (CNN). Хоча CNN зазвичай асоціюються з обробкою зображень, їх також можна ефективно використовувати для аналізу часових рядів. CNN здатні автоматично виявляти локальні патерни у даних, що робить їх корисними для виявлення аномалій. Наприклад, вони можуть бути застосовані для аналізу динаміки серцевого ритму, де виявлення певних форм або коливань може свідчити про патологічні стани.

Трансформери, зокрема їх архітектура, побудована на основі механізму уваги, також знайшли своє застосування в аналізі часових рядів. Завдяки своїй здатності моделювати залежності на різних часових масштабах, вони можуть виявляти складні патерни, не вимагаючи послідовної обробки даних, що підвищує їхню ефективність і швидкість.

5. ПОРІВНЯННЯ МЕТОДІВ

Навіть в одній категорії – глибокі нейронні мережі – різні рішення мають варіативний успіх у вирішенні задач, також різні методи мають відмінні вимоги. Наприклад, рекурентні нейронні мережі (RNN), такі як довга короткочасна пам'ять (LSTM), зазвичай потребують менше ресурсів, ніж трансформери, які мають більшу обчислювальну потужність завдяки паралельній обробці послідовностей. У той же час LSTM можуть досягати точності, порівнянної з ранніми версіями трансформерів, що робить їх оптимальними для задач, де є обмеження щодо ресурсів. Це підтверджується результатами з Kaggle, де використання моделей з оптимальною структурою, такою як XGBoost, демонструє високу ефективність при меншій вимогливості до ресурсів, порівняно з іншими складними алгоритмами [10].

При порівнянні з іншими категоріями моделей різниця у вимогах та швидкості стає ще більш помітною. Статистичні моделі, як-от регресійний аналіз або методи класифікації, є надзвичайно легкими, але зазвичай поступаються у точності, особливо в задачах, що потребують складного аналізу нелінійних зв’язків, як це має місце в медичних і фінансових даних. Зокрема, результати Kaggle показують, що виявлення аномалій за допомогою дерев рішень (як у випадку LightGBM і XGBoost) забезпечує баланс між точністю та швидкістю, що дозволяє використовувати їх навіть в умовах обмежених обчислювальних ресурсів, без значного компромісу у точності [10].

6. ВИСНОВКИ

У цьому дослідженні було проаналізовано ефективність різних методів виявлення аномалій у часових рядах медичних даних, зокрема звуків серцебиття та дихання. Виявлено, що статистичні методи та XGBoost є придатними для задач, де пріоритетом є швидкість і мінімальні обчислювальні витрати, а глибокі нейронні мережі, зокрема RNN і CNN, дозволяють досягти вищої точності, хоча потребують більших ресурсів. Застосування обраних методів на прикінцевих платформах відкриває нові можливості для мобільної медицини, дозволяючи використовувати діагностичні системи з обмеженими ресурсами для своєчасного виявлення відхилень у стані здоров’я пацієнтів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ed King, "Heartbeat Sounds", *Kaggle*, Доступно: https://www.kaggle.com/datasets/kinguistics/heartbeat-sounds.

2. "HeartAnomalyDetection_HeartbeatSounds", *Hugging Face*, Доступно: https://huggingface.co/datasets/DynamicSuperb/HeartAnomalyDetection_HeartbeatSounds.

3. "RespiratorySoundClassification_ICBHI2017", *Hugging Face*, Доступно: https://huggingface.co/datasets/DynamicSuperb/RespiratorySoundClassification_ICBHI2017.

4. "Respiratory Sound Database", *BHI Challenge*, Доступно: https://bhichallenge.med.auth.gr/ICBHI_2017_Challenge.

5. E. F. S. Guy, J. A. Clifton, T. Caljé-van der Klei, R. Chen, J. Knopp, K. Moeller, and J. G. Chase, "Respiratory dataset from PEEP study with expiratory occlusion (version 1.0.0)", *PhysioNet*, 2023. Доступно: https://doi.org/10.13026/d767-e709.

6. B. M. Rocha, D. Filos, L. Mendes, I. Vogiatzis, E. Perantoni, E. Kaimakamis, P. Natsiavas, A. Oliveira, C. Jácome, A. Marques, and R. P. Paiva, "A Respiratory Sound Database for the Development of Automated Classification" in Precision Medicine Powered by pHealth and Connected Health, Singapore: Springer, 2018, pp. 51-55. [Online]. Доступно: https://eden.dei.uc.pt/~ruipedro/publications/Conferences/ICBHI2017a.pdf

7. J. Karan, "How to select appropriate statistical test?", Journal of Pharmacy Negative Results, vol. 1, pp. 61-63, 2010.

8. M. J. Campbell and T. D. V. Swinscow, Statistics at Square One, 11th ed., Wiley-Blackwell: BMJ Books, 2009.

9. T. Hastie, R. Tibshirani, and J. H. Friedman, The Elements of Statistical Learning, 2nd ed., New York: Springer, 2009.

10. SeungWon Kim, "[Anomaly Detection] RF, LightGBM, XGB, and SHAP", *Kaggle*, Доступно: https://www.kaggle.com/code/possiblemanjr/anomaly-detection-rf-lightgbm-xgb-and-shap.

СЦЕНАРІЇ ОБРОБКИ МУЛЬТИМОДАЛЬНИХ ЗАПИТІВ У МЕДИЧНІЙ ПЛАТФОРМІ

Ващевський Д.Ф.[1], Безносик О.Ю.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] vaschchevskyi.danylo@lll.kpi.ua, [2] beznosyk.oleksandr@lll.kpi.ua [0000-0003-2775-6070]

У статті досліджується застосування мультиагентного підходу для покращення обробки мультимодальних медичних запитів, зокрема, для підвищення точності медичних рішень через взаємодію спеціалізованих агентів, що аналізують різноманітні типи даних, такі як зображення та лабораторні результати. Метою дослідження є розробка ефективного методу для інтеграції даних з різних джерел з використанням мультиагентної системи. Результати демонструють значне покращення точності, проте також вказують на підвищену складність і витрати ресурсів. Практична значимість дослідження полягає у створенні методології для аналізу медичних запитів, яка здатна адаптуватись до великої кількості факторів та модальностей.

Ключові слова: NLP, LLM, reasoning, мультиагентна система, мультимодальна система.

1. ВСТУП

За останній рік стався помітний прогрес у розвитку великих лінгвістичних моделей (LLM), вони показали значні успіхи в задачах узагальнення серед широкого спектру завдань та застосувань. Такі спроможності обумовлені тренуванням великих лінгвістичних моделей на обширній вибірці масивних корпусів тексту, що покривають різноманітні галузі знань. Успіхи застосування LLM у різноманітних галузях сприяли інтересу до адаптації LLM і до медичної галузі. Найбільшими викликами для LLM у медичній сфері є малий об’єм та сильна специфічність медичних даних [1] та потреба у розвинених навичках міркування [2].

Проте, нещодавні дослідження показують, що якість машинного міркування можна суттєво покращити за допомогою мультиагентного підходу, який полягає в тому, щоб координувати дії та обмінюватися інформацією між кількома агентами для досягнення спільної мети [3]. У такій співпраці кожен агент може мати власні завдання, знання та цілі, але взаємодіє з іншими агентами, щоб узгоджувати дії, обмінюватися даними й адаптувати свої стратегії в залежності від контексту або нової інформації. Це особливо важливо в складних системах, таких як медичні платформи, де агенти можуть виконувати різні функції: оцінювати запити, залучати експертів або синтезувати рішення. Зокрема, такий підхід дозволяє залучати агентів, що спеціалізуються на аналізі різних видів даних, таких як зображення, текстові записи або лабораторні показники, що сприяє більш точній та всебічній оцінці медичних запитів і прийняттю зважених рішень на основі комплексного розгляду інформації з різних джерел.

2. LLM ДЛЯ ПРИЙНЯТТЯ МЕДИЧНИХ РІШЕНЬ

Великі лінгвістичні моделі показують багатообіцяючі результати в низці задач у медичному полі. Для покращення можливостей медичних LLM було досліджено два основні

підходи: навчання на специфічних для галузі даних [4] та застосування таких методів, як промпт-інженерія [5] та Retrieval Augmented Generation (RAG) [6], на етапі прогнозування або прийняття рішень. Хоча початкові дослідження були зосереджені на попередньому навчанні та донавчанні з медичними знаннями, зростання великих LLM загального призначення дозволило застосовувати методи, які не потребують навчання, де моделі використовують свої приховані медичні знання.

Різноманітні дослідження вивчали ефективні рамки для співпраці між кількома агентами LLM з метою підвищення їхніх можливостей. Однією з таких рамок є рольова гра [7], де кожен агент виконує конкретну роль, а завдання розбивається на під етапи і вирішується колективно. Існують також інші підходи, такі як голосування, міждисциплінарна співпраця та переговори. Такий підхід особливо корисний у мультимодальних системах, де агенти можуть спеціалізуватися на аналізі різних типів даних, таких як текст, зображення або відео. Спільна робота агентів дозволяє інтегрувати ці різноманітні джерела інформації для більш точного та комплексного вирішення задач.

3. ЗАПРОПОНОВАНА АРХІТЕКТУРА ТА НАБОРИ ДАНИХ

3.1. Архітектура

Процес обробки медичного запиту передбачає динамічне визначення складності, підбір ролей та мультиагентні дебати [8] і складається з наступних кроків:

1. Аналіз складності питання – за допомогою LLM визначається складність питання (базова чи середня);

2. Підбір експертів – спираючись на зміст питання та галузі медицини, що спеціалізуються на вирішенні презентованих проблем, за допомогою LLM-рекрутера визначається набір ролей спеціалістів, які найкраще би підійшли для надання компетентної відповіді. Також, визначається головний спеціаліст, ґрунтуючись на експертизі якого буде прийматись фінальне рішення;

3. Аналіз модальностей вхідних даних – за допомогою мультимодальної узагальнюючої LLM класифікуються додаткові файли до питання. Це можуть бути різні види медичних зображень, аудіо чи відео записи. До кожного спеціаліста асигнується певний вид даних, наприклад, ЕКГ до кардіолога, а результати аналізів до терапевта;

4. Аналіз даних – попередній аналіз файлів відповідальними агентами, комунікація та спілкування агентів для досягнення консенсусу у відповіді на питання;

5. Прийняття рішення – синтез всіх вхідних даних для надання обґрунтованої відповіді на медичне питання.

Агенти можуть мати наступні ролі:

Модератор – визначає складність медичного питання та всіляко супроводжує прийняття рішення. На фінальному етапі він отримує роль лікаря, експертиза якого є найбільш підходящою для прийняття остаточного рішення. Також, класифікує вхідні файли по типу зображень.

Рекрутер – здійснює підбір ролей експертів, що найкраще підходять для вирішення медичного питання. Крім цього, розподіляє вхідні файли в залежності від типу між ролями. В залежності від типу файлу можуть використовуватись різні моделі, зокрема, спеціально натреновані на таких типах.

Спеціаліст – дані агенти є вузькопрофільними експертами або лікарями загальної практики, яких залучає агент-рекрутер. Залежно від складності випадку, вони можуть працювати самостійно або в складі команди. Ці агенти беруть участь у колаборативному процесі прийняття рішень, вносячи свій експертний вклад для досягнення консенсусу або підготовки детальних звітів для випадків високої складності.

На початку обробки медичного запиту агентом-модератором визначається складність. Йому надається інформація про те, як слід визначати медичну складність, і доручається класифікувати запит за одним із двох рівнів:

- Базовий – задачі, що можуть бути вирішені в рамках компетенції одного експерта.
- Середній – включає численні фактори взаємодії, що вимагає колаборативного підходу в межах мультидисциплінарної команди. Такі сценарії потребують інтеграції різноманітних медичних знань та координації між фахівцями через консультації для розробки ефективних стратегій догляду.

Рекрутинг та класифікація вхідних даних. В залежності від складності запиту, агент-рекрутер підбирає ролі експертів, що найкраще підійдуть для вирішення цього питання. Далі, агентом-модератором відбувається класифікація вхідних файлів і розподіл агентом-рекрутером цих файлів між експертами таким чином, щоб експертиза ролі найкраще відповідала компетентному аналізу певного типу даних. В залежності від типу файлу, для агента може підбиратись модель, яка спеціально натренована на такому виді даних.

Модель колаборації (режим роботи):

- Базова – до LLM моделі, роль та тип якої визначив агент-рекрутер, застосовується few-shot промптинг для отримання відповіді (промптинг з кількома прикладами відповідей).
- Мультиагентна – медична команда, що складається з агентів, ролі яких визначив агент-рекрутер, аналізують відведені їм типи файлів та починають процес ітеративної дискусії, для того щоб дійти згоди за максимум N раундів. У кожному $n \in N$ раунді агенти A_i, $i \in 1, \dots, K$ висловлюють бажання брати участь у дискусії та називають агентів, з якими вони бажають прокомунікувати. Система забезпечує обмін повідомленнями протягом M кроків. Якщо консенсусу не досягнуто, і агенти погоджуються продовжити, розпочинається новий раунд з доступом до попередніх обговорень.

Прийняття рішення. При базовій складності запиту, відповідь агента-спеціаліста і є відповіддю на медичне питання. При середній складності агент-модератор, експертиза якого є найбільш підходящою для прийняття остаточного рішення, ознайомлюється з історією дискусії та приймає остаточне рішення, чи формує остаточну відповідь.

3.2. Набори даних

MedQA. Датасет MedQA складається з питань з професійних медичних екзаменів, що проводяться у США, Китаї та Тайвані [9]. Записи в цьому наборі даних мають формат текстових питань з вибором із п'яти варіантів відповіді.

LITFL_dataset. Цей датасет був створений в рамках цього дослідження. Він ґрунтується на наборі кейсів з бібліотеки LITFL (Life in the Fast Lane), антології та збірки матеріалів з навчання екстреної медицини та інтенсивної терапії [10]. Клінічні кейси на LITFL охоплюють різноманітні медичні ситуації, від невідкладних станів, таких як аритмії та токсикологічні випадки, до складних респіраторних і неврологічних патологій. Кожен випадок включає історію хвороби, симптоматику, діагностичні зображення, результати аналізів та покрокове клінічне міркування. Це допомагає удосконалювати навички діагностики та вибору лікування, орієнтуючись на найкращі практики медицини.

Приклад медичного кейсу та причетної картинки (рис. 1): "*A 32 year-old woman is referred to the ICU post caesarean section. She had an uneventful elective caesarian section 10 hours previously. She has been referred because in the last 2 hours she has developed vaginal bleeding and oozing from her epidural site. Her vital signs are unremarkable except for a small oxygen requirement. The following coagulation results are obtained*:".

Приклад питання: "*What abnormality would you expect to see on a blood film?*".

Приклад відповіді: “*In severe DIC there are fragmented red blood cells (schistocytes)*”.

LITFL_options є модифікацією LITFL_dataset, що передбачає наявність питань з кількома варіантами відповідей. Ці питання та відповіді були синтетично створені на базі старих питань з відкритою відповіддю та відповідей на них, залишаючи незмінними медичний кейс та інші причетні файли. Приклад нового питання: “*What type of cellular change would typically indicate significant coagulation issues following an event like a caesarean section?*”. Приклади варіантів відповідей: *{*“*A*”*:* “*Increased white blood cell count.*”, “*B*”*:* “*High levels of platelets.*”, “*C*”*:* “*Normal red blood cell morphology.*”, “*D*”*:* “*Presence of immature granulocytes.*”, “*E*”*:* “*Presence of fragmented red blood cells (schistocytes).*”*}*.

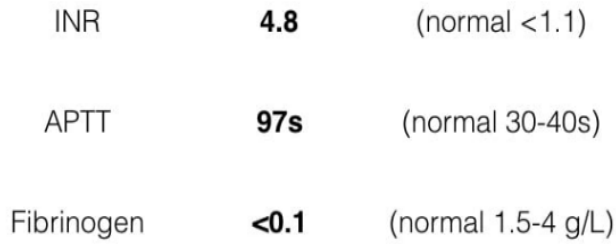


Рисунок 1. Приклад картинки у медичному кейсі

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

В результаті проведеної роботи буди проведені тести архітектури на трьох наборах даних. MedQA та LITFL_options передбачали відповіді на питання з кількома варіантами відповідей, а LITFL_dataset передбачав питання з відкритою відповіддю. У всіх випадках кількість тестової вибірки складала 50 випадково обраних питань. У дослідженні використовувались моделі сімейства OpenAI: gpt-3.5, gpt-4o-mini для переважної роботи з текстом, gpt-4o для аналізу зображень. Режими роботи ділились на базовий (один агент), мультиагентний (багато агентів) та адаптивний (визначення складності питання та застосування базового або мультиагентного режиму в залежності від складності). Результати роботи системи на вищенаведених наборах даних представлені у таблиці 1 та таблиці 2.

Таблиця 1. Результати роботи системи при відповідях на питання з декількома варіантами відповідей

Режим	Набір даних	Точність	Середня кількість токенів	Середній час обробки запиту (с)
базовий	MedQA	0.7	579.68	5.1
мультиагентний	MedQA	0.66	55127	72
адаптивний	MedQA	0.72	44475.11	63
базовий	LITFL_options	0.7	699	14.3
мультиагентний	LITFL_options	0.92	119020.54	68.5
адаптивний	LITFL_options	0.96	124195.34	72.4

З таблиці 1 можна зробити висновки, що завдяки адаптивному підходу до прийняття рішення можна отримати суттєве покращення точності, виділяючи лише необхідні ресурси. Перевагою мультиагентного підходу є здатність правильно відповідати на ширші та складніші питання, але недоліком є менша точність відповідей на простіші питання та великий час обробки відповіді та відносно велика кількість оброблених токенів, що каже про більшу вартість використання мультиагентного підходу.

Оцінювання правильності відповідей на відкриті питання є складним завданням, яке потребує врахування як змістової точності, так і стилю відповіді. Метрики BLEU та ROUGE,

використовують порівняння з еталонними відповідями на основі спільних n-грам. BLEU фокусується на точності збігу з еталонним текстом, тоді як ROUGE-1 і ROUGE-L аналізують спільні слова та послідовності, орієнтуючись на змістовий збіг. Проте, для відкритих запитань ці методи можуть не врахувати семантичної глибини чи варіативності. Особливо гостро ця проблема виражається у медичній сфері, де терміни в різних методологіях можуть називатись по-різному, і для того щоб вирішити цю проблему будують спеціальні фреймворки перекладу медичних термінів [11]. Також, дрібниці, що є життєво важливими у медичних текстах, також не можуть бути вловлені класичними метриками схожості тексту.

Найкращим рішенням для оцінки якості відповідей на питання з відкритою відповіддю є залучення людей-спеціалістів, але в рамках обмежених ресурсів дослідження можна використати LLM в якості судді. LLM-as-a-judge дозволяють оцінювати відповіді на основі смислової адекватності та контекстуальної відповідності, що робить оцінювання гнучкішим і точнішим [12]. Хоч LLM-as-a-judge не досягає рівня точності експертів і не здатен завжди розпізнати тонкі помилки, він може ефективно оцінювати загальну повноту відповіді. Така система аналізує відповідність змісту запиту і виявляє основні елементи відповіді. Це дозволяє створювати автоматизовані оцінки на основі релевантності та семантичного узгодження, зберігаючи при цьому широту аналізу, що може бути корисним для масового оцінювання відкритих запитань [13].

Таблиця 2. Результати роботи системи при відповідях на питання з відкритою відповіддю

Режим	Набір даних	BLEU	ROUGE-1	ROUGE-L	Оцінка	Середня кількість токенів	Середній час обробки запиту, с
базовий	LITFL_dataset	3.8	0.33	0.19	6.8	3843	3.85
мультиагентний	LITFL_dataset	2.84	0.29	0.163	7	268358	122
адаптивний	LITFL_dataset	2.91	0.29	0.161	7.1	168314	115

У таблиці 2 представлені результати оцінювання LLM-суддею відповідей системи на питання з відкритою відповіддю. Суддя був інструктований на перевірку повноти та правильності відповіді. Результати є більш суперечливими, ніш у попередньому випадку, модель що використовувалась – gpt-4o. При мультиагентному підході, оцінка є приблизно такою ж самою, як при використанні одного агента, але схожість відповідей систем з одним агентом більша. Це може пояснюватись тим, що в результаті дискусії мультиагентна система може відійти від медичних термінів, що використовуються у еталонній відповіді. З тієї ж причини, стилістика відповідей може відрізнятись, що впливає на класичні метрики схожості тексту.

5. ВИСНОВКИ

Проведене дослідження підтверджує перспективність використання мультиагентного підходу для обробки мультимодальних запитів. Завдяки взаємодії спеціалізованих агентів, кожен з яких аналізує конкретний тип даних (наприклад, зображення, текстові записи або лабораторні показники), таким чином, контекст наповнюється більш різноманітними даними, сприяючи зменшенню галюцинацій та створюючи можливості для глибшого і всебічного аналізу при формуванні відповіді.

Разом із цим, застосування мультиагентного підходу супроводжується підвищенням витратних ресурсів, зокрема часу обробки та обчислювальної складності, що збільшує вартість підтримки таких рішень. Крім того, у випадках з питаннями з відкритими відповідями виникають певні складнощі для оцінювання мультиагентної системи внаслідок специфічності

медичної інформації та стилістичної різноманітності еталонних відповідей, що відображається на класичних метриках якості схожості текстів.

Отже, мультиагентні підходи демонструють не тільки значний потенціал у підвищенні точності та комплексності аналізу медичних запитів, а й відкривають простір для покращення його якості завдяки промпт-інженерії та великим за обсягом текстовим корпусам, на яких натреновані сучасні великі лінгвістичні моделі. Проте, їх використання потребує подальших досліджень для оптимізації витратних ресурсів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Thirunavukarasu, A. J., Ting, D. S. J., Elangovan, K., Gutierrez, L., Tan, T. F., & Ting, D. S. W. (2023). Large language models in medicine. *Nature medicine*, *29*(8), 1930–1940. https://doi.org/10.1038/s41591-023-02448-8
2. Liévin, V., Hother, C. E., Motzfeldt, A. G., & Winther, O. (2023). Can large language models reason about medical questions? *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2207.08143
3. Tang, X., Zou, A., Zhang, Z., Li, Z., Zhao, Y., Zhang, X., … Gerstein, M. (2024). MedAgents: Large Language Models as Collaborators for Zero-shot Medical Reasoning. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2311.10537
4. Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., … Poon, H. (2021). Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Transactions on Computing for Healthcare*, *3*(1), 1–23. doi:10.1145/3458754
5. Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., … Natarajan, V. (2022). Large Language Models Encode Clinical Knowledge. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2212.13138
6. Zakka, C., Shad, R., Chaurasia, A., Dalal, A. R., Kim, J. L., Moor, M., Fong, R., Phillips, C., Alexander, K., Ashley, E., Boyd, J., Boyd, K., Hirsch, K., Langlotz, C., Lee, R., Melia, J., Nelson, J., Sallam, K., Tullis, S., Vogelsong, M. A., … Hiesinger, W. (2024). Almanac - Retrieval-Augmented Language Models for Clinical Medicine. *NEJM AI*, *1*(2), 10.1056/aioa2300068. https://doi.org/10.1056/aioa2300068
7. Chen, J., Wang, X., Xu, R., Yuan, S., Zhang, Y., Shi, W., … Xiao, Y. (2024). From Persona to Personalization: A Survey on Role-Playing Language Agents. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2404.18231
8. Kim, Y., Park, C., Jeong, H., Chan, Y. S., Xu, X., McDuff, D., … Park, H. W. (2024). MDAgents: An Adaptive Collaboration of LLMs for Medical Decision-Making. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2404.15155
9. Jin, D., Pan, E., Oufattole, N., Weng, W.-H., Fang, H., & Szolovits, P. (2020). What Disease does this Patient Have? A Large-scale Open Domain Question Answering Dataset from Medical Exams. *arXiv Preprint arXiv:2009. 13081*.
10. Cadogan, M. (2023, August 17). *Clinical cases • LITFL Medical Blog • Case collection*. Life in the Fast Lane • LITFL. https://litfl.com/clinical-cases/ Дата звернення: 01.11.2024.
11. Yuan, Z., Zhao, Z., Sun, H., Li, J., Wang, F., & Yu, S. (2021). CODER: Knowledge infused cross-lingual medical term embedding for term normalization. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2011.02947
12. Son, G., Ko, H., Lee, H., Kim, Y., & Hong, S. (2024). LLM-as-a-Judge & Reward Model: What They Can and Cannot Do. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2409.11239
13. Li, J., Wang, S., Zhang, M., Li, W., Lai, Y., Kang, X., … Liu, Y. (2024). Agent Hospital: A Simulacrum of Hospital with Evolvable Medical Agents. *arXiv [Cs.AI]*. Retrieved from http://arxiv.org/abs/2405.02957

АВТОМАТИЗАЦІЯ USABILITY ТЕСТУВАННЯ ВЕБ-САЙТІВ

Дідок Т.А.[1], Кисельов Г.Д.[2]

[1,2]Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] didok.taras@lll.kpi.ua, [2] g.kyselov@gmail.com [0000-0003-2682-3593]

У даному дослідженні розглядається процес тестування зручності користування веб-сайтами, його проблеми і особливості. У роботі описано класичні методи проведення цього процесу, а також способи та підходи до його автоматизації, включаючи їх переваги та недоліки. Як результат, реалізована програма, яка дозволяє проводити тестування usability автоматично, без залучення кінцевих користувачів. Практична значимість даного дослідження полягає у пришвидшенні usability тестування та зниженні витрат на тестування.

Ключові слова: usability, зручність користування програмним забезпеченням, веб-сайт, автоматизація тестування, доступність веб-додатків, продуктивність веб-інтерфейсів.

1. ВСТУП

У сучасних умовах швидкого розвитку інформаційних технологій та постійно зростаючої конкуренції на ринку веб-застосунків, забезпечення якісної та зручної взаємодії користувача з продуктом є важливим фактором для успіху проектів. Проблеми з використанням веб-сайту можуть викликати незадоволення у його відвідувачів і призводити до втрати клієнтів. У зв’язку з цим, usability тестування, яке спрямоване на оцінювання зручності та ефективності користування програмними застосунками, стає одним з ключових етапів у розробці веб-сайтів

Тим не менш, класичні методи проведення usability тестування мають високу вартість і вимагають значних затрат часу, а також залучення великої кількості людських ресурсів, що робить їх складними у проведенні для багатьох проектів.

Основна мета даної роботи – висвітлити способи підвищення ефективності процесу usability тестування за рахунок його автоматизації, що дозволить компаніям отримувати більш об’єктивні результати при менших затратах часу і ресурсів, і відповідно прискорити час виходу застосунків на ринок та покращити їх якість.

2. КЛАСИЧНІ МЕТОДИ USABILITY ТЕСТУВАННЯ

Usability тестування — це процес оцінювання ефективності та зручності взаємодії з програмним продуктом з точки зору користувача. Його основна ціль полягає в тому, щоб визначити, наскільки зрозумілим та зручним для взаємодії є інтерфейс користувача (UI), та виявити можливі проблеми, що перешкоджають роботі з системою.

Відповідно до традиційного визначення usability розглядається на основі трьох складових:

1) **Ефективність** – міра того, наскільки успішно (з точки зору зусиль та кількості помилок) відвідувачі можуть виконувати певні задачі при роботі з веб-сайтом;
2) **Продуктивність** – міра того, скільки часу користувачу необхідно витрачати для досягнення поставленої цілі;

3) **Задоволеність** – складова, що розглядає суб'єктивні враження користувача від взаємодії з системою [1].

Залежно від вимог та цілей застосунку, а також наявних ресурсів, для тестування зручності користування можуть застосовуватися різні підходи. Найпоширенішими класичними методами є наступні:

- Лабораторні дослідження — один з основних підходів до оцінки usability відповідно до традиційного визначення. При його використанні формується контрольна група учасників, яких запрошують у заздалегідь підготоване контрольоване середовище, де вони виконують певні завдання під спостереженням експертів;
- Дистанційне тестування та проведення опитувань — альтернатива попередньому методу, при якій учасники тестування виконують завдання на власних пристроях у звичному для них середовищі. Їх дії фіксуються за допомогою відеокамери та програм для запису екрану, після чого проводяться опитування щодо задоволеності роботою з продуктом;
- A/B тестування — підхід, ідея якого полягає у порівнянні кількох варіантів користувацького інтерфейсу та оцінки того, який з них більше подобається користувачам. При використанні цього методу групу учасників розділяють на кілька підгруп і дають їм взаємодіяти з різними версіями застосунку, після чого аналізують враження тестувальників та вирішують, який з розглянутих варіантів є кращим [2].

Як можна побачити, традиційне визначення usability та класичні підходи до його тестування приділяють значну увагу особистим враженням користувачів та їх індивідуальним вподобанням. У зв'язку з цим отримані результати тестування нерідко бувають надто суб'єктивними, а також не мають чітких критеріїв оцінювання. Крім того, проведення класичного тестування зручності користування зазвичай є трудомістким, дорогим та вимагає залучення значної кількості учасників. Для вирішення цих проблем можуть розглядатися інші напрями оцінки usability, зокрема підходи до автоматизації цього процесу.

3. ПІДХОДИ ДО АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ USABILITY

Методи автоматизації тестування зручності користування можна умовно розділити на дві категорії, залежно від способів проведення оцінки та етапів життєвого циклу продукту, на яких ці методи застосовуються.

3.1. Автоматизована аналітика поведінки користувачів

Ідея автоматизованої аналітики полягає в тому, щоб автоматично збирати інформацію про те, як саме реальні користувачі взаємодіють з веб-сайтом. На основі зібраних даних формуються певні статистичні відомості, аналіз яких дозволяє оцінити зручність використання програмного продукту.

Для збору даних про поведінку користувачів зазвичай використовуються JavaScript скрипти, які включаються у html структуру веб-сайту. Ці скрипти фіксують певні події на сторінках, зокрема натискання на елементи інтерфейсу, переміщення до різних частин сторінки, перехід по посиланням тощо. Інформація про зафіксовані події відправляється на сервер, де вони обробляються за допомогою алгоритмів кластеризації для отримання більш об'єктивної статистичної інформації з багатьох користувацьких сесій [3].

Одним з найпопулярніших методів для аналітики поведінки користувачів є теплові карти. При його застосуванні зібрані дані про взаємодію з веб-сайтом групуються та накладаються поверх зображення веб сторінки, виділяючи різними кольорами ділянки, що привертають більше чи менше уваги користувачів (Рис. 1). На теплових картах можна побачити певні проблеми зі зручністю користування інтерфейсом — наприклад про це може

свідчити велика кількість взаємодій з елементами веб-сторінки, що не є інтерактивними, такими як секції тексту, зображення без посилань чи декоративні об’єкти.

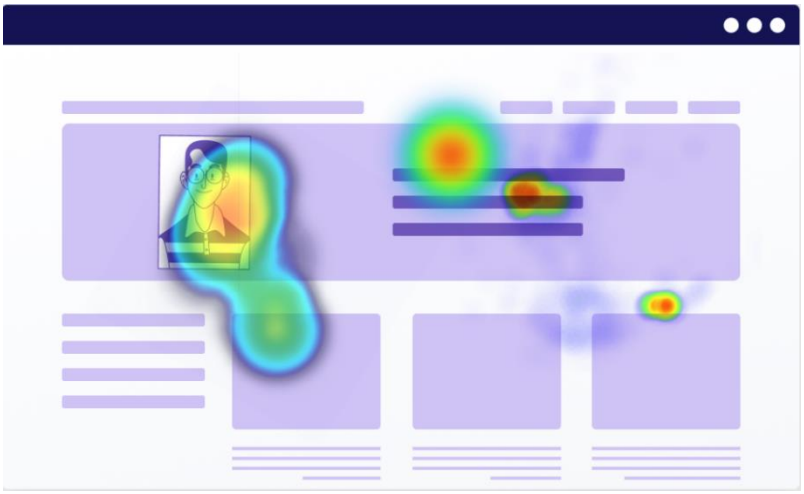


Рисунок 1. Приклад теплової карти веб-сайту [4]

Ще одним популярним методом є карти навігації, які показують, яким чином користувачі пересуваються сторінками веб-сайту. Це дає можливість виявляти та оптимізовувати неробочі чи переускладнені шляхи навігації, що сприяє покращенню зручності користування.

Ці та інші методи автоматизованої аналітики дозволяють отримувати корисну та достатньо об’єктивну інформацію про поведінку реальних користувачів. При цьому затрати на проведення таких досліджень є нижчими, ніж для класичних методів usability тестування, і вони не вимагають залучення окремої групи тестувальників.

Серед недоліків таких методів можна виділити те, що вони потребують значної кількості даних про взаємодію реальних користувачів з веб-сайтом, що унеможливлює їх застосування на ранніх етапах розробки програмних застосунків — а це може призводити до того, що певні проблеми зі зручністю будуть знаходитись надто пізно і вимагатимуть значних змін у коді для їх вирішення. Також варто відзначити, що при зборі користувацьких даних необхідно дотримуватися багатьох міжнародних нормативів та актів, які регулюють умови обробки персональної інформації.

3.2. Автоматизована оцінка метрик зручності

Підхід автоматизованої оцінки метрик зручності ще більше відходить від взаємодії з користувачами, і натомість зосереджується на дослідженні об’єктивних критеріїв, які так чи інакше впливають на usability веб-сайту. Цей напрям більш схожий на традиційне функціональне тестування, і дозволяє отримати об’єктивні, відтворювані результати за чітко визначеними вимогами. Зокрема, часто розглядаються метрики в таких категоріях:

1) **Доступність** – в першу чергу визначає, наскільки гарним є досвід використання застосунку для людей з обмеженими можливостями, проте слідування рекомендаціям щодо доступності (наприклад, регуляція контрастності елементів чи збільшення розміру шрифтів) також сприяє покращенню зручності користування загалом. Для метрик доступності існує чіткий стандарт WCAG, в якому докладно описані вимоги до веб-інтерфейсів [5];
2) **Сприйнята продуктивність** – визначає продуктивність роботи веб-інтерфейсу, тобто швидкість завантаження елементів сторінки та ефективність поведінки об’єктів на сторінці при взаємодії з ними. Продуктивність є одним з найважливіших

критеріїв для багатьох користувачів і напряму впливає на їх враження від взаємодії з веб-сайтом. Для автоматизованої оцінки цього критерію можуть використовуватися різні метрики, наприклад First Contentful Paint (FCP) чи Time To Interactive (TTI) [6];

3) **SEO та передові методики** — не оцінюють зручність користування напряму, проте також покращують загальний досвід від роботи з веб-сайтом. Наприклад, до цієї категорії можуть відноситися перевірки використання захищених протоколів чи відсутність неробочих посилань на сторінці;
4) **Крос-платформна оптимізація** — перевіряє, чи відрізняється досвід при роботі з сайтом з різних платформ. При автоматизованих перевірках це може оцінюватися за рахунок емулювання розміру екрану та мережевих налаштувань мобільних пристроїв.

Можна побачити, що при використанні підходу автоматизованої оцінки метрик не розглядається інформація про враження реальних користувачів від роботи з веб-сайтами. З іншого боку, його застосування дає інші корисні, і що важливо — більш об'єктивні результати про зручність застосунків загалом. Також ці перевірки відбуваються значно швидше, ніж інші розглянуті методи, і можуть застосовуватися навіть на ранніх етапах розробки, що дозволяє напряму інтегрувати їх у життєвий цикл програмного продукту.

4. ДОСЛІДЖЕННЯ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

Для дослідження перспектив автоматизації usability тестування було використано підхід автоматизованої оцінки метрик зручності, зокрема дослідження сприйнятої продуктивності та доступності. Процес тестування відбувався в наступні етапи:

1) Запуск драйвера веб-браузеру (Chrome) та навігація за URL адресою сайту;
2) Запуск скриптів для збору інформації про DOM-структуру сторінки та швидкість завантаження;
3) Аналіз зібраних даних відповідно до метрик оцінювання;
4) Підготовка звіту про результати тестування.

В якості сайту для дослідження було обрано веб-сторінку Національної Бібліотеки України імені В. І. Вернадського. Результати проведення тестування наведені на рисунку 3.

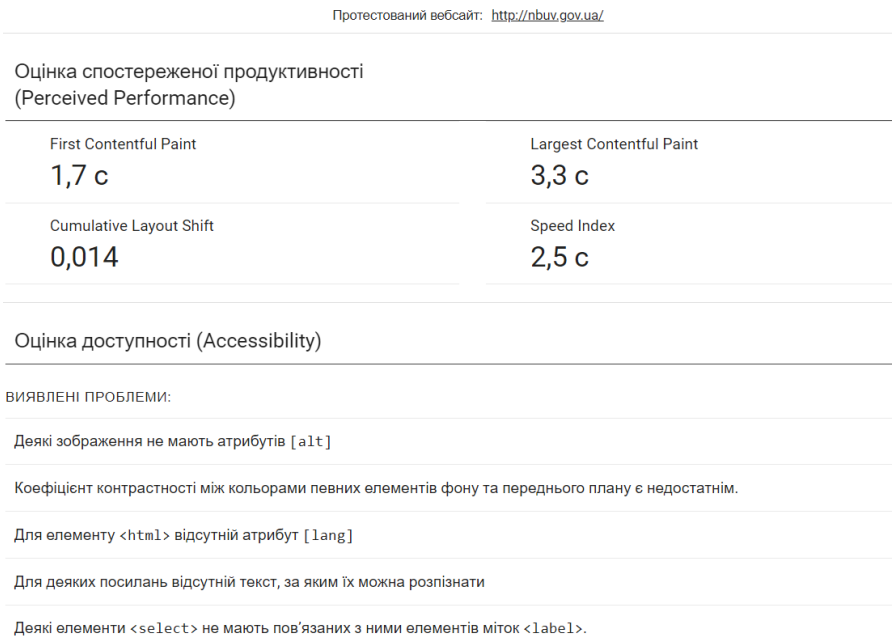


Рисунок 2. Результати проведення тестування

Як можна побачити, автоматизоване тестування дозволило знайти на веб-сторінці певні проблеми з доступністю. Їх вирішення дасть змогу покращити досвід взаємодії з веб-сайтом для людей з обмеженими можливостями та загалом підвищить рівень зручності користування.

5. ВИСНОВКИ

Проведене дослідження продемонструвало, що класичні методи тестування зручності користування, хоч і є важливими та корисними для сучасних веб-застосунків, мають ряд обмежень, зокрема високу вартість, тривалість та суб'єктивність результатів. Впровадження автоматизованих підходів тестування, зокрема аналітики поведінки користувачів та оцінки метрик зручності, дозволяє не лише виявляти певні проблеми зі зручністю користування швидше та з меншими затратами ресурсів, але й інтегрувати відповідні перевірки на більш ранніх етапах життєвого циклу програмного продукту.

Таким чином, автоматизація usability тестування створює нові можливості для розробників та організацій, спрощуючи процеси оптимізації веб-інтерфейсів та роблячи їх більш ефективними. Подальший розвиток у цьому напрямі дозволить забезпечити більш високі стандарти якості та доступності веб-сайтів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Jokela, T., Iivari. N., Matero, J., Karukka., M. (2003). The standard of user-centered design and the standard definition of usability: analyzing ISO 13407 against ISO 9241-11. Proceedings of the Latin American Conference on Human-Computer Interaction, Rio de Janeiro, Brazil, 53-60.

2. Rubin, J., Chisnell, D. (2008). Handbook of Usability Testing, 2nd Edition: How to Plan, Design, and Conduct Effective Tests. Wiley Publishing, Inc., 2008, 386 p.

3. Shahida, N., Grundy, J., Vasa, R. (2016). Reporting Usability Defects: A Systematic Literature Review. IEEE Transactions on Software Engineering.

4. How Click Maps Take Web Analytics to Next Level. [Електронний ресурс] Режим доступу: https://glassbox.com/blog/click-maps/

5. Web Content Accessibility Guidelines (WCAG) 2.1. [Електронний ресурс] Режим доступу: https://www.w3.org/TR/WCAG21/

6. Namoun, A., Tufail, A. (2021) Review of Automated Website Usability Evaluation Tools: Research Issues and Challenges. Faculty of Computer and Information Systems, Islamic University in Madinah, Saudi Arabia.

ПРОГНОЗУВАННЯ СЕРЙОЗНОСТІ ПОМИЛОК В ПРОГРАМАХ ЗА ДОПОМОГОЮ ВПРОВАДЖЕННЯ МОДЕЛІ ГЛИБОКОГО НАВЧАННЯ (LSTM)

Коваль П.С.[1] , Кисельов Г.Д.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] koval.pavlo@lll.kpi.ua, [2] g.kyselov@gmail.com [0000-0003-2682-3593]

Дослідження присвячено автоматизації процесу класифікації серйозності програмних помилок з використанням мережі довгострокової короткочасної пам'яті (LSTM). Запропонована модель аналізує текстові звіти про помилки та прогнозує їх серйозність, що дозволяє підвищити ефективність тестування. Модель забезпечує точне визначення рівня серйозності помилок, що сприяє оптимізації пріоритетів для розробників. Наукова новизна полягає у застосуванні LSTM для прогнозування серйозності помилок, використовуючи дані звітів про помилки. Практична цінність – у створенні інструменту, що скорочує час на класифікацію рівня серйозності помилок.

Ключові слова: прогнозування серйозності помилок, автоматизація тестування, LSTM, глибоке навчання, класифікація помилок, машинне навчання.

1. ВСТУП

Сучасна розробка програмного забезпечення (ПЗ) потребує ефективного управління процесами тестування, оскільки якість та надійність ПЗ є важливими критеріями успішності проекту. Зі збільшенням складності програмних продуктів та ростом обсягу звітів про помилки стає важливим не лише швидке виявлення дефектів, але й оцінка їх серйозності для оптимізації розподілу ресурсів на їх усунення.

Глибоке навчання, зокрема нейронні мережі довгострокової короткочасної пам'яті (LSTM), відкриває нові можливості в автоматизації класифікації серйозності помилок. LSTM дозволяє працювати з текстовими даними, виявляючи залежності, які ефективно підтримують прогнозування серйозності помилок на основі описів у звітах.

Актуальність цієї роботи полягає в розробці моделі на основі LSTM для автоматизації класифікації серйозності помилок за текстовими звітами. Це допоможе знизити навантаження на команди тестування, зменшити вплив людського фактору та підвищити точність класифікації.

Метою роботи є створення ефективної моделі глибокого навчання для автоматичної класифікації серйозності програмних помилок, що дозволить розробникам швидше пріоритизувати виправлення багів, скорочуючи час і підвищуючи загальну якість продукту.

2. ОСНОВНИЙ ЗМІСТ

Сучасна розробка програмного забезпечення (ПЗ) потребує ефективного управління процесами тестування, оскільки якість та надійність ПЗ є важливими критеріями успішності проекту. Використання методів глибокого навчання, зокрема моделей довгострокової

короткочасної пам’яті (LSTM), дозволяє автоматизувати процес оцінки серйозності помилок та досягти високої точності їх класифікації.

LSTM (Long Short-Term Memory) — це спеціальний вид рекурентної нейронної мережі (RNN), розроблений для подолання проблеми «зникання градієнта», що є типовою проблемою для стандартних RNN при роботі з довгими послідовностями даних.

Архітектуру LSTM можна уявити як послідовність повторюваних «блоків» або «комірок», кожен з яких містить набір пов’язаних між собою елементів, що дозволяють моделі зберігати та обробляти важливу інформацію впродовж усієї послідовності даних.

На кожному кроці часу модель LSTM отримує вхідний вектор x_t , який представляє поточний елемент у послідовності, наприклад, маркер або спостереження. Окрім цього, LSTM зберігає прихований стан h_t, що представляє поточну «пам’ять» мережі, який ініціалізується нульовим вектором на початку послідовності. Додатково, LSTM підтримує стан комірки c_t , що відповідає за зберігання довгострокової інформації. Як і прихований стан, стан комірки також ініціалізується нульовим вектором.

Керування потоком інформації через мережу здійснюється за допомогою трьох типів «шлюзів», які забезпечують гнучке збереження або очищення інформації. Перший — це ворота забування, які отримують попередній прихований стан h_{t-1} і поточний вхід x_t, генеруючи вектор значень від 0 до 1. Цей вектор визначає, яку частину попереднього стану комірки слід зберегти, а яку «забути». Це дає можливість LSTM вибірково очищати інформацію з попередніх кроків.

Другий тип – вхідний шлюз, який також приймає попередній прихований стан h_{t-1} та поточний вхід x_t, генеруючи вектор значень від 0 до 1. Цей вектор визначає, яку частину поточного вхідного сигналу необхідно додати до стану комірки. Завдяки цьому шлюзу, LSTM може гнучко оновлювати стан комірки, «включаючи» нову інформацію або «ігноруючи» її за необхідності.

Третій – вихідний шлюз, який обробляє попередній прихований стан h_{t-1} , поточний вхід x_t та стан комірки c_t, також формуючи вектор значень від 0 до 1. Цей вектор визначає, яку частину поточного стану комірки слід використовувати для формування нового прихованого стану h_t, що забезпечує гнучкий вибір інформації, яка буде врахована у вихідних даних.

На кожному часовому кроці LSTM виводить вектор y_t, який відображає прогноз або кодування поточного введення. Поєднання стану комірки, прихованого стану та трьох типів шлюзів дозволяє LSTM вибірково зберігати чи «стирати» інформацію, що робить цю архітектуру оптимальною для завдань, які вимагають моделювання довготривалих залежностей або послідовностей даних. Описану архітектуру представлено на рисунку (рис. 1).

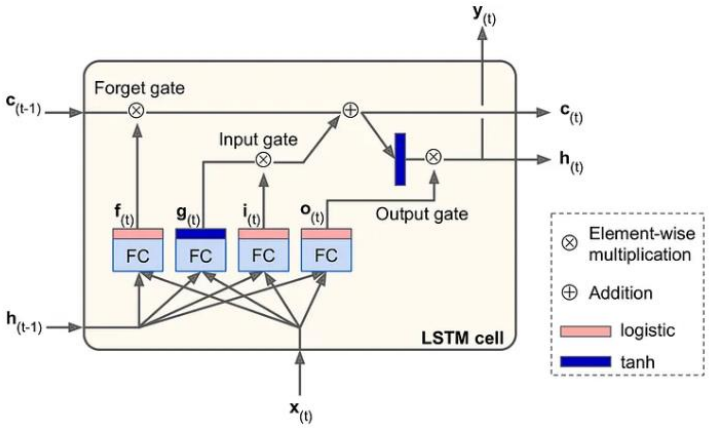


Рисунок 1. Архітектура LSTM

Основні переваги LSTM полягають у здатності зберігати інформацію на великих послідовностях, працювати з різними типами даних, такими як тимчасові ряди та текстові звіти, та уникати проблеми зникання градієнта завдяки структурі воріт у блоках. Недоліками LSTM є висока обчислювальна складність, яка потребує значних ресурсів для навчання, а також складність налаштування гіперпараметрів та інтерпретації результатів.

Для реалізації LSTM в середовищі Google Colab за допомогою Python використовуються такі бібліотеки, як TensorFlow та Keras, які спрощують роботу з нейронними мережами. Google Colab забезпечує доступ до обчислювальних потужностей, включаючи GPU, що значно прискорює процес навчання моделі.

Для дослідження з прогнозування серйозності помилок за допомогою моделі LSTM було вирішено використати звіти про помилки платформи Eclipse як основний набір даних. Рівні серйозності помилок у данному наборі даних включають сім основних категорій: Normal, Enhancement, Major, Minor, Critical, Blocker та Trivial.

Рівень Normal включає помилки з мінімальним впливом на функціональність програми, де виправлення не є терміновим. На відміну від Minor, Normal помилки можуть мати трохи більший функціональний вплив, хоча вони не є критичними. Прикладом можуть бути некоректне відображення тексту в інтерфейсі або дрібні помилки в логах, які не впливають на роботу.

Minor — це незначні помилки, здебільшого косметичні, які не зачіпають основну функціональність програми. Цей рівень більше стосується візуальних проблем або дрібних естетичних недоліків, на відміну від Normal. Сюди можна віднести неправильний колір кнопки, нерівне розташування елементів на сторінці або незначні затримки в анімації.

Enhancement — це пропозиції на покращення функціональності, а не власне баги. Вони відображають побажання щодо нових функцій або змін, які покращують роботу програми, але не є критичними. Наприклад, додавання автозаповнення у форму, покращення пошукових можливостей або нового інструменту для фільтрації даних.

Major помилки є серйозними, вони впливають на важливі функції, але мають обхідні шляхи для продовження роботи. На відміну від Blocker чи Critical, ці помилки не блокують роботу повністю, хоча значно впливають на користувача. Прикладом можуть бути збій під час великого навантаження або некоректна робота окремого модуля, який можна оминути.

Critical включає помилки, які серйозно порушують важливу функціональність і потребують швидкого виправлення. Ці помилки більш серйозні, ніж Major, оскільки можуть вплинути на дані або стабільність програми. Прикладами є втрати даних при збереженні або проблеми з базою даних, що призводять до збоїв.

Blocker — найвищий рівень серйозності, де помилка блокує основні функції або робить програму повністю непрацездатною. Це повністю зупиняє роботу програми, на відміну від Critical, де робота ще можлива. Прикладами є збій програми під час запуску або помилка, що унеможливлює доступ до головних функцій.

Trivial включає несуттєві баги, які практично не впливають на роботу програми. Цей рівень має найменший вплив серед усіх рівнів серйозності, і помилки майже непомітні для користувача. Сюди входять орфографічні помилки, неточності в описах або несуттєві відхилення у відображенні дрібних елементів.

Реалізація моделі включає підготовку даних та побудову нейронної мережі. Для початку необхідно імпортувати необхідні бібліотеки: numpy, pandas, tensorflow, sklearn. Потім текстові звіти про помилки готуються для навчання шляхом очищення даних та векторизації. Приклад побудови моделі LSTM на Python може виглядати так: спочатку створюється послідовна модель Sequential(), де додається шар вбудовування (Embedding) для кодування вхідних даних, далі два шари LSTM з різною кількістю блоків, та завершальний

шар для класифікації на сім рівнів серйозності помилок. Після цього модель компілюється з використанням оптимізатора adam та функції втрат categorical_crossentropy, після чого запускається навчання моделі на підготовлених даних з використанням функції fit.

Оцінка моделі глибокого навчання, такої як LSTM, є важливим етапом, що дозволяє визначити її продуктивність та якість прогнозів. Основні метрики, що використовуються для оцінки класифікаційних моделей, включають Precision, Recall, F1 Score, ROC-криву та AUC.

Precision (точність) визначає частку правильних позитивних передбачень серед усіх передбачень позитивного класу, показуючи, наскільки модель точна у своїх прогнозах для цього класу. Recall (повнота) показує, яку частку всіх реальних позитивних прикладів модель правильно передбачила, що є важливим, коли необхідно мінімізувати пропущені позитивні передбачення. F1 Score є зваженим середнім точності та повноти, забезпечуючи збалансовану оцінку моделі, особливо корисну, коли важливо мати обидві метрики на високому рівні.

$$\text{Precision} = \frac{TP}{TP+FP}, \quad (1)$$

де TP – кількість істинно позитивних передбачень, а FP — кількість хибно позитивних.

$$\text{Recall} = \frac{TP}{TP+FN}, \quad (2)$$

де FN – хибно негативні передбачення.

$$\text{F1 Score} = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (3)$$

ROC-крива (Receiver Operating Characteristic) є графічним відображенням якості моделі для розрізнення позитивних та негативних класів при різних порогах. Вона будується шляхом нанесення True Positive Rate (TPR) (або Recall) проти False Positive Rate (FPR), що дозволяє побачити, наскільки добре модель здатна відрізняти класи. AUC (Area Under the Curve) відображає площу під ROC-кривою — чим більша площа, тим краще модель відрізняє позитивні та негативні класи, де значення, близьке до 1, вказує на високу якість.

3. РЕЗУЛЬТАТИ РОБОТИ

Для прогнозування серйозності програмних помилок була використана модель на основі LSTM із наступними характеристиками. В архітектурі моделі використовується Embedding шар для зменшення розмірності тексту, що дозволяє кожному слову представлятися у вигляді векторного простору з розмірністю 50. Після нього розташований один LSTM-шар з 64 нейронами, який допомагає запам'ятовувати послідовності та вловлювати контекст у тексті. Завершує архітектуру вихідний класифікаційний шар із 7 нейронами (по одному для кожного класу серйозності помилок) з активацією softmax, що видає ймовірності для кожного класу. Під час навчання моделі використовувався оптимізатор Adam, який забезпечує адаптивне коригування швидкості навчання, а також функція втрат Sparse Categorical Crossentropy, що підходить для багатокласової класифікації. Модель навчалася протягом 10 епох із розміром батчу 32, що дозволяє ефективно працювати з обмеженими обчислювальними ресурсами. Для токенізації тексту було обрано максимальну кількість слів 10,000, що дозволяє моделі працювати з обширним лексиконом, і максимальну довжину послідовності 100 токенів, що охоплює основний зміст опису помилок. Ці параметри забезпечують ефективне оброблення текстових даних для прогнозування серйозності помилок. Модель тренувалася протягом 10 епох із поступовим зниженням функції втрат та підвищенням точності на тренувальних та тестових даних.

На основі точності, повноти та F1-оцінки для кожного класу серйозності (Normal, Enhancement, Major, Minor, Critical, Blocker та Trivial) були отримані наступні результати роботи моделі зображені на рисунку 2.

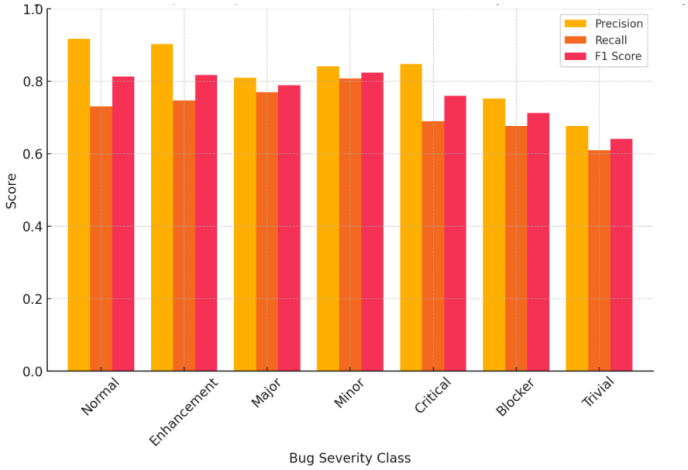


Рисунок 2. Метрики класифікації за рівнем серйозності

Ця діаграма показує значення Precision (точність), Recall (повнота) та F1 Score для кожного класу серйозності помилок ("Normal," "Enhancement," "Major," "Minor," "Critical," "Blocker," і "Trivial"). Модель працює найкраще для класів "Normal" і "Enhancement," маючи високі значення всіх метрик, що свідчить про високу точність в ідентифікації цих класів. Результати трохи зміщені на користь Precision, що вказує на меншу кількість хибних спрацьовувань порівняно з хибними пропусками. Класи "Major" і "Minor" показують збалансовані значення Precision, Recall та F1 Score, що свідчить про стабільну, але дещо нижчу продуктивність порівняно з найкращими класами.

Клас "Critical" демонструє помірні значення, що означає певні труднощі моделі у відокремленні його від інших класів, ймовірно, через більшу кількість хибних спрацьовувань або хибних пропусків. Класи "Blocker" і "Trivial" мають найнижчі значення, що вказує на найбільші труднощі моделі у точному прогнозуванні цих класів, можливо, через дисбаланс класів, схожість з іншими класами або обмеження самої моделі.

Розглянемо побудовані ROC-криві. Результат зображено на рисунку 3.

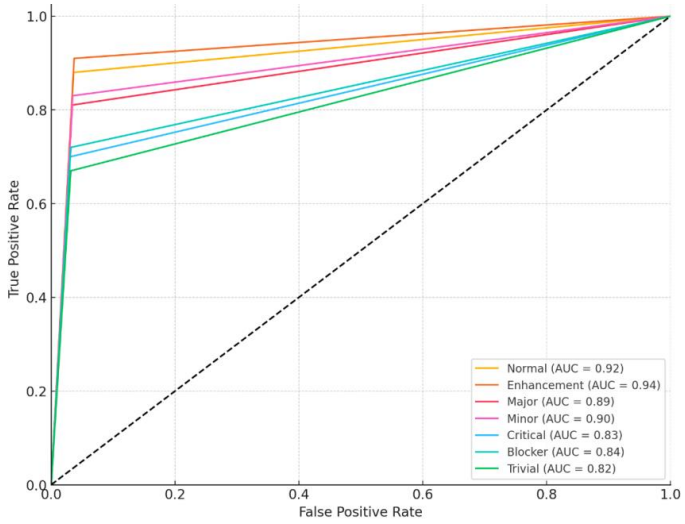


Рисунок 3. ROC-криві та значення AUC

Графік ROC показує, що модель чудово розпізнає класи "Enhancement" (AUC = 0,94) та "Normal" (AUC = 0,92), забезпечуючи високі значення True Positive Rate при низькому рівні хибних спрацьовувань. Класи "Minor" (AUC = 0,90) та "Major" (AUC = 0,89) також демонструють високі показники, що свідчить про стабільну та надійну продуктивність моделі для цих категорій.

Для класів "Critical" (AUC = 0,83), "Blocker" (AUC = 0,84) та "Trivial" (AUC = 0,82) модель показує дещо нижчі, але все ще непогані результати, що відкриває можливості для подальшого покращення. В цілому модель добре працює для більшості класів, а її продуктивність можна ще підвищити через додаткове налаштування, щоб максимально точніше відокремлювати всі класи серйозності.

4. ВИСНОВКИ

У статті було розглянуто проблему прогнозування серйозності програмних помилок за допомогою глибокої навчальної моделі на основі LSTM. Розроблена модель продемонструвала здатність ефективно класифікувати рівень серйозності помилок на основі текстових описів, що суттєво спрощує процес пріоритезації та аналізу помилок для команд розробників.

Під час експериментів модель досягла високих показників точності на тренувальних та тестових даних. Модель показує високі результати в розпізнаванні класів серйозності помилок, особливо для класів "Enhancement" (AUC = 0.94) та "Normal" (AUC = 0.92), демонструючи впевненість та точність у класифікації цих категорій. Це свідчить про високу здатність моделі коректно ідентифікувати ці класи, зберігаючи низький рівень хибних спрацьовувань і пропусків.

Класи "Major" і "Minor" також мають хороші показники з AUC, близькими до 0.9, що вказує на стабільність і надійність моделі для цих класів. Навіть класи з трохи нижчими AUC, такі як "Critical," "Blocker" і "Trivial," показують досить високі значення, що вказує на ефективність моделі у розрізненні цих класів, хоч і з дещо вищим рівнем хибних прогнозів.

Загалом модель LSTM демонструє високу продуктивність та потенціал для використання в задачах класифікації помилок, зберігаючи хороший баланс між точністю (Precision), повнотою (Recall) та загальною ефективністю (F1 Score). З невеликим доопрацюванням та налаштуванням, особливо для класів "Critical," "Blocker" та "Trivial," можна очікувати подальшого покращення результатів.

Застосування цієї моделі на практиці дозволить автоматизувати процес оцінки критичності помилок, що допоможе швидше вирішувати найважливіші проблеми та підвищити ефективність роботи команд підтримки та розробки. Подальші дослідження можуть зосередитися на покращенні моделі, підвищенні її стійкості до нових даних та адаптації до різних типів програмного забезпечення, щоб зробити її ще більш універсальною та надійною.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. An intuitive explanation of LSTM [Електронний ресурс]. URL: https://medium.com/@ottaviocalzone/an-intuitive-explanation-of-lstm-a035eb6ab42c

2. Understanding LSTM architecture: pros and cons and implementation [Електронний ресурс]. URL: https://medium.com/@anishnama20/understanding-lstm-architecture-pros-and-cons-and-implementation-3e0cca194094

3. Understanding precision, recall, F1-score, and support in machine learning evaluation [Електронний ресурс]. URL: https://medium.com/@nirajan.acharya777/understanding-precision-recall-f1-score-and-support-in-machine-learning-evaluation-7ec935e8512e

4. Goodfellow I., Bengio Y., Courville A. Deep Learning. Cambridge, MA: MIT Press, 2016.

5. LeCun Y., Bengio Y., Hinton G. Deep learning // Nature. 2015, vol. 521(7553). P. 436–444.

6. Krizhevsky A., Sutskever I., Hinton G.E. ImageNet Classification with Deep Convolutional Neural Networks // Advances in Neural Information Processing Systems. 2012, vol. 25.

СИСТЕМА ДИСТАНЦІЙНОГО НАВЧАННЯ НА ОСНОВІ ФОРМУВАННЯ ЄДИНОГО ІНФОРМАЦІЙНОГО ПРОСТОРУ

Косюк О.М.[1], Мухін В.Є.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] kosiuk.oleksiy@lll.kpi.ua, [2] v_mukhin@i.ua [0000-0003-2775-6071]

Метою даного дослідження є вивчення процесу дистанційного навчання, який включає використання інтегрованих інформаційних ресурсів та технологій для забезпечення ефективного навчального процесу, а також вивчення існуючих методів проектування, які мають на меті створення єдиної моделі дистанційного навчання, що забезпечуватиме логічну узгодженість між окремими її компонентами, високу семантичну інтероберабельність та високу швидкість і ефективність процесів проектування та розробки системи і розробку нових методів, які відповідатимуть цим вимогам. Наукова новизна роботи полягає у порівнянні ефективності сучасних методів проектування, таких як онтологічний метод чи предметно-орієнтоване проектування (Domain Driven Design, далі DDD) у прикладанні до задачі проектування системи дистанційного навчання. Крім цього, у цій роботі вперше вивчена можливість створення нового об'єктно-орієнтованого методу проектування на основі комбінування онтологічного методу, перевагою якого є високий рівень формалізації, та DDD, перевагою якого є його універсальність та простота.

Ключові слова: система дистанційного навчання, дистанційний навчальний процес, об'єктно-орієнтовані методи проектування, онтології, DDD.

1. ВСТУП

Процес модернізації освіти нерозривно пов'язаний з інформатизацією, що передбачає розвиток інтегрованої освітньої інформаційної системи та активне використання дистанційного навчання. Однак, відсутність єдиної системи опису об'єктів системи дистанційного навчання, зв'язків між ними та навчального контенту призводить до несумісності різних систем дистанційного навчання, обмежуючи їх семантичну інтероперабельність та ускладнюючи інтеграцію різноманітних електронних освітніх ресурсів. Для підвищення ефективності та швидкості проектування систем дистанційного навчання і покращення можливостей їх взаємної інтеграції необхідно впровадити семантичний опис предметної області на основі онтологічного аналізу, що забезпечить більш точне та структуроване представлення знань та дозволить створювати легко сумісні моделі різних систем дистанційного навчання.

Ця робота спрямована на дослідження ефективності вже існуючих методів (онтологічного та DDD) у прикладанні до задачі моделювання системи дистанційного навчання на базі єдиного інформаційного простору, а також на створення нових ефективних методів моделювання.

Крім цього, метою роботи є дослідження можливостей автоматизації усіх або деяких етапів моделювання системи дистанційного навчання з використанням запропонованих методів, а також можливе створення системи автоматизованого проектування.

Нарешті, кінцевою метою роботи є розробка системи дистанційного навчання з використанням зазначених вище методів і систем і перевірка їх ефективності на практиці.

2. ЗАСТОСУВАННЯ ОНТОЛОГІЧНОГО МЕТОДУ ТА МЕТОДУ ПРЕДМЕТНО-ОРІЄНТОВАНОГО ПРОЕКТУВАННЯ ДО ЗАДАЧІ ПРОЕКТУВАННЯ СИСТЕМИ ДИСТАНЦІЙНОГО НАВЧАННЯ

2.1. Онтологічний метод

Для подачі системи дистанційного навчання у вигляді онтології було використано середовище Protege. Спочатку були виділені найбільш узагальнені концепти, а вже потім вони розширювалися більш специфічними за допомогою відношення is-a. концепти і відношення наслідування між ними вказані на рис. 1.

Також у таблиці 1 представлені бінарні відношення (окрім is-a) між концептами онтології.

Загалом, головною перевагою такого представлення системи дистанційного навчання є високий рівень формалізації та детальний опис семантичних зв’язків між концептами. Онтологічний метод спрямований на забезпечення узгодженості і несуперечної інтерпретації даних у домені, але може не враховувати деякі деталі, необхідні для програмної реалізації системи.

Ще одною перевагою є наявність чітких стандартів і мов опису моделей, створених за допомогою онтологічного методу. На скріншоті ничже наведено фрагмент коду мовою OWL, який описує онтологію СДН.

2.2. Метод DDD

Метод предметно-орієнтованого проектування дозволяє швидко та гнучко представити предметну область та змоделювати систему, що відповідатиме конкретним вимогам. Цей метод складається з трьох етапів:

Big Picture Event Storming – метод для швидкого виявлення ключових бізнес-подій та побудови загального уявлення про домен. Залучає всі зацікавлені сторони, щоб знайти вузькі місця та визначити потенційні Bounded Contexts.

Process Modelling – деталізація бізнес-процесів із фокусом на взаємодії між подіями, командами й контекстами. Допомагає чітко структурувати процеси для подальшої імплементації.

Software Design – переклад бізнес-вимог на архітектуру та програмну реалізацію. Включає створення доменної моделі (Entities, Value Objects, Aggregates) і визначення API та сервісів.

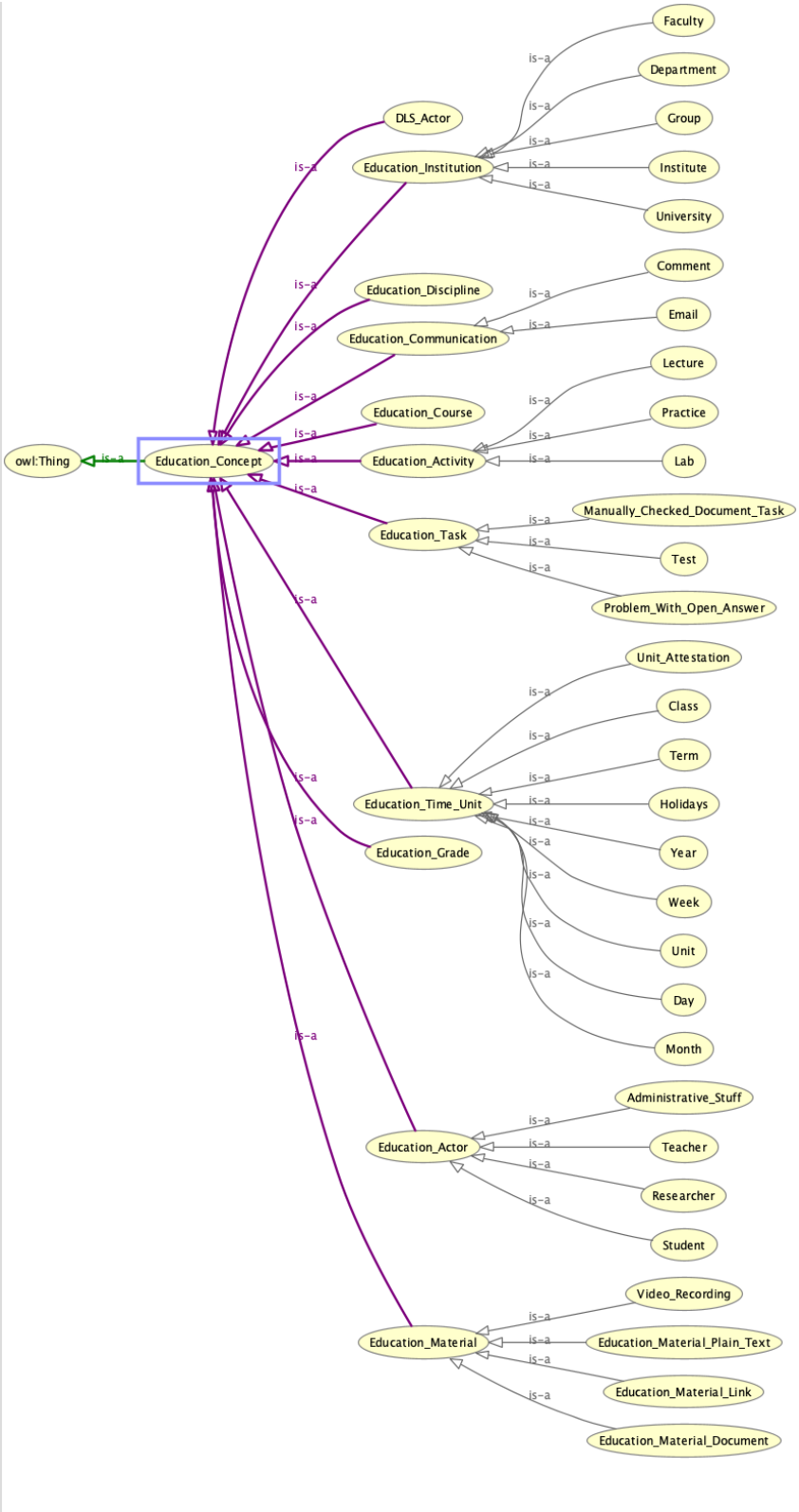


Рисунок 1. Концепти СДН і візуалізація відношень is-a між ними

Таблиця 1. Бінарні відношення між концептами онтології

Назва відношення	Domain -> Range	Обернене відношення
partOf	Education_Institution -> Education_Institution, Education_Time_Unit -> Education_Time_Unit, Education_Activity -> Education_Course, Education_Task -> Education_Activity, Education_Material -> Education_Activity	hasPart
hasPart	Education_Institution -> Education_Institution, Education_Time_Unit -> Education_Time_Unit, Education_Course -> Education_Activity, Education_Activity -> Education_Task, Education_Activity -> Education_Material	partOf
lasts	Education_Concept -> Education_Time_Unit	
hasRole	Education_Actor -> DLS_Actor	
studies	Student -> Education_Course	isStudiedBy
teaches	Teacher -> Education_Course	isTaughtBy
isStudiedBy	Education_Course -> Student	studies
isTaughtBy	Education_Course -> Teacher	teaches
memberOf	Education_Actor -> Education_Institution	
covers	Education_Course -> Education-Discipline	
isMaterialForTask	Education_Material -> Education_Task	
isAssignedTo	Education_Task, Education_Activity -> Education_Student	
isRated	Education_Task -> Education_Grade	
isRatedBy	Education_Task -> Teacher	
isCommentedOn	Comment -> Education_Task, Education_Activity	
isCommentedBy	Comment -> Education_Actor	
isSentTo	Email -> Education_Actor	
isSentBy	Email -> Education_Actor	

Нижче показано використання першого етапу DDD у прикладанні до проектування системи дистанційного навчання. Процес проектування всієї системи був дуже масштабним, тому показано процес опису лише життєвого циклу навчального курсу.

На етапі Big Picture Event Storming спочатку виписуються усі події, які є можливими в системі, серед них виділяються події контексту, що описують необхідний для початку процесу стан системи (на діаграмі Contextual Events) і безпосередньо події самого процесу, що відображають зміни станів системи під часу плину процесу (рис. 2).

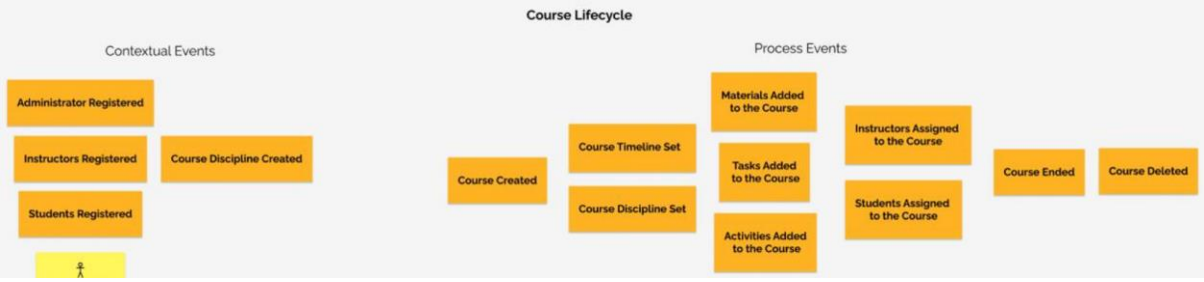


Рисунок 2. Події життєвого циклу навчального курсу

Далі виписуються актори (користувачі, що беруть участь в роботі системи, у даному випадку Адміністратор, Інструктор курсу і Студент) та системи (окрім безпосередньо системи дистанційного навчання можуть існувати також допоміжні підсистеми як Scheduling Subsystem – система для планування навчального процесу та сторонні системи, із якими взаємодіє СДН, такі як Email System) (рис. 3).

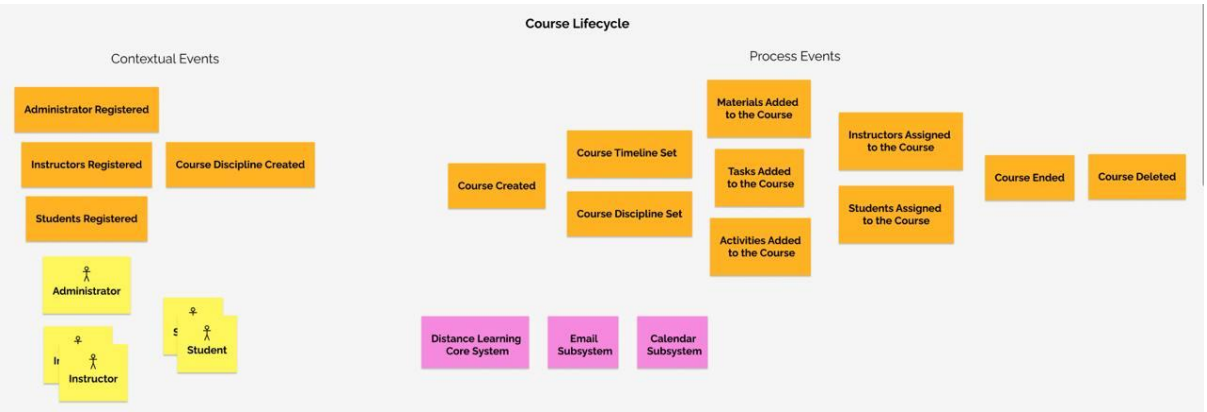


Рисунок 3. Події, системи та актори життєвого циклу навчального курсу

Після цього на етапі Process Modeling події впорядковуються у часі та за допомогою наступних зв’язків, зображених на рис. 4–6, моделюються процеси.

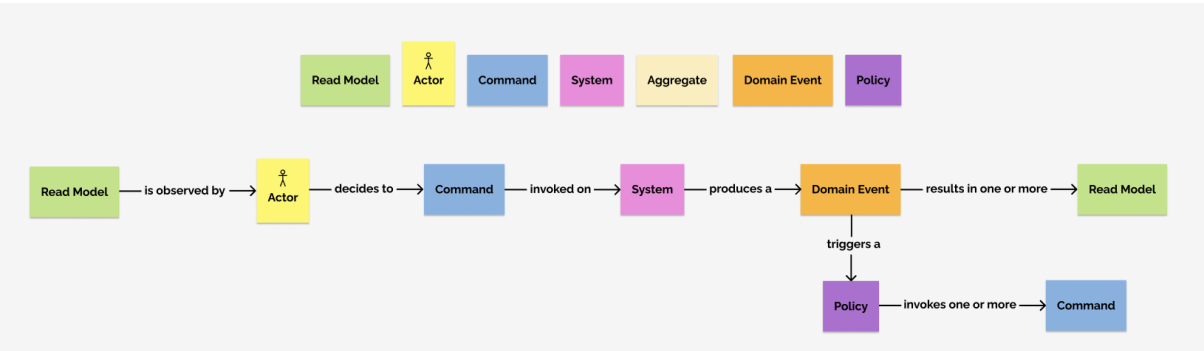


Рисунок 4. Схема моделювання процесів у DDD

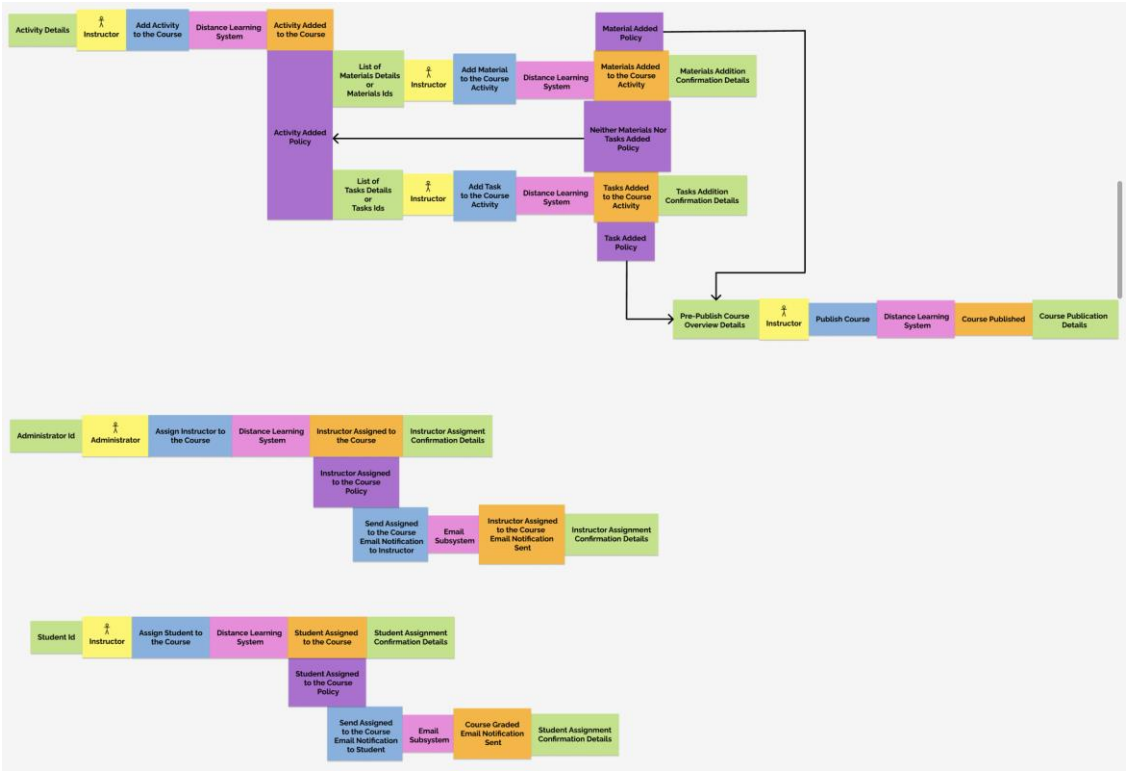


Рисунок 5. Змодельовані процеси життєвого циклу навчального курсу (частина 1)

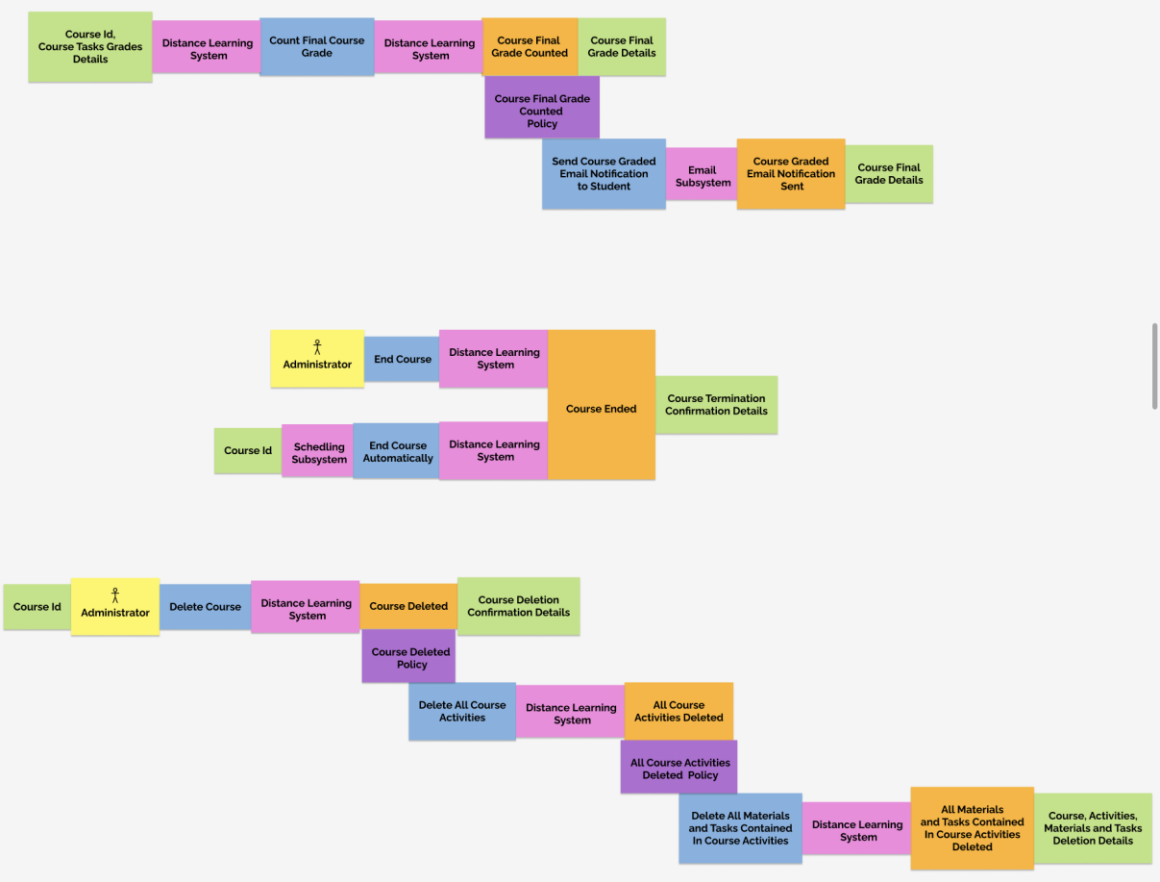


Рисунок 6. Змодельовані процеси життєвого циклу навчального курсу (частина 2)

Нарешті на етапі Software Design створюється доменна модель системи шляхом виділення агрегатів (aggregates), сутностей (entities) та об’єктів-значень (value objects).

Агрегат – сукупність об’єктів домену (як-от сутностей та об’єктів-значень), що об'єднані під одним коренем (Aggregate Root) і повинні розглядатися як єдине ціле.

Сутність – об'єкт домену, який ідентифікується за унікальним ідентифікатором, а не за своїми атрибутами.

Об'єкт-значення – об'єкт, який описує певне значення або характеристику і визначається своїми атрибутами, а не ідентифікатором.

Процес виділення агрегатів, сутностей і об'єктів-значень у доменній моделі базується на правилах, що враховують бізнес-логіку, зв'язки між об'єктами та вимоги тривалості життєвого циклу, консистентності й унікальності.

В нашому випадку ми спочатку виписуємо усі концепти, наявні в системі і пов’язані з ними концепти розміщуємо поряд (рис. 7–9).

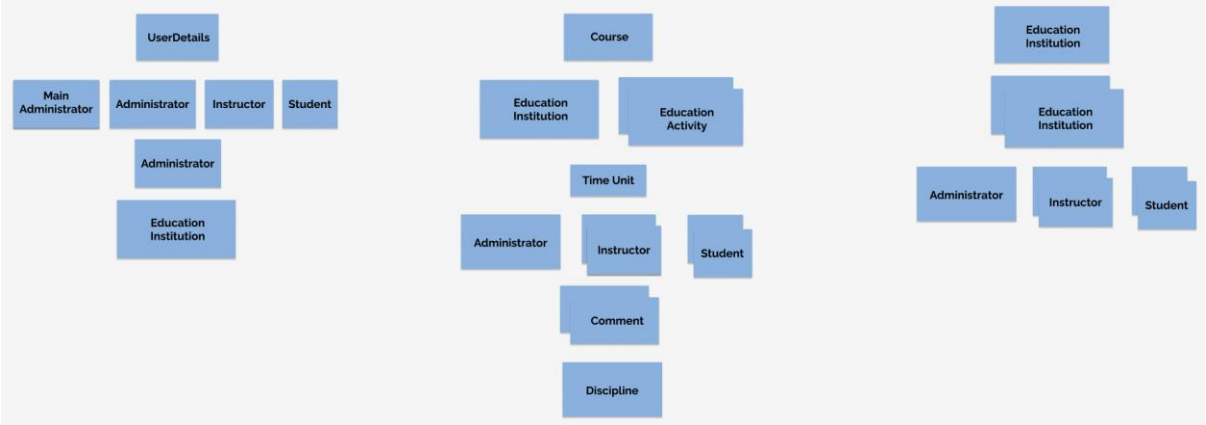


Рисунок 7. Взаємозв’язок між концептами системи дистанційного навчання

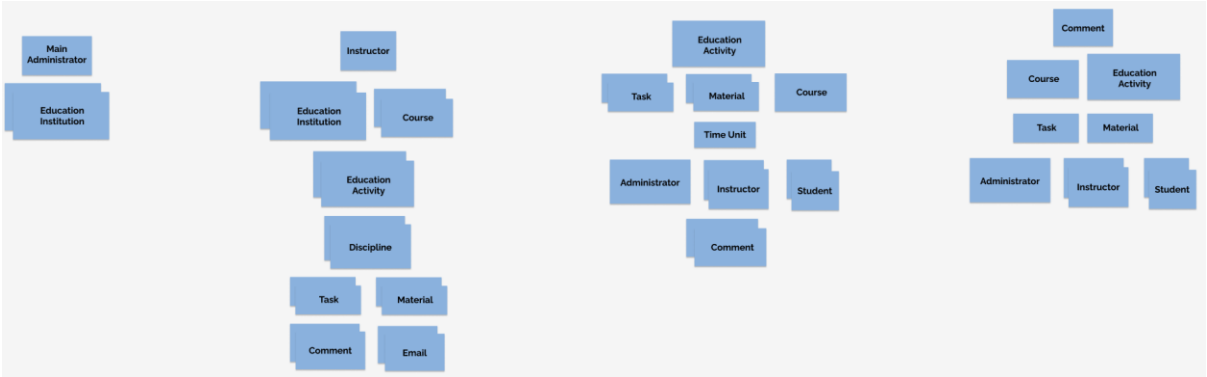


Рисунок 8. Взаємозв’язок між концептами системи дистанційного навчання

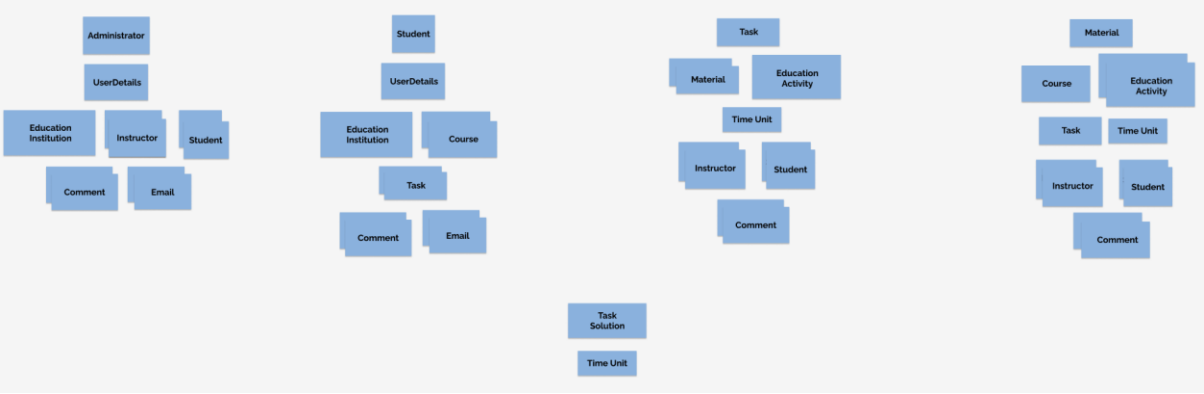


Рисунок 9. Взаємозв'язок між концептами системи дистанційного навчання

Після цього за допомогою вище зазначених правил виділяємо агрегати, сутності та об'єкти-значення, які позначаються кольорами, як показано на рис. 10.

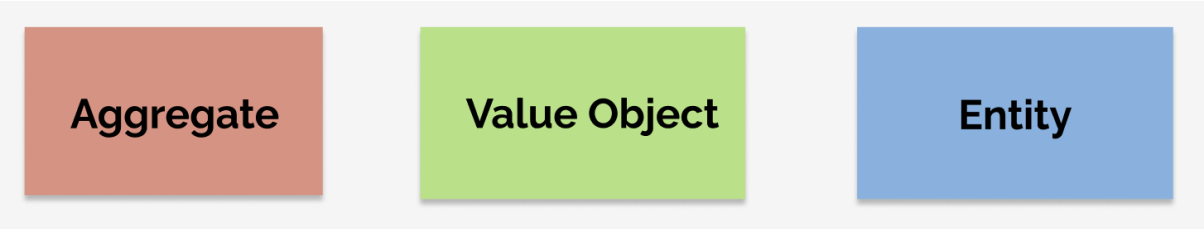


Рисунок 10. Кольори для позначення агрегатів, сутностей та об'єктів-значень

На відміну від онтологічного методу DDD орієнтований більше не на детальний опис семантичних зв'язків між сутностями і формалізацію знань, а на опис програмної моделі для реалізації бізнес-логіки. Наприклад, якщо не враховувати опис процесів, то опис взаємозв'язків між концептами на етапі Software Design обмежується просто відношенням "пов'язаний, має відношення, використовується" між парами концептів і кількісними характеристиками. На відміну від онтологічного методу, відношення накшталт "teaches", "isRatedBy", які відображають семантику зв'язків між концептами просто не використовуються.

2.3. Програмна реалізація спроектованої моделі

На базі отриманих моделей систему дистанційного навчання було реалізовано програмно. Для розробки використовувалася мова програмування Java та фреймворки Spring Boot і Hibernate. Важливою деталлю програмної реалізації є доменна модель системи, яка описана за допомогою агрегатів, сутностей та об'єктів-значень. Структура проекту, що відображає цю модель, наведена на рис. 11.

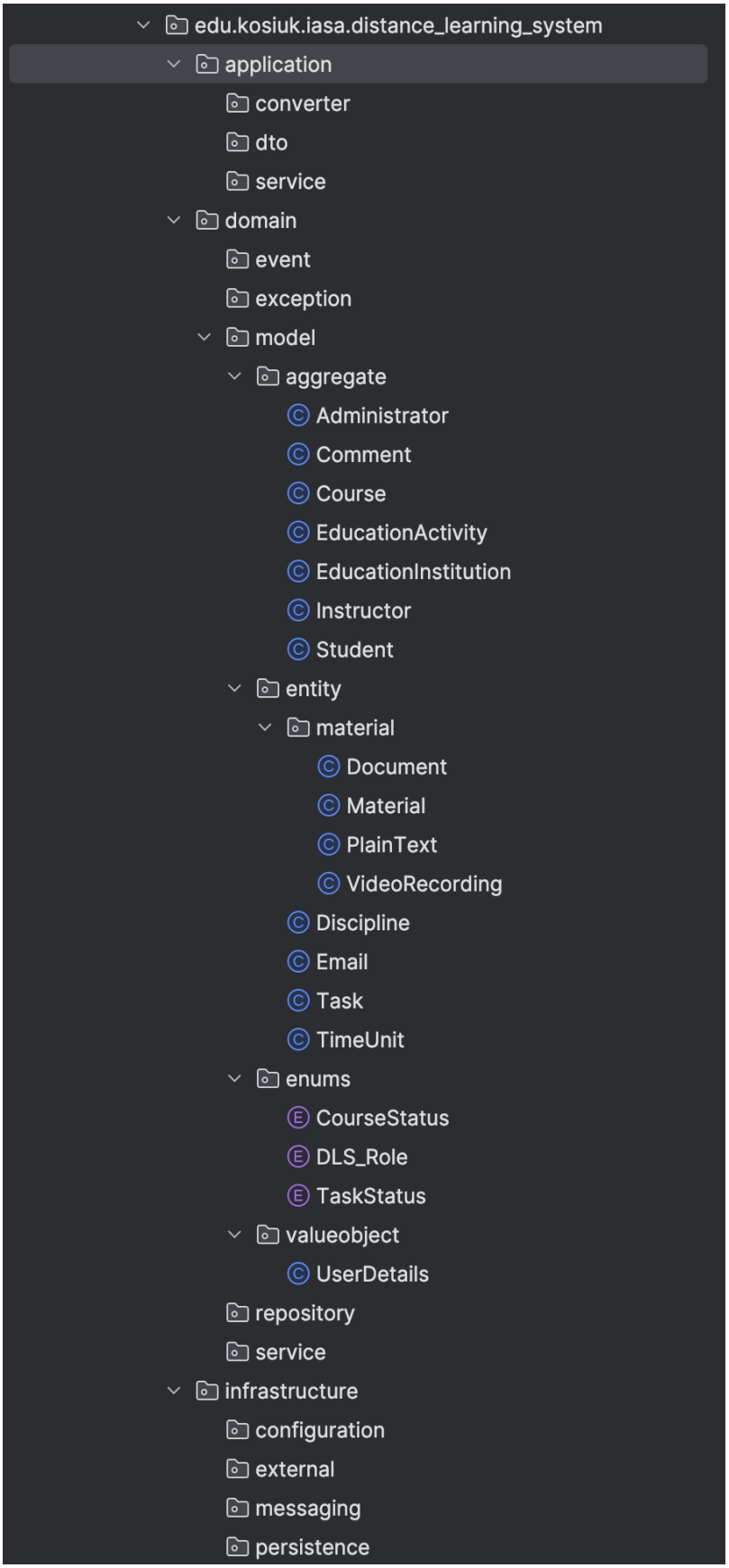


Рисунок 11. Структура проекту, що відображає доменну модель

Фрагменти коду не наведені виходячи з вимог до стислості опису роботи.

2.4. Висновки до розділу

Загалом, головними відмінностями DDD і онтологічного методу, які відображать їх переваги і недоліки, виявилися наступні:

1) Онтологічний метод більше підходить для проектування статичних моделей і майже не підходить для побудови динамічних, тобто моделювати якісь процеси, що змінюють стан із плином часу залежно від багатьох чинників за допомогою онтологічного методу майже неможливо. Якщо другий етап DDD – Process Modelling спрямований виключно на моделювання процесів і дозволяє створювати з різних подій, акторів, команд і систем узгоджені часові ланцюги, то в онтологічному методі такої опції не передбачено: можна створити окремо онтологію подій, окремо онтологію команд, онтологію акторів і так далі і визначити зв'язки і правила взаємодії між ними, але це будуть правила, що накладатимуть обмеження на поведінку системи вцілому, без моделювання окремих сценаріїв поведінки системи в часі. Наприклад, у створеній в цій роботі онтологічній моделі СДН є наступні відношення, наведені в табл. 2.

Таблиця 2. Бінарні відношення процесу оцінювання завдання

Назва відношення	Domain -> Range	Обернене відношення
isRated	Education_Task -> Education_Grade	
isRatedBy	Education_Task -> Teacher	
isSentTo	Email -> Education_Actor	
isSentBy	Email -> Education_Actor	

Відношення isRated між завданням і оцінкою показує, що у системі, яку описує онтологія, завдання може бути оцінене.

Відношення isRatedBy між оцінкою і викладачем показує, що оцінку ставить викладач.

Відношення isSentTo між електронним листом і будь-яким актором показує, що електронний лист може бути надісланий до будь-якого актора-отримувача.

Відношення isSentBy між електронним листом і будь-яким актором показує, що електронний лист може бути надісланий будь-яким актором-відправником.

Але при цьому всі ці відношення виступають у ролі окремих правил і обмежень, не моделюючи послідовний процес оцінювання завдання викладачем, можливого зниження оцінки шляхом застосування штрафу за завантаження розв'язку із запізненням і автоматичної відправки системою електронного листа з оцінкою до студента від імені викладача. При цьому у фрагменті DDD-моделі нижче чітко видно змодельований процес (рис. 14).

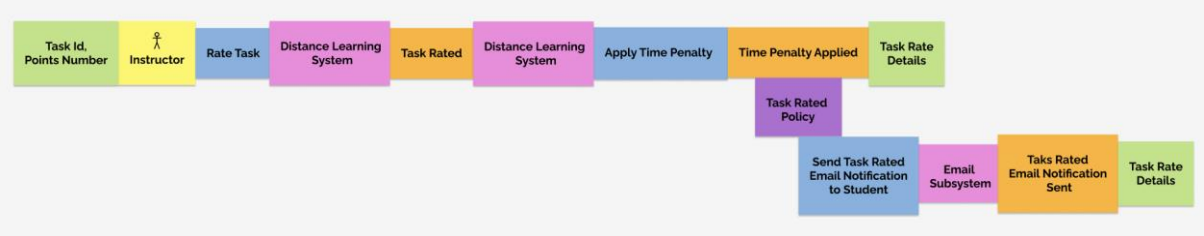


Рисунок 14. Модель процесу оцінювання завдання

Так само, використовуючи виключно онтологічний метод, важко врахувати деякі аспекти програмної реалізації оцінювання завдання викладачем, наприклад програмну реалізацію різних статусів завдання, що змінюються залежно від подій у системі і відображені за допомогою enum-у TaskStatus (рис. 15, 16).

```
package edu.kosiuk.iasa.distance_learning_system.domain.model.enums;

public enum TaskStatus {
    READY, ASSIGNED, COMPLETED, RATED
}
```

Рисунок 15. Enum TaskStatus

```
package edu.kosiuk.iasa.distance_learning_system.domain.model.entity;

import
edu.kosiuk.iasa.distance_learning_system.domain.model.aggregate.EducationActivit
y;
import edu.kosiuk.iasa.distance_learning_system.domain.model.aggregate.Student;
import
edu.kosiuk.iasa.distance_learning_system.domain.model.entity.material.Material;
import edu.kosiuk.iasa.distance_learning_system.domain.model.enums.TaskStatus;
import jakarta.persistence.*;
import lombok.*;

import java.util.Set;
import java.util.UUID;

@Entity
@NoArgsConstructor
@AllArgsConstructor
@Data
@Builder
@EqualsAndHashCode
public class Task {
    @Id
    @GeneratedValue(strategy = GenerationType.UUID)
    private UUID id;

    private String title;

    private TaskStatus status;

    @ManyToOne
    @JoinColumn(name = "student_id")
    private Student assignee;

    @ManyToOne
    @JoinColumn(name = "education_activity_id") // Foreign key column
    private EducationActivity educationActivity;

    @ManyToMany(mappedBy = "relatedTasks")
    private Set<Material> relatedMaterials;
}
```

Рисунок 16. Сутність Task і поле status у ній

2) При цьому серйозною перевагою онтологічного методу є його стандартизація (можна використовувати мови OWL та RDF для опису та збереження моделей), формалізація (правила

та обмеження домену можна формалізувати за допомогою інструментів, що використовують булеву алгебру (SWRL, SPARQL)) і можливість автоматизації процесу проектування за допомогою таких середовищ як Protege, що дозволяють автоматично аналізувати та генерувати моделі і відшукувати нові знання і відношення на базі вже існуючих за допомогою інструменту під назвою Reasoner.

3. РОЗРОБКА ВЛАСНОГО МЕТОДУ ПРОЕКТУВАННЯ СИСТЕМИ ДИСТАНЦІЙНОГО НАВЧАННЯ НА БАЗІ КОМБІНАЦІЇ DDD І ОНТОЛОГІЧНОГО МЕТОДУ

Враховуючи усі експериментально виявлені переваги та недоліки онтологічного методу і DDD у прикладанні до задачі проектування системи дистанційного навчання, спробуємо запропонувати метод проектування, що за допомогою комбінації онтологічного методу і DDD посилював би їх переваги та нівелював би недоліки.

Насамперед вирішимо який метод проектування буде основним, а який - допоміжним. Оскільки в даній роботі нас цікавить динамічна модель, яка описувала б систему таким чином, щоб її можна було реалізувати програмно, а за допомогою онтологічного методу, який більше підходить для опису баз знань і статичних моделей, створити таку модель майже неможливо, то основним методом проектування буде DDD, а онтологічний метод буде виступати як допоміжний на деяких етапах DDD.

Тепер визначимо етапи DDD, на яких використання онтологічного методу як допоміжного може суттєво підвищити ефективність проектування. Одразу зауважимо, що на етапі Process Modelling – другому етапі DDD, використання онтологічного методу є майже неможливим, тому що як вже було показано вище, онтологічний метод не дозволяє моделювати процеси, що змінюються в часі. Перший етап DDD – Big Picture Event Storming, є занадто абстрактним і недостатньо деталізованим аби використовувати такий високо формалізований метод як онтологічний. Але на етапі Software Design – третьому етапі DDD ми як раз маємо вже достатньо інформації про систему щоб використати онтологічний метод як допоміжний і при цьому процес виділення агрегатів, сутностей, об’єктів-значень та за можливості обмежених контекстів є статичним процесом і ідеально підходить під специфіку онтологічного методу.

Запропонуємо наступне покращення, яке може дати застосування онтологічного методу на останньому етапі DDD “System Design”:

Семантичне моделювання зв’язків між об’єктами

Покращення:

Використання онтологічної семантики для аналізу зв'язків між об’єктами дозволяє підвищити рівень формалізації автоматично визначити межі агрегатів, сутностей та об’єктів-значень і відношень між об’єктами в системі з вищою точністю.

Комбінація методів:

DDD: Виділяє агрегати, ґрунтуючись на транзакційних межах.

Онтологічний підхід: Використовує семантичний аналіз зв’язків між об’єктами для уточнення меж агрегатів.

Метод:

Аналіз відношень типу *part-of, is-a* і слабких відношень типу *depends-on*: у цій роботі наприклад відношення covers: Education_Course->Education_Discipline та studies: Student->Education_Course допомагає виділяти .

Відношення *part-of* автоматично визначає внутрішні межі агрегату;

Слабші відношення типу *depends-on* вказують на зовнішні залежності, що належать іншим агрегатам.

Відношення is-a допомагає відносити об’єкти в системі, пов’язані цим відношенням до одного типу (агрегат, сутність чи об’єкт-значення).

Також для розділу об’єктів, що не є *агрегатами*, на *сутності* і *об’єкти-значення* враховується наявність в об’єкту свого власного життєвого циклу: якщо об’єкт не є *агрегатом*, але має свій власний життєвий цикл, то такий об’єкт - це сутність, якщо ж не має власного життєвого циклу, то такий об’єкт - це об’єкт значення.

Послідовність:

1) За допомогою відношення *part-of* об’єкти групуються у кластери, де об’єкт найвищого рівня є *агрегатом.*

2) За допомогою слабких відношень по типу *depends-on* кластери і відповідно *агрегати* можуть ділитися на менші частини. Також за допомогою таких відношень можна виділити *обмежені контексти (Bounded Contexts)*.

3) Об’єкти, що не були виділені як агрегати, відносяться до *сутностей*, якщо вони мають власний життєвий цикл і до *об’єктів-значень* якщо вони такого циклу не мають.

4) Нарешті, за допомогою відношення is-a тип батьківського об’єкту наслідується дочірніми об’єктами.

Впродовж проектування на різних етапах методу тип об’єктів може мінятися у наступній послідовності: об’єкт-значення->сутність->агрегат. Але ні в якій іншій, тобто об’єкт, що одного разу вже був визначений *сутністю* вже не зможе стати *об’єктом-значенням*, але ще зможе стати *агрегатом*.

Приклад:

Приклад використовує фрагмент онтологічної моделі запропонованої у цій роботі раніше системи. Діаграма, що відображає онтологію станом на початок процесу моделювання наведена на рис. 17. Далі під описом кожного етапу моделювання наведена діаграма, що відображає зміни щодо виділення агрегатів, сутностей, об’єктів-значень і обмежених контекстів, що відбулися після завершення цього етапу.

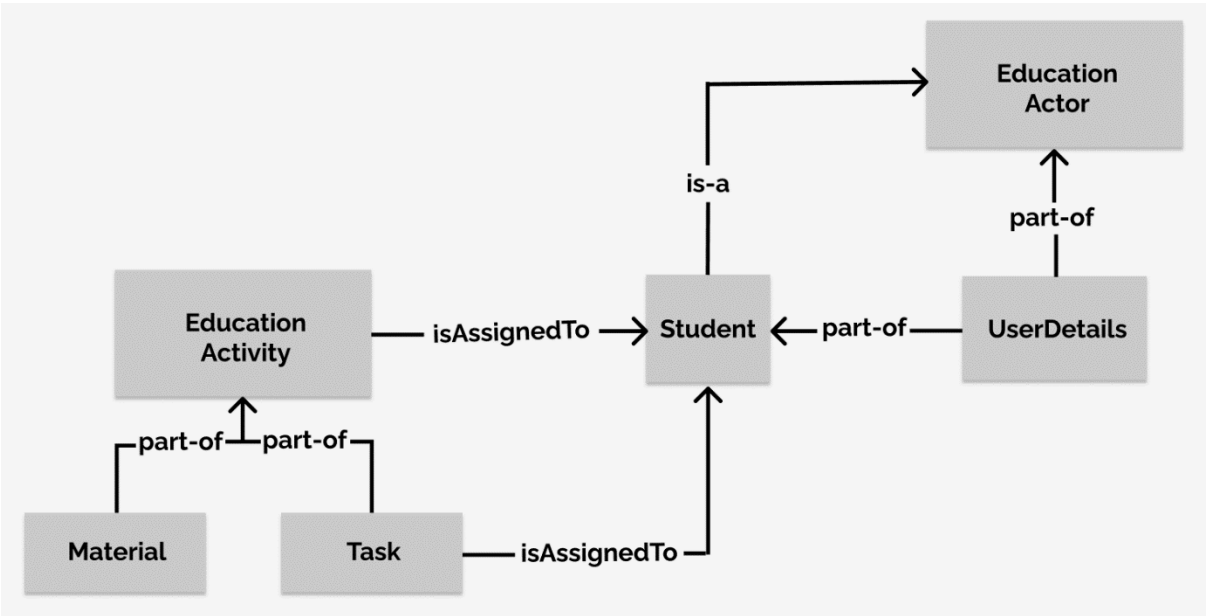


Рисунок 17. Фрагмент моделі на початку етапу System Design

1) Об’єкти (концепти) Task і Material пов’язані з об’ктом Education_Activity, який має свій власний життєвий цикл, за допомогою відношення *part-of.* Тому об’єкт Education_Activity

виділяється як *агрегат*. Так само об’єкт UserDetails пов’язаний з об’єктом Education_Actor відношенням *part-of*, тому об’єкт Education_Actor виділяється як агрегат (рис. 18).

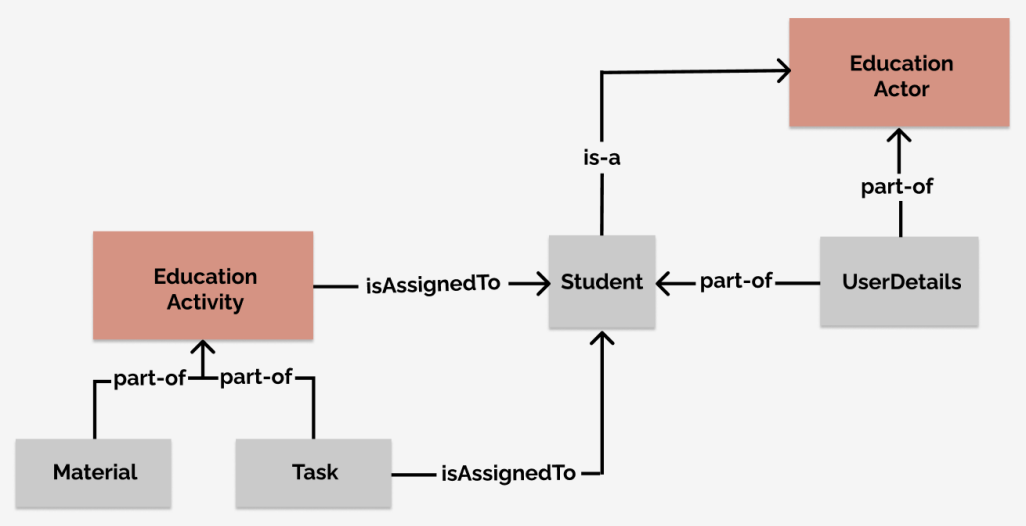


Рисунок 18. Фрагмент моделі після першого етапу

2) *Агрегат* Education_Activity і об’єкт Task пов’язані з об’єктом Student за допомогою слабкого відношення isAssignedTo, тому вони разом не є частиною одного більшого агрегату, але є частиною одного обмеженого контексту (рис. 19).

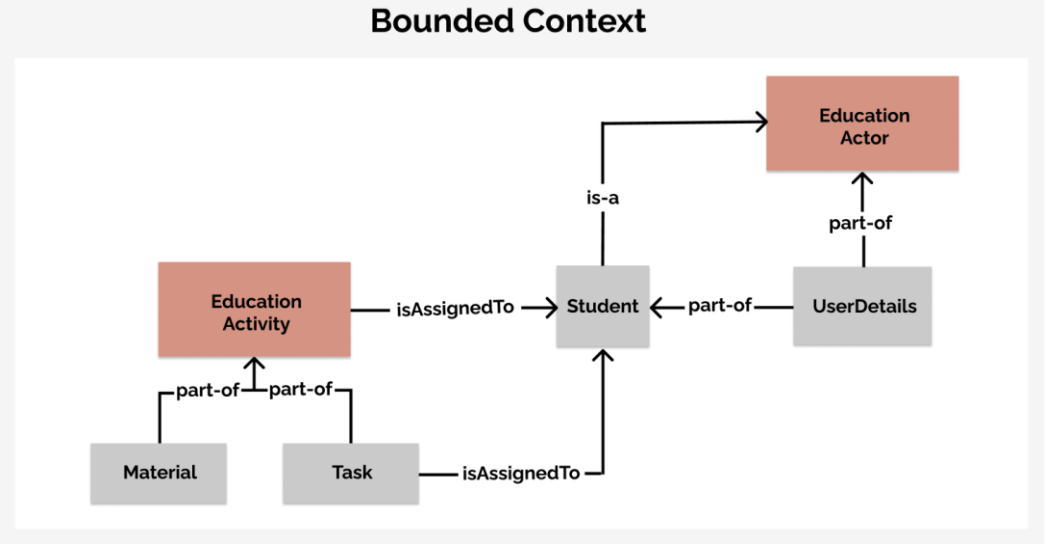


Рисунок 19. Фрагмент моделі після другого етапу

3) Об’єкт Student поки що не є *агрегатом,* але має свій власний життєвий цикл, тому виділяється як *сутність*. Об’єкт Task поки що не є *агрегатом,* але має свій власний життєвий цикл, тому виділяється як *сутність*. Об’єкт *Material* поки що не є *агрегатом,* але має свій власний життєвий цикл, тому виділяється як *сутність*. Об’єкт UserDetails поки що не є *агрегатом*, але не має свого власного життєвого циклу, тому виділяється як *об’єкт-значення* (рис. 20).

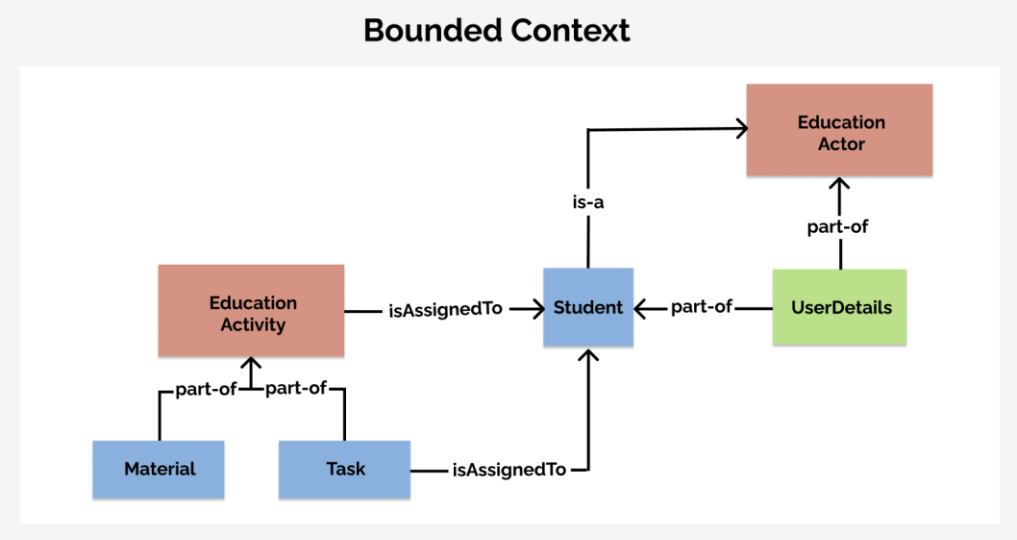


Рисунок 20. Фрагмент моделі після третього етапу

4) *Сутність* Student, має відношення is-a з *агрегатом* Education_Actor, тобто наслідує *агрегат* Education_Actor і є його дочірнім об'єктом, тому так само як і Education_Actor замість *сутності* стає *агрегатом* (рис. 21).

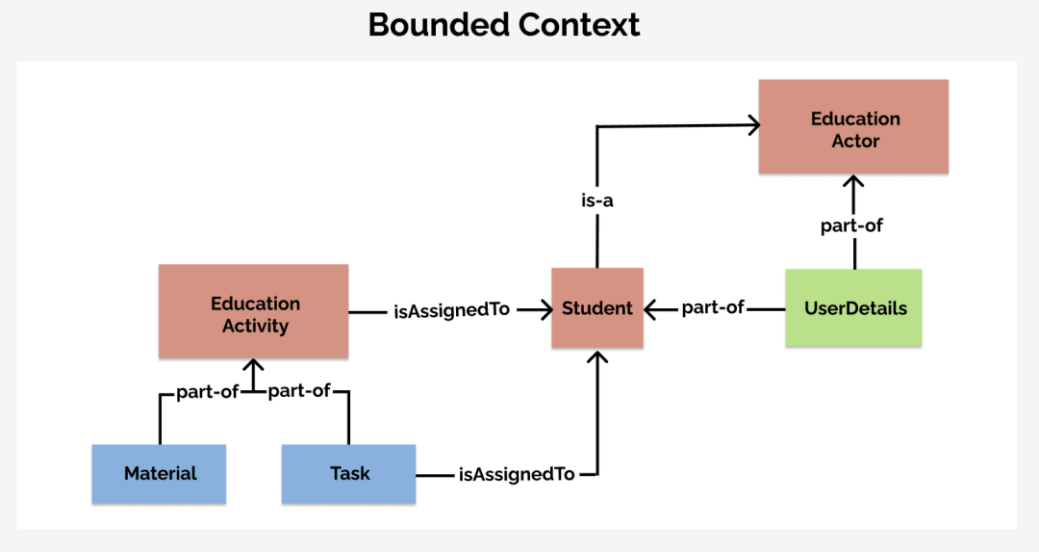


Рисунок 21. Фрагмент моделі після четвертого етапу

Результат:

На скріншоті (рис. 22) синім кольором наведено фрагмент структури проекту, змодельований на третьому етапі DDD System Design із використанням семантичного моделювання зв'язків за допомогою онтологічного методу. Порівняно із застосуванням класичного DDD перевагою такого підходу є чітка формалізація правил виділення агрегатів, сутностей, об'єктів-значень та обмежених контекстів за допомогою онтологічного методу, можливість автоматизації етапу System Design предметно-орієнтованого проектування завдяки наявності чітких правил і алгоритму, можливість використання вже існуючих мов опису онтологій, таких як OWL, для опису та збереження проміжних та кінцевих результатів етапу System Design у предметно-орієнтованому проектуванні.

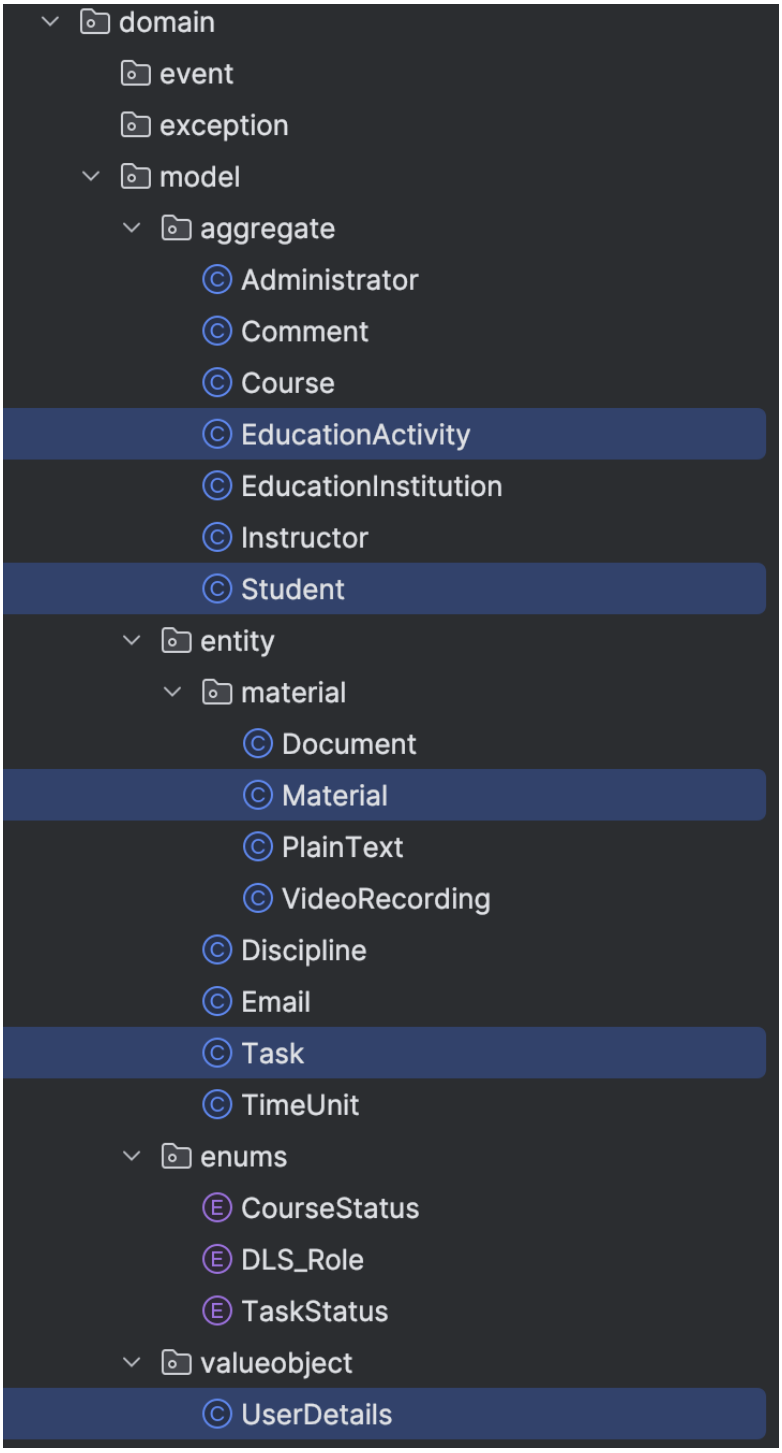


Рисунок 22. Фрагмент структури проекту, спроектований з використанням семантичного моделювання зв’язків

Покращений метод моделювання за допомогою DDD із використанням семантичного моделювання зв’язків за допомогою онтологічного методу на етапі System Design представлено на діаграмі (рис. 23).

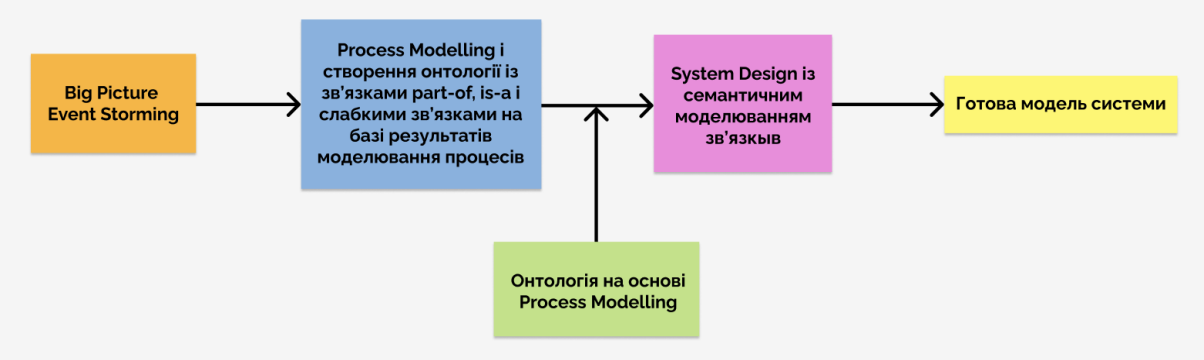


Рисунок 23. Покращений метод предметно-орієнтованого проектування (Domain Driven Design) з використанням семантичного моделювання зв'язків

4. ВИСНОВКИ

У даній роботі досліджено застосування таких методів як онтологічний метод та метод предметно-орієнтованого проектування (Domain Driven Design) до задачі проектування системи дистанційного навчання, проведено експеримент із проектування системи дистанційного навчання обома методами і створення програмної реалізації СДН на базі отриманих моделей. Проаналізовано результати експерименту та виявлено переваги та недоліки обох методів. Нарешті, створено новий метод моделювання - **предметно-орієнтоване проектування із семантичним моделюванням зв'язків за допомогою онтологій** і показано, що він є ефективнішим за класичний DDD завдяки високому рівню формалізації та чіткому опису правил проектування на останньому етапі Software Design, можливості часткової автоматизації процесу проектування і можливості використання вже існуючих мов опису онтологій, таких як OWL, для збереження проміжних результатів проектування в стандартизованому форматі.

Наступним етапом цієї роботи може бути програмна реалізація системи на базі створеного методу, яка дозволить частково автоматизувати процес предметно-орієнтованого проектування у прикладанні до задачі моделювання СДН.

АНАЛІЗ ТА ПОРІВНЯННЯ МЕТОДІВ ЗБОРУ ПОКАЗНИКІВ СЕРЦЕВОГО РИТМУ З ПРИСТРОЇВ ІОТ

Крушельницький І.М.[1], Харченко К.В.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] krushelniczkij.illia@lll.kpi.ua, [2] konst1970@gmail.com

Розвиток технології Інтернету речей суттєво вплинув на сферу медицини, де тепер відіграє ключову роль у моніторингу здоров'я пацієнтів. У роботі розглянуто три основні методи для збору показників серцевого ритму: фотоплетизмографія, електрокардіографія та біо-імпедансний метод, з описом принципів їх роботи, математичних основ та застосування в медицині. У порівнянні методів виокремлено переваги та обмеження кожного з них. Фотоплетизмографія підходить для носимих пристроїв завдяки своїй компактності та безконтактності, хоча точність знижується при русі. Електрокардіографія забезпечує високу точність, але потребує спеціалізованих електродів і складнішого обладнання. Біо-імпедансний метод є корисним для тривалого моніторингу, проте його точність може варіюватися в залежності від фізіологічних змін користувача.

Ключові слова: моніторинг серцевого ритму, медичні IoT пристрої, біометричні показники, методи вимірювання життєвих показників

1. ВСТУП

Розвиток технології Інтернет речей суттєво змінив наше життя. Люди інтегрували IoT пристрої у різні сфери – від побутової техніки до складних промислових виробництв. Завдяки можливостям IoT людство покращило рівні автоматизації, контролю та зручності виробничих та буденних процесів життя. Не дивно, що технологія Інтернет речей також знайшла своє місце в сфері медицини, де відіграє ключову роль у покращенні якості та доступності медичного обслуговування.

Використання IoT в медицині дозволяє виконувати постійний моніторинг стану здоров’я людей за допомогою різновидних сенсорів. Це може бути як звичайне вимірювання серцевого ритму, так і наприклад моніторинг якості сну. Безперервне відслідковування життєвих показників дає можливість оперативно реагувати на критичні зміни у стані здоров’я пацієнтів, що наприклад особливо важливо для людей із хронічними захворюваннями.

Не варто забувати, що у медицині точність даних є неймовірно важливим аспектом, оскільки вони впливають на правильність поставленого діагнозу та подальший план лікування. Неправильно зібрані дані можуть призвести до серйозних наслідків, включаючи ускладнення стану здоров’я пацієнта, або навіть летальних випадків. Це підкреслює необхідність дослідження та аналізу існуючих методів збору медичних даних, для того, щоб забезпечити якість, надійність та точність отримуваної інформації.

Дослідження в цій області може суттєво підвищити точність та оперативність медичної діагностики за допомогою технології Інтернет речей. Це забезпечить швидке прийняття рішень, поліпшить якість лікування, а також зменшить навантаження на медичний персонал.

Як наслідок, пацієнти отримають більш ефективне обслуговування і швидше виявлення захворювань, що сприятиме покращенню їхнього здоров'я.

2. МЕТОДИ ЗБОРУ ПОКАЗНИКІВ СЕРЦЕВОГО РИТМУ З ПРИСТРОЇВ ІОТ

2.1. Фотоплетизмографія

Фотоплетизмографія базується на аналізі змін у світловому поглинанні крові під час серцевих циклів. Кров має різні оптичні властивості в залежності від її оксигенації та об'єму під час скорочення серця і розслаблення.

Світодіод випромінює світло, яке проникає через шкіру. Частина світла поглинається еретроцитами, а частина відбивається назад. Фотодетектор реєструє інтенсивність світла, що повертається, і ці дані використовуються для аналізу змін об'єму крові у капілярах. На рисунку 1 графічно зображено принцип роботи [1].

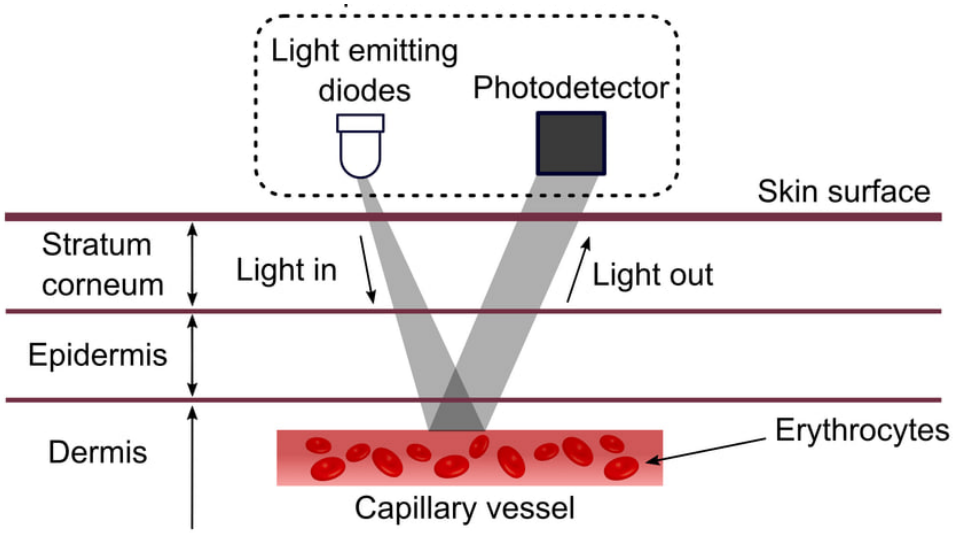


Рисунок 1. Принцип роботи фотоплетизмографії

Інтенсивність світла, що відбивається або поглинається, описується рівнянням Бугера-Ламберта-Бера (1):

$$I = I_0 * e^{-\mu_a * L} \quad (1)$$

де:

- I_0 – початкова інтенсивність світла,
- I – інтенсивність світла після проходження через тканину,
- μ_a – коефіцієнт поглинання тканини,
- L – довжина шляху світла через тканину.

Під час кожного серцевого циклу об'єм крові змінюється, змінюючи L, що й дозволяє фіксувати серцевий ритм [2].

Сигнал фотоплетизмографії є періодичним і може бути описаний як функція часу $S(t)$, яку часто піддають аналізу через перетворення Фур'є для виділення частоти пульсу (2):

$$S(t) = A * \sin(2\pi f t + \varphi) + \text{шум} \quad (2)$$

де:

- A – амплітуда сигналу,

- f – частота серцевого ритму,
- φ – фазовий зсув.

2.2. Електрокардіографія

Електрокардіографія (ЕКГ) базується на вимірюванні електричної активності серця, яка виникає під час кожного скорочення [3].

Електроди прикріплюються до тіла людини в області грудної клітини, рук та ніг. Під час кожного серцевого скорочення виникають електричні імпульси, які утворюються внаслідок скорочення (деполяризація) та розслаблення (реполяризація) серцевого м'яза. Ці імпульси поширюються через тканини тіла. Електроди фіксують сигнали та передають до апарата ЕКГ, який обробляє дані та створює графік активності серця у вигляді хвиль. На графіку (рис. 2) відображаються скорочення передсердь (хвиля P), скорочення шлуночків (комплекс QRS) та розслаблення шлуночків (хвиля T) [4].

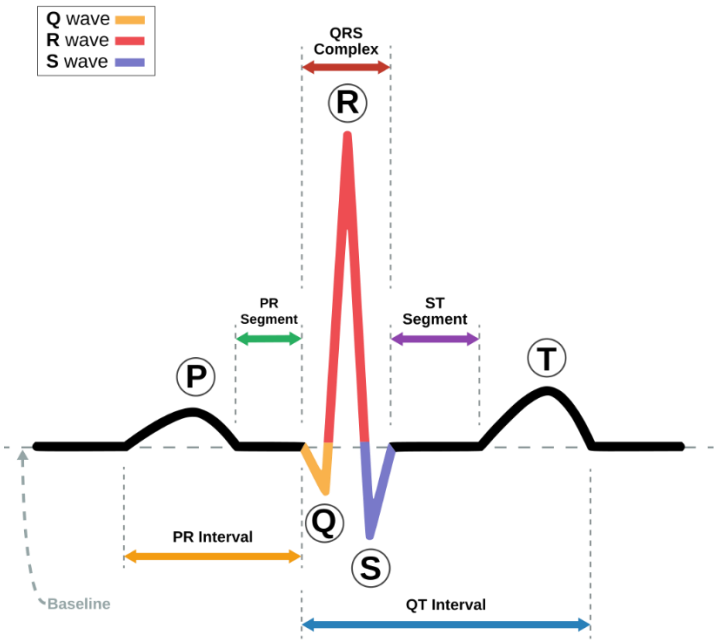


Рисунок 2. Основні елементи електрокардіограми

Електрична активність серця зображується на сітці квадратів, яку можна використати для обчислення серцевих скорочень. П'ять великих квадратів зображують 1 секунду, відповідно один великий квадрат відповідає 0,2 секундам, а маленький дорівнює 0,04 секунди. Найбільшим відхиленням на ЕКГ є зубець R, що і означає скорочення головного м'язу серця. Визначивши зубець R в кожному ударі, можна виміряти час, що проходить між одним серцебиттям і наступним [5].

При регулярному серцевому ритмі частоту серцевих скорочень можна визначити за допомогою великих квадратів (0,2 с) на ЕКГ, тоді частота серцебиття = 300 / к-ть великих квадратів між R-хвилями.

При швидкому серцевому ритмі дві послідовні R-хвилі можуть бути занадто близько одна до одної, щоб точно оцінити частоту серцевих скорочень людини за допомогою великих квадратів. У таких випадках частота серцевих скорочень визначається на основі менших квадратів, тоді частота серцебиття = 1500 / к-ть маленьких квадратів між R-хвилями [5].

Визначення частоти серцевих скорочень при нерегулярному серцевому ритмі є складним завданням, оскільки серцеві скорочення нестабільні і нерівномірно розташовані один від одного. Правильним рішенням в таких ситуаціях є залальний підрахунок R-хвиль за певний час (наприклад 10 секунд), тоді частота серцеюиття = к-ть R-хвиль за 10 секунд * 6.

2.3. Біо-імпедансний метод

Принцип даного методу базується на вимірюванні змін електричного опору тканин під час серцевих скорочень.

Кров має високий електричний опір, порівняно з іншими тканинами. При серцевих скороченнях кров проходить через судини, змінюючи загальний опір, який можна зафіксувати за допомогою електродів.

Для цього використовується набір сенсорів, які розташовуються на людському тілі. Сенсори можна розташовувати в області грудної клітини, на ногах чи руках, як зображено на рисунку 3 [6].

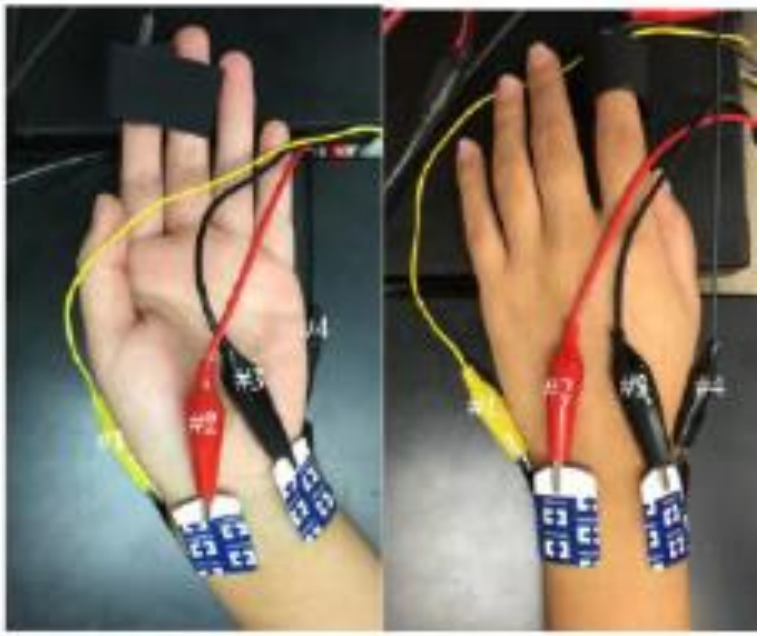


Рисунок 3. Біо-імпедансні сенсори розташовані на руці

Коли струм проходить через тіло, змінюється його опір у відповідь на різний об’єм крові, що циркулює в судинах під час скорочень серця. Зміни імпедансу реєструються сенсорами, які передають дані в систему для проведення розрахунків [7].

Імпеданс вираховується через закон Ома (3):

$$Z = \frac{V}{I}, \tag{3}$$

де:

- V – напруга,
- I – струм.

Зміни в імпедансі відповідають серцевим скороченням. Визначивши період між двома послідовними піками цих змін, можна обчислити частоту серцевих скорочень, використовуючи формулу частоти (4):

$$HR = \frac{60}{T}, \quad (4)$$

де T – період між двома послідовними піками імпедансу, які відповідають одному серцевому ритму.

3. ПОРІВНЯННЯ ОПИСАНИХ МЕТОДІВ

Кожен із методів має різні принципи роботи, точність, вимоги до обладнання, чутливість до зовнішніх факторів та інші особливості. Проведемо порівняння розглянутих методів за наступними критеріями:

- точність,
- чутливість до руху,
- області застосування,
- простота інтеграції в IoT пристрої,
- час вимірювання.

Для кращого сприйняття порівняння сформовано у вигляді таблиці 1:

Таблиця 1. Порівняння методів збору показників серцевого ритму за певними метриками

Метод	Точність	Чутливість до руху	Області застосування	Простота інтеграції в IoT	Час вимірювання
Фотоплетизмографія	Середня	Висока, точність вимірювання суттєво зменшується при русі	Смарт-годинники, фітнес-трекери, медичні пристрої домашнього моніторингу	Висока, можна інтегрувати в компактні пристрої	1–5 секунд
Електрокардіографія	Висока	Висока, людина повинен знаходитись в малорухомому стані	Медичні пристрої, лікарні, портативні монітори для серцевої діяльності	Середня, вимагає більше навичок і складніша в інтеграції	5–10 секунд
Біо-імпедансний метод	Середня	Низька, якщо немає інших чинників, які можуть змінити стан людини	Медичні пристрої, фітнес пристрої	Середня, необхідні додаткові сенсори для більш точного вимірювання	5–15 секунд

Порівняння методів показує, що фотоплетизмографія відзначається простотою інтеграції в носимі пристрої, але втрачає точність під час руху; електрокардіографія є високоточним

методом, що вимагає спеціалізованого обладнання; біо-імпедансний метод підходить для тривалого моніторингу, однак його точність може варіюватися залежно від фізіологічних змін.

4. ВИСНОВКИ

У роботі розглянуто різні методи збору показників серцевого ритму з медичних IoT-пристроїв, включаючи фотоплетизмографію, електрокардіографію та біо-імпедансний метод. Детально описано принципи роботи цих методів та їх математичну основу. Проведено порівняльний аналіз їх переваг і обмежень, що дозволило виявити найефективніші підходи для різних умов застосування.

Науковим здобутком роботи є детальний аналіз та порівняння методів збору даних, що можуть бути використані для подальшого розвитку та впровадження ефективних систем моніторингу, які забезпечать високу якість діагностики та безперервне спостереження за пацієнтами.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Advances in Photopletysmography Signal Analysis for Biomedical Applications. URL: https://www.researchgate.net/publication/325675749_Advances_in_Photopletysmography_Signal_Analysis_for_Biomedical_Applications (дата звернення: 10.11.2024).

2. Photoplethysmography in Wearable Devices: A Comprehensive Review of Technological Advances, Current Challenges, and Future Directions. URL: https://www.mdpi.com/2079-9292/12/13/2923 (дата звернення: 10.11.2024).

3. Cardiology Explained. Conquering the ECG. URL: https://www.ncbi.nlm.nih.gov/books/NBK2214/ (дата звернення: 10.11.2024).

4. Understanding an ECG. URL: https://geekymedics.com/understanding-an-ecg/ (дата звернення: 11.11.2024).

5. How to calculate heart rate from ECG (or EKG). URL: https://seermedical.com/blog/calculate-heart-rate-ecg#what-is-an-ecg (дата звернення: 11.11.2024)

6. Wristband Heartbeat Monitoring System Based on Bio-Impedance Analysis. URL: https://www.researchgate.net/publication/307560583_Wristband_Heartbeat_Monitoring_System_Based_on_Bio-Impedance_Analysis (дата звернення 12.11.2024)

7. Electrical impedance. URL: https://www.britannica.com/science/electrical-impedance (дата звернення: 12.11.2024)

МОДЕЛЬ КЛАСИФІКАЦІЇ ІНФОРМАЦІЙНИХ ОБ'ЄКТІВ НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ ТА НЕЧІТКОЇ ЛОГІКИ

Мухін О.В.[1], Кисельов Г.Д.[2]

Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”, Київ, Україна

[1]o.mukhin01@gmail.com [0009-0005-5301-8276],
[2] g.kyselov@gmail.com [0000-0003-2682-3593],

У даній доповіді розглядається питання реалізації автоматизованих методів класифікації інформаційних матеріалів, що дозволить оптимізувати процес їх пошуку. Сучасні механізми класифікації, зокрема нейронні мережі та нечітка логіка, можуть бути ефективними у вирішенні цієї проблеми, дозволяючи обробляти складні і нечіткі дані, які часто виникають в інформаційному середовищі.

Ключові слова: нейронні мережі, нечітка логіка, інформаційні об’єкти, класифікація.

1. ВСТУП

Основна мета даного дослідження – розробка та реалізація моделі класифікації інформаційних об'єктів на основі поєднання нейронних мереж і нечіткої логіки. Запропонована модель спрямована на підвищення точності класифікації інформаційних матеріалів за допомогою обробки нечітких та неоднозначних даних. До таких матеріалів відносяться, зокрема, текстові документи, відео, аудіофайли та інші мультимедійні об’єкти. В ході дослідження поставлено завдання розробити механізм побудови нечітких правил, створити алгоритм класифікації з використанням нейронних мереж та провести експериментальну перевірку ефективності моделі на реальних даних.

2. МОДЕЛЬ НА ОСНОВІ НЕЧІТКОЇ ЛОГІКИ ТА НЕЙРОННИХ МЕРЕЖ

Розглянута модель класифікації інформаційних об'єктів, яка базується на поєднанні нейронних мереж і нечіткої логіки. Використання нейронної мережі для обробки чітких характеристик дозволяє моделі навчатись на нових даних і адаптуватися до змін у властивостях інформаційних матеріалів [1, 2]. У свою чергу, нечітка логіка ефективно працює з нечіткими характеристиками, що дозволяє створювати систему, здатну класифікувати об’єкти, характеристики яких можуть бути нечіткими або двозначними [3–7].

Процес класифікації складається з кількох етапів [1]:

1. Фазифікація – перетворення точних значень характеристик об'єктів на нечіткі множини, що дає можливість враховувати нечіткість, яка є характерною для реальних даних у інформаціних середовищах.

2. Створення бази нечітких правил – розробка набору правил, що описують залежність між характеристиками об'єктів та їх класами.

3. Модуль прийняття рішень – використання розроблених правил для обчислення результатів, що визначають клас об'єкта.

3. Дефаззифікація – перетворення нечітких множин у конкретні значення, завдяки чому можна визначити підсумковий клас об’єкта.

Використовуються нечіткі продукційні мережі, такі як модель Мамдані-Заде і модель Ванга-Менделя, через їх здатність класифікувати об’єкти, які можуть належати до кількох класів одночасно. Це особливо важливо для матеріалів, які мають різні формати та містять складні характеристики.

На етапі фазифікації точна множина вхідних даних $IO_{in} = (IO_{in_1}, IO_{in_2}, \dots, IO_{in_i}) \in FS_{in}$ перетворюється в нечітку множину $Fuzz \subseteq FS_{in}$ за допомогою конкретної функції приналежності $f_{Fuzz}(io)$, де: $Fuzz = Fuzz_1 \times Fuzz_2 \times \dots \times Fuzz_i$, де i – кількість інформаційних об’єктів [1].

Всі S правил пов'язані між собою логічною операцією «АБО», при цьому виходи правил $IO_{out}^1, IO_{out}^1, \dots, IO_{out}^S$ вважаються взаємно незалежними. Отже, можна використовувати правила наступного типу [1]:

$Rule^j$: Якщо $IO_{in_1}^j$ належить $Fuzz_1^j$ ТА $IO_{n_2}^j$ належить $Fuzz_2^j$ ТА ... ТА $i_{in_i}^j$ належить $Fuzz_i^j$, ТО IO_{out}^j належить FS^j.

3. ОПИС МОДЕЛІ КЛАСИФІКАЦІЇ

Модель включає два основні компоненти:

1. Класична нейронна мережа – використовується для обробки точних параметрів інформаційних об'єктів. Структура мережі включає вхідний шар, де кожен нейрон представляє певну характеристику об'єкта, прихований шар для виявлення патернів, та вихідний шар для кінцевої класифікації (рис. 1).

Модель нейронної мережі для класифікації інформаційних об’єктів (*IO*) складається з таких компонентів:

1) Вхідний шар: кожен вузол цього шару представляє певний аспект інформаційного об’єкта. Наприклад, це можуть бути характеристики тексту, зображень, аудіо або відео, в залежності від типу інформаційного матеріалу.

2) Ваги та зв'язки: кожен зв’язок між вузлами вхідного та прихованого шарів має свою вагу, що визначає важливість кожного аспекту під час класифікації. Ці ваги налаштовуються в процесі навчання нейронної мережі.

3) Прихований шар: цей шар складається з нейронів, які виконують операції лінійних комбінацій вхідних даних з їхніми вагами та застосовують функції активації. Прихований шар дозволяє моделі витягувати більш абстрактні та складні ознаки з вхідних даних.

4) Функції активації: кожен нейрон у прихованому шарі активується за певних умов, заданих функцією активації. Включення нелінійностей дає змогу нейронній мережі навчатися складним взаємозв’язкам у даних.

5) Вихідний шар: формує результат нейронної мережі, передбачаючи приналежність об’єкта до певного класу. Кількість вузлів у вихідному шарі відповідає кількості класів, за якими здійснюється класифікація.

6) Функція втрат: визначає, наскільки передбачення моделі відповідає фактичному класу. Це дозволяє оцінити якість роботи нейронної мережі та коригувати ваги під час навчання.

7) Навчальна вибірка: нейронна мережа навчається на навчальній вибірці, що містить дані про об’єкти та відповідні їм мітки класів. Навчання полягає в налаштуванні ваг мережі так, щоб мінімізувати значення функції втрат.

8) Алгоритм навчання: визначає метод, за допомогою якого здійснюється навчання нейронної мережі. Один із найбільш поширених алгоритмів — зворотне поширення помилки (Backpropagation).

2. Нейронна мережа з нечіткими параметрами – містить набір нечітких правил для кожного класу об’єктів. Кожне правило має антецедент (умови активації) та консеквент (результат виконання правила).

Кожне правило складається з двох частин [1]:

- Антецедент – визначає умови активації правила на основі вхідних параметрів.
- Консеквент – визначає, до якого класу відноситься об’єкт. Це дозволяє обробляти дані навіть у випадках нечітко визначених або неоднозначних характеристик об'єкта.

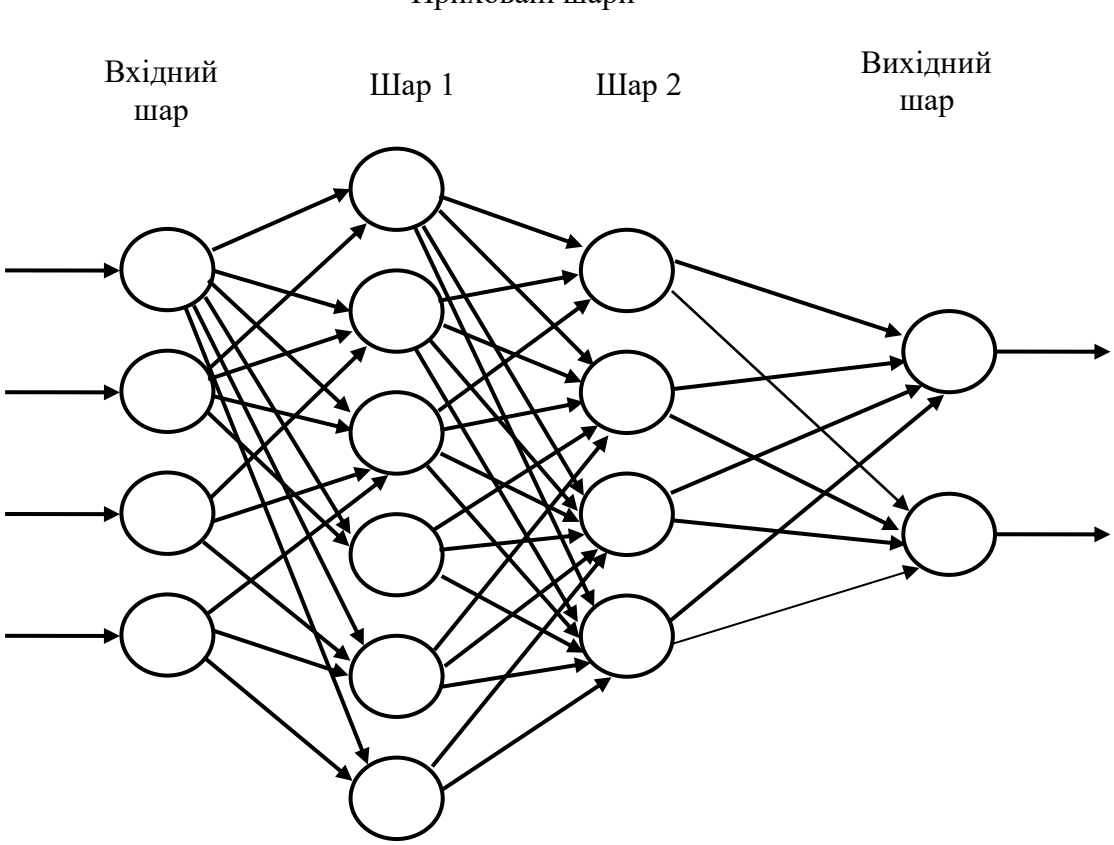


Рисунок 1. Модель нейронної мережі

4. ЕКСПЕРИМЕНТАЛЬНЕ ТЕСТУВАННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

Для перевірки ефективності моделі було проведено експерименти з реальними інформаційними матеріалами. Дані були поділені на навчальну та тестову вибірки у співвідношенні 80/20, що забезпечило репрезентативність результатів. Після підготовки даних модель тренували на протязі 1000 епох із використанням різних варіантів нейронних мереж для знаходження оптимальних параметрів.

Результати показали, що модель досягла високих показників точності, зокрема за умов поєднання нейронної мережі та нечіткої логіки. Це дозволило точно визначити клас матеріалу навіть за умов нечітких вхідних даних. Результати експерименту свідчать про високу ефективність застосування нечіткої логіки у класифікації інформаційних об'єктів, що надає моделі адаптивності та гнучкості. Використання нечітких правил дозволяє обробляти матеріали навіть у випадках нечітко визначених або неоднозначних параметрів, що значно спрощує роботу з різноманітними типами даних.

У реальних умовах ця модель дозволяє автоматизувати процес класифікації інформаційних матеріалів. Масштабування моделі дозволяє застосовувати її для великих платформ, що є перспективним напрямком розвитку.

5. ПЕРСПЕКТИВИ ТА ОБМЕЖЕННЯ МОДЕЛІ

Незважаючи на високі результати, модель має обмеження, зокрема, потребу у створенні великої бази нечітких правил, що потребує експертних знань. Для розширення можливостей моделі перспективним є автоматизація процесу створення правил.

Додатковим напрямком для майбутніх досліджень є оптимізація моделі для обробки великих обсягів даних. Важливим завданням є скорочення часу обчислень через використання паралельної обробки та хмарних обчислень. Інтеграція таких методів дозволить підвищити швидкість та точність роботи моделі, роблячи її більш придатною для масштабних онлайн-платформ.

6. ВИСНОВКИ

У доповіді представлено інноваційну модель класифікації інформаційних об'єктів на основі нейронних мереж та нечіткої логіки. Запропонована модель демонструє високу точність і гнучкість у класифікації інформаційних матеріалів, дозволяючи ефективно автоматизувати управління контентом на різних платформах.

Модель також має значний потенціал для застосування у майбутніх дослідженнях, зокрема, у сфері персоналізованого навчання, де можлива інтеграція з адаптивними системами для індивідуального підбору навчального контенту на основі рівня знань і прогресу студента.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. V. Mukhin, V. Zavgorodnii, V. Liskin, S. Syrota, V. Koval, L. Honchar. Classification of Information Objects with Fuzzy Parameters in E-Learning Systems. IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Dortmund, Germany, 2023, pp. 1189-1193, doi: https://10.1109/IDAACS58523.2023.10348768

2. Valerii Zavgorodnii, Nadiya Braykovs′ka, Oleksandr Yarovyi, Anna Zavgorodnya, Viacheslav Liskin, Oleg Mukhin, "The Method of Restoring Parameters of Mobile Agents in a Unified Dynamic Environment Considering Similarity Coefficients", International Journal of Computer Network and Information Security (IJCNIS), Vol.15, No.4, pp.25-35, 2023. DOI: https://doi.org/10.5815/ijcnis.2023.04.03

3. Paulo Vitor de Campos Souza. Fuzzy neural networks and neuro-fuzzy networks: A review the main techniques and applications used in the literature. Applied Soft Computing. Vol. 92. 2020. ISSN 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2020.106275

4. Wang, X., Chen, Y., Jin, J. et al. Fuzzy-clustering and fuzzy network based interpretable fuzzy model for prediction. Sci Rep 12, 16279 (2022). https://doi.org/10.1038/s41598-022-20015-y

5. Askari, S., Montazerin, N. & Zarandi, M. F. Generalized possibilistic fuzzy c-means with novel cluster validity indices for clustering noisy data. Appl. Soft Comput. Vol. 53. 2017. P. 262–283. DOI: https://doi.org/10.1016/j.asoc.2016.12.049

6. Mansoureh, N., Mohammad HosseinFazel, Z. & Susan, B. A fuzzy cluster-validity index based on the topology structure and node attribute in complex networks. Expert Syst. Appl. 187, 115913. 2022. DOI: https://doi.org/10.1016/j.eswa.2021.115913

7. Alipour, S., Shanbehzadeh, J. Fast automatic medical image segmentation based on spatial kernel fuzzy c-means on level set method. Machine Vision and Applications. Vol. 25. 2014. P.1469–1488. DOI: https://doi.org/10.1007/s00138-014-0606-5

ХМАРНІ ТЕХНОЛОГІЇ ТА СЕРВІСИ ДЛЯ ОБРОБКИ ДАНИХ З МЕДИЧНИХ ІОТ ПРИСТРОЇВ

Насікан Д.Ю.[1], Харченко К.В.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] nasikan.dmytro@lll.kpi.ua [0009-0007-1840-4344], [2] konst1970@gmail.com

Мета цієї роботи – всебічний аналіз задачі побудови систем для обробки даних з медичних IoT пристроїв за допомогою хмарних технологій та сервісів. У роботі розглянуто структуру таких систем, проаналізовано можливості сучасних хмарних технологій та сервісів для їх побудови та наведено порівняння окремих сервісів пропонованих хмарними провайдерами. Також, розглянуто типову архітектуру таких хмарних систем.

Ключові слова: AWS, GCP, Azure, хмарні технології та сервіси, медичні IoT пристрої, обробка даних.

1. ВСТУП

Використання хмарних технологій і сервісів для обробки даних з медичних IoT пристроїв набуває все більшого значення у сучасній медицині. Такі технології дозволяють створювати системи моніторингу стану здоров'я, які можуть збирати, обробляти і аналізувати велику кількість даних, що надходять у реальному часі від різноманітних сенсорів. Це відкриває нові можливості для віддаленого медичного обслуговування, оперативного реагування на відхилення у показниках пацієнтів, а також довготривалого спостереження за хронічними хворобами [1]. Ця робота фокусується на аналізі використання хмарних технологій та сервісів для обробки даних з медичних IoT пристроїв та надає всебічний огляд цієї задачі.

2. СИСТЕМИ НА БАЗІ ІОТ ПРИСТРОЇВ ТА ХТС В МЕДИЦИНІ

Системи, що займаються збором та обробкою медичних даних з IoT пристроїв зазвичай вирішують задачу моніторингу стану живого організму. Моніторинг стану живого організму – це процес безперервного спостереження за ключовими показниками життєдіяльності організму з метою виявлення відхилень від норми та реагування на них. Такі системи зазвичай складаються з наступного набору функціональних компонентів [1].

- Деякої кількості IoT пристроїв медичного спрямування для збору та попередньої обробки даних медичних показників.
- Сервера-отримувача даних, що отримує дані з IoT пристроїв по мережі, виконує попередню обробку та збереження.
- Сховища або бази для збереження даних.
- Вузла обробки даних, що виконує основний процес обробки даних та виявлення відхилень.
- Веб сервера, що дозволяє взаємодію з системою медичному персоналу.

Важливо зазначити, що системи моніторингу стану людського організму поділяються на два види за своїм кінцевим призначенням - системи критичного та некритичного моніторингу.

Системи критичного моніторингу призначені для використання у випадках, де будь-яка зміна показників стану людського організму може мати серйозні та швидкі наслідки для його життєдіяльності. Такі системи зазвичай сфокусовані на виявлені більш простих аномалій із якнайменшою затримкою в часі, що накладає певні обмеження на використання хмарних технологій та складних алгоритмів обробки даних. Також, додатковим важливим компонентом таких систем є надійний та швидкий механізм оповіщення медичного персоналу про виявлені відхилення [1].

Системи некритичного моніторингу призначені для використання у випадках, що не пов'язані із критичним станом людського організму та не потребують моментального втручання. Натомість, такі системи призначені для довготривалого спостереження з метою більш складного аналізу показників IoT пристроїв. Такі системи використовуються для наступних цілей [1].

- Нагляду за процесом реабілітації після оперативних втручань, та одужання після складних хвороб.
- Нагляду за людьми із хронічними або невиліковними хворобами.
- Моніторингу організму з метою запобіганню чи ранньому виявленню патологічних станів.
- Спостереженням за показниками життєдіяльності організму з немедичною метою.

3. ДАНІ З МЕДИЧНИХ IOT ПРИСТРОЇВ ТА АЛГОРИТМИ ЇХ ОБРОБКИ

Так як дані з медичних IoT пристроїв це набір показників з датчиків, що фіксуються у конкретний момент часу, цілком очевидною природною структурою даних для них є часовий ряд. Найбільш часто використовувані види медичних даних наведені у таблиці 1.

Таблиця 1. Види медичних показників та їх одиниці виміру

Назва	**Одиниці виміру**
Температура тіла	°C, °F
Частота дихання	n, Гц
Частота серцевих скорочень	n, Гц
Артеріальний тиск	мм рт. ст
Сатурація крові киснем	%
Рівень глюкози в крові	ммоль/л
Дані електрокардіограми	мВ
Дані електроенцефалограми	мВ
Дані електроміограми	мВ

Зважаючи на структуру таких даних, математичною основою алгоритмів обробки даних з медичних IoT пристроїв є теорія з аналізу та прогнозування часових рядів, обробки сигналів та прості математичні операції. Разом з цим, алгоритми адаптовані під вирішення конкретних медичних задач, тому також містять медичні аспекти. Популярними класичними алгоритмами є алгоритм Пан-Томпкінса та Вейвлет-трансформації для аналізу даних ЕКГ та виявлення аритмії, FFT для частотного аналізу даних електроенцефалограми [2].

Варто також зазначити, що останнім часом традиційні алгоритми все більш витісняються підходами штучного інтелекту та машинного навчання. Часто використовуються рекурентні нейронні мережі, логістична регресія та дерева рішень.

Наприклад, для тієї ж самої задачі по виявленню аритмії запропонований новий підхід, що базується на використанні CNN та SVM [3].

4. АНАЛІЗ МОЖЛИВОСТЕЙ ХМАРНИХ ТЕХНОЛОГІЙ ДЛЯ ОБРОБКИ ІНФОРМАЦІЇ З МЕДИЧНИХ IOT ПРИСТРОЇВ

Варто почати з того, що насправді задача з обробки медичних даних з IoT пристроїв є частковим випадком задачі з обробки великої кількості даних. Тому не зважаючи на те, що ця підзадача має свої особливості, які безперечно варті уваги, для неї справедливі більшість принципів та архітектурних рішень, що стосуються систем по обробці великих обсягів даних в цілому.

4.1. Інструменти для роботи з пристроями IoT

Вагомим компонентом системи для обробки даних з медичних IoT пристроїв є власне робота з IoT пристроями, під час якої відбувається отримання даних та імпорт їх у хмарну частину системи. Зважаючи на те, що управління IoT пристроями та забезпечення надійного отримання даних при масштабуванні може виявитись досить складною задачею, сучасні хмарні провайдери пропонують сервіси, що у вигляді PaaS або SaaS надають можливість спростити проведення таких операцій.

Основним елементом таких сервісів є MQTT-брокер разом з мережевим балансувальником навантажень, які забезпечують двосторонній зв'язок між пристроями та хмарною частиною системи та роблять його більш надійним під час значних навантажень. Більшість сервісів також включають набір елементів для управління IoT пристроями: сховище облікових даних пристроїв, механізми аутентифікації та авторизації, реєстр пристроїв, системи управління конфігурацією та оновленнями, а також набір правил для обробки даних й сповіщень [4].

У таблиці 2 представлено порівняння сервісів для роботи з IoT від найбільших хмарний провайдерів [5].

Таблиця 2. Порівняння сервісів управління IoT пристроями

Порівняльна характеристика	**GCP IoT Platform**	**AWS IoT Core**	**Azure IoT Hub**
Підтримувані протоколи комунікації	MQTT, HTTP	MQTT, HTTP, WebSockets	MQTT, HTTPS, AMQP, WebSockets
Управління пристроями	Базове управління, реєстрація та моніторинг пристроїв	Розширене управління, включаючи групування, моніторинг та оновлення прошивки	Розширене управління, включаючи реєстрацію, моніторинг та оновлення прошивки
Масштабованість	Висока	Висока	Висока
Підтримка Edge Computing	Cloud IoT Edge для обробки даних на пристроях	AWS IoT Greengrass для обробки даних на пристроях	Azure IoT Edge для обробки даних на пристроях

4.2. Інструменти для обробки даних

Ключовим компонентом систем, що розглядаються у цій роботі, є обробка та аналіз даних. Від його реалізації залежить правильність роботи всієї системи та коректність її

кінцевих результатів, тому важливо обирати такі інструменти для обробки даних, які максимально полегшать реалізацію та дозволять сфокусуватись на власне алгоритмі обробки. Якщо розглядати саме інструменти організації обробки даних у хмарних системах, то перед інженерами стоїть досить обширний їх вибір – хмарними провайдерами пропонується багато IaaS та PaaS рішень, подекуди спроєктованих саме для роботи з даними IoT.

Яскравим прикладом вище описаних сервісів є AWS IoT Analytics – PaaS інструмент, який автоматизує процеси збору, обробки, зберігання та аналізу даних від IoT-пристроїв. Він надає можливість фільтрувати, трансформувати та збагачувати дані після їх надходження, зберігати у сховищі даних з підтримкою часових рядів, та аналізувати дані за допомогою SQL запитів, виконання будь-якого коду та моделей штучного інтелекту. Схожий за своїм функціоналом сервіс є і в Azure – Azure Time Series Insights. Він також надає можливість аналізувати та обробляти дані з IoT пристроїв за допомогою На відміну від попередніх хмарних провайдерів, GCP не має в арсеналі аналогічного сервісу, що спроєктований спеціально для обробки даних з IoT пристроїв [4, 5].

Також, окрім спеціалізованих інструментів, хмарні провайдери надають широкий вибір сервісів для обробки даних будь-яких видів, які можна налаштувати для конкретної задачі. Варто звернути увагу на такі сервіси, як Amazon EMR, GCP Dataproc та Azure Databricks для обробки даних використовуючи фреймворки Hadoop та Spark; Amazon SageMaker, VertexAI, Bigquery ML, Cloud Dataflow та Azure Machine Learning для обробки даних за допомогою методів машинного навчання; AWS Lambda, GCP Cloud Run, Azure Functions для serverless обробки даних [6].

5. ТИПОВА АРХІТЕКТУРА СИСТЕМИ ДЛЯ ОБРОБКИ ДАНИХ З МЕДИЧНИХ ІОТ ПРИСТРОЇВ

Після загального аналізу можливостей хмарних систем, варто розглянути архітектуру типової системи для обробки даних з медичних IoT пристроїв. Зазвичай, така система складається з двох основних частин – клієнтської та хмарної.

При цьому, клієнтською частиною системи є набір N датчиків, для виміру N показників життєдіяльності людського організму, N контролерів для забезпечення функціонування цих датчиків та один або декілька контролерів передачі показників у хмарну частину. Хмарною частиною є набір віртуальних або фізичних обчислювальних вузлів для виконання обробки, зберігання, аналізу та відображення даних показників, що були отримані від клієнтської частини [6].

На рисунку 1 зображено вище описану архітектуру системи за допомогою UML діаграми компонентів.

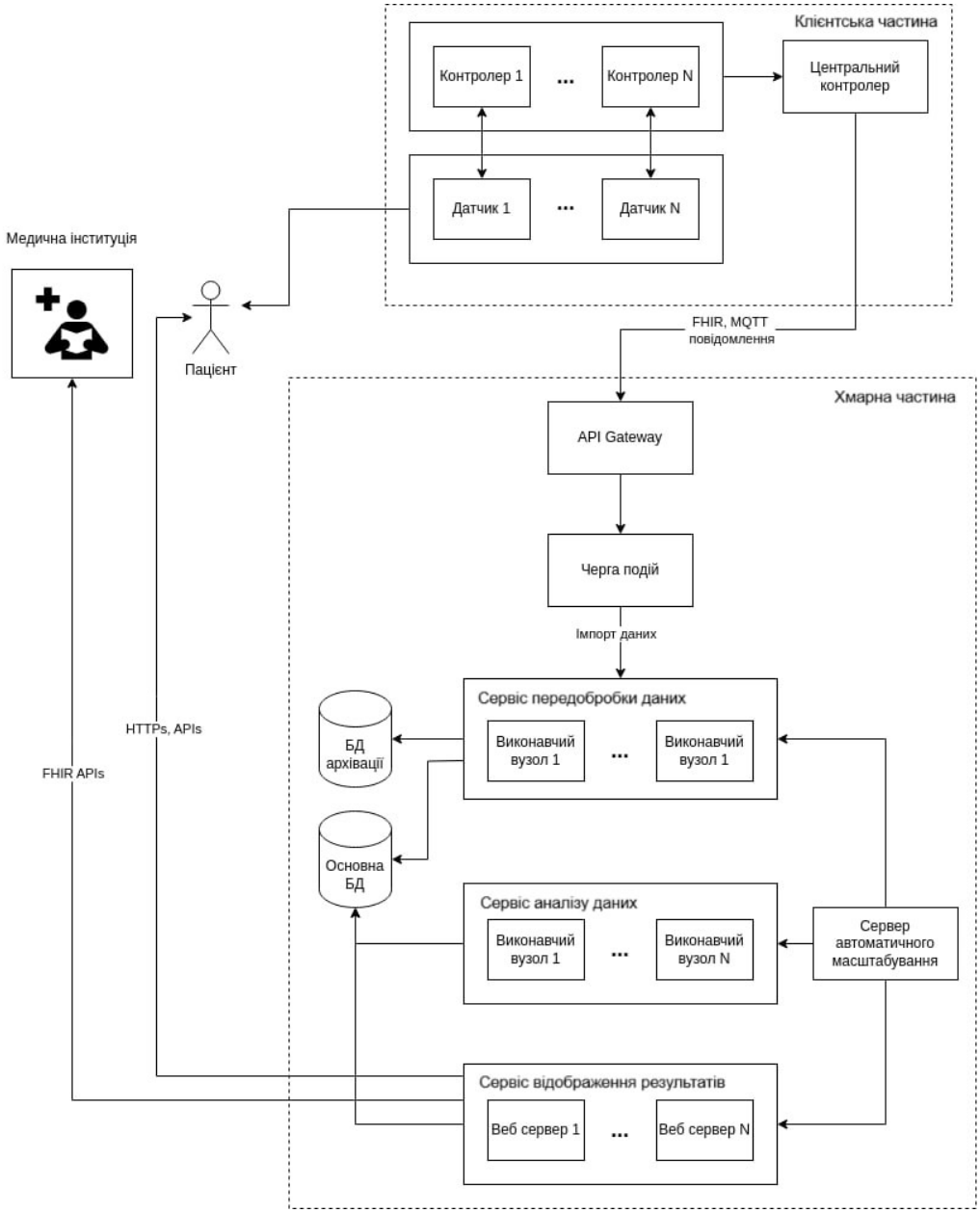


Рисунок 1. Типова архітектура системи для обробки медичних даних з IoT пристроїв

Як видно, хмарна частина системи має 3 основних сервіси – передобробки даних, аналізу даних та відображення результатів. Черга подій та база даних є службовими компонентами, а сервер автоматичного масштабування та API Gateway – інфраструктурними компонентами. При цьому набір даних, що збирається, як і реалізація сервісів обробки даних залежить від конкретної прикладної задачі. Також, як видно із рисунку 1, комунікація між клієнтською та хмарною частиною відбувається по протоколу MQTT, що передає дані у форматі FHIR [6].

6. ВИСНОВКИ

У цій роботі було розглянуто використання хмарних технологій та сервісів для обробки даних з медичних IoT пристроїв. Спочатку було коротко описано використання таких систем у медицині, потім – описано основні типи даних, що вимірюються медичними IoT пристроями та алгоритми, що їх обробляють.

Далі, було проаналізовано можливості сучасних хмарних технологій для реалізації таких систем. Було проведено порівняльний аналіз сервісів з управління IoT пристроями різних хмарних провайдерів та описано можливості сервісів для обробки даних. У результаті аналізу, можна підсумувати, що хмарна платформа AWS надає найбільш широкий та гнучкий функціонал для проєктування таких систем, а тому може бути рекомендована до використання.

У останній частині роботи було представлено та описано типову архітектуру системи для обробки даних з медичних IoT пристроїв. Ця архітектура ілюструє типову організацію таких систем використовуючи загальні концепції хмарних технологій та сервісів, без використання конкретних технологій чи фокусу на конкретну задачу по обробці даних. Використовуючи цю шаблонну архітектуру у подальшому може бути спроєктована конкретна система по обробці даних, що використовує сервіси описані у третьому розділі роботи.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Malasinghe L. P., Ramzan N., Dahal K. Remote patient monitoring: a comprehensive study. Journal of Ambient Intelligence and Humanized Computing. 2017. Т. 10, № 1. С. 57–76. URL: https://doi.org/10.1007/s12652-017-0598-x
2. Contributors to Wikimedia projects. Pan–Tompkins algorithm - Wikipedia. *Wikipedia, the free encyclopedia*. URL: https://en.wikipedia.org/wiki/Pan–Tompkins_algorithm.
3. Detection of arrhythmia from electrocardiogram signals using a novel gaussian assisted signal smoothing and pattern recognition / A. Chandrasekar та ін. *Biomedical Signal Processing and Control*. 2022. Т. 73. С. 103469. URL: https://doi.org/10.1016/j.bspc.2021.103469
4. Agarwal P., Alam M. Investigating IoT Middleware Platforms for Smart Application Development. *Jamia Millia Islamia*. 2019. URL: https://arxiv.org/abs/1810.12292.
5. The IoT cloud: Microsoft Azure vs. AWS vs. Google Cloud. *IoT Analytics*. URL: https://iot-analytics.com/iot-cloud/.
6. Addressing Health Equity through Remote Patient Monitoring and Continuity of Care | Amazon Web Services. Amazon Web Services. URL: https://aws.amazon.com/blogs/industries/addressing-health-equity-through-remote-patient-monitoring-and-continuity-of-care/.

ДОСЛІДЖЕННЯ ВИКОРИСТАННЯ ДИФУЗІЙНИХ МОДЕЛЕЙ ДЛЯ ЗБАГАЧЕННЯ І БАЛАНСУВАННЯ ТРЕНУВАЛЬНИХ НАБОРІВ ДАНИХ

Наумчик М.С.[1], Кислий Р.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] Naumchyk.Maksym@lll.kpi.ua, [2] kyslyi.roman@lll.kpi.ua

Метою даного дослідження є оцінка використання дифузійних моделей для збагачення даних та порівняння ефективності навченої моделі на оригінальному датасеті та збагаченому згенерованими даними. Цінність дослідження полягає у можливості покращування навчання моделі, особливо в сценаріях, де даних мало або вони незбалансовані. Також дифузійні моделі особливо ефективні для доповнення наборів даних у завданнях, де збір даних є дорогим, тривалим або складним.

Ключові слова: комп'ютерний зір, виявлення об'єктів, дифузійні моделі, машинне навчання.

1. ВСТУП

За останні роки моделі машинного навчання досягли значних успіхів у різних сферах, від комп'ютерного зору до обробки природніх мов. Однак однією з найпоширеніших проблем при навчанні цих моделей є потреба у великих наборах високоякісних даних. У багатьох реальних застосуваннях збір і анотування достатньої кількості даних може бути дорогим, трудомістким або непрактичним. Збагачення (або доповнення) даних, метод, який штучно збільшує різноманітність навчальних даних, став потужним рішенням цієї проблеми. Традиційно доповнення даних в обробці зображень передбачає такі перетворення, як обертання, масштабування, перевертання і коригування кольору, але ці методи можуть бути обмежені в їх здатності створювати справді нові дані, які відображають складність реальних середовищ.

Одним із варіантів збагачення даних який може перебороти ці обмеження, це дифузійні моделі – нова розробка в генеративному моделюванні. Дифузійні моделі, які спочатку були створені для створення високоякісних, реалістичних зображень, пропонують новий підхід для доповнення даних. Використовуючи дифузійні моделі, дослідники можуть створювати синтетичні, але дуже реалістичні дані, які збагачують навчальні набори, оминаючи обмеження традиційних методів збагачення даних.

Оскільки сфера машинного навчання продовжує розширювати свої межі, дифузійні моделі є перспективним інструментом для підвищення надійності та продуктивності моделей за рахунок збагачення навчальних наборів даних високоякісними синтетичними даними.

2. ПЕРЕВАГИ ВИКОРИСТАННЯ ДИФУЗІЙНИХ МОДЕЛЕЙ ДЛЯ ЗБАГАЧЕННЯ ТРЕНУВАЛЬНИХ ДАНИХ

Використання дифузійних моделей для доповнення даних має кілька переваг. Вони чудово підходять для створення синтетичних даних, які точно імітують складні закономірності, що зустрічаються в реальних наборах даних, покращуючи багатість

навчальних наборів. Крім того, вони особливо цінні для усунення дизбалансу класів, збереження дрібних деталей, покращення узагальнення, зменшення вартості збору даних.

2.1 Вирішення проблеми дисбалансу даних

Дисбаланс даних є значною проблемою в багатьох задачах машинного навчання. У незбалансованих наборах даних певні класи недостатньо представлені порівняно з іншими, що призводить до того, що моделі зміщуються в бік домінуючих класів. Це особливо проблематично в таких областях, як медична діагностика або виявлення аномалій, де рідкісні, але критичні випадки недостатньо представлені, але мають велике значення. Наприклад, у медичній візуалізації може бути переважна кількість нормальних знімків порівняно зі знімками, що демонструють рідкісні стани. В автономному водінні рідкісні події, такі як аварії або незвичні умови освітлення, часто недостатньо представлені в наборах даних.

Традиційні методи усунення класового дисбалансу часто передбачають надмірну вибірку класу меншини або недостатню вибірку класу більшості. Однак ці методи мають обмеження. Надмірна вибірка може призвести до надмірного пристосування моделі до одних і тих самих екземплярів меншості, тоді як недостатня вибірка може призвести до втрати цінної інформації від найбільш представленого класу. Далі буде розглянуто як дифузійні моделі пропонують усунення дисбалансу даних.

Дифузійні моделі можна навчити генерувати абсолютно нові зображення для недостатньо представлених класів, наприклад, зображення рідкісних захворювань, незвичайних умов навколишнього середовища або рідкісних об'єктів. На відміну від традиційних методів збагачення даних, які просто дублюють існуючі зображення, дифузійні моделі створюють реалістичні та різноманітні синтетичні зображення, які збагачують набір даних. Це призводить до більш збалансованого впливу на всі класи під час навчання.

У багатьох випадках зображення міноритарних класів містять тонкі або складні ознаки, які є важливими для правильної класифікації. Наприклад, рідкісні захворювання на медичних зображеннях можуть мати дрібнозернисті аномалії, або певні дефекти при промисловому контролі може бути важко виявити. Дифузійні моделі здатні генерувати синтетичні зображення, які зберігають ці дрібні деталі, гарантуючи, що модель навчається на критичних особливостях класу меншості, а не на загальних перетвореннях або спрощених зображеннях з вибірки (sampled images).

Таким чином, збагачуючи набір даних високоякісними зразками з класу меншин, дифузійні моделі зменшують класове упередження, допомагаючи моделі вивчати більш збалансовані межі прийняття рішень. Це призводить до покращення точності, достовірності та запам'ятовування, особливо для недостатньо представлених класів.

2.2 Покращення генералізації

Генералізація є одним з найважливіших аспектів машинного навчання, оскільки воно визначає, наскільки добре модель працює з невидимими даними за межами навчальної вибірки. Для даних зображень досягнення сильного узагальнення є особливо складним завданням через складну та різноманітну природу візуальних патернів. Моделі, навчені на обмежених або однорідних наборах даних, схильні до надмірного пристосування - навчання розпізнаванню специфічних особливостей на навчальних зображеннях, але не здатні застосувати ці вивчені шаблони до нових, різноманітних або складних прикладів.

Дифузійні моделі покращують узагальнюючі можливості моделей, що працюють з даними зображень, генеруючи реалістичні та різноманітні синтетичні зображення, які розширюють діапазон шаблонів, структур і сценаріїв, з якими стикається модель під час навчання.

Справа в тім, що реальні дані зображень часто набагато різноманітніші, ніж ті, що містяться в навчальних наборах даних. Зображення можуть відрізнятися через зміну освітлення, перспективи, роздільної здатності або фонового шуму, з якими модель повинна навчитися справлятися. Дифузійні моделі можуть генерувати синтетичні зображення, які включають ці види мінливості, допомагаючи моделі навчитися узагальнювати широкий спектр реальних умов. Це особливо корисно в таких областях, як автономне водіння, де система повинна функціонувати за різних погодних умов і освітлення, або в медичній візуалізації, де різні пристрої і протоколи візуалізації можуть впливати на зовнішній вигляд даних.

2.3 Зменшення витрат на збір даних

Збір великих і різноманітних наборів даних зображень часто є одним з найбільш ресурсоємних аспектів навчання моделей машинного навчання. У багатьох сферах отримання достатньої кількості високоякісних маркованих зображень може бути дорогим, трудомістким і логістично складним завданням. Дифузійні моделі, як потужні інструменти генерації, пропонують рішення, дозволяючи генерувати реалістичні, синтетичні дані зображень.

У таких сферах, як автономне водіння, медична візуалізація та промисловий контроль, збір достатньої кількості різноманітних зображень часто вимагає дорогого обладнання, спеціалізованого середовища та тривалих процесів налаштування. Наприклад, збір медичних зображень вимагає дорогого обладнання, співпраці з медичними установами і часто згоди пацієнта. За допомогою дифузійних моделей значну частину набору даних можна доповнити або повністю згенерувати, що зменшує потребу в обширному зборі реальних даних. Замість того, щоб отримувати нові зображення за допомогою дорогих засобів, можна генерувати синтетичні дані за невелику частину вартості.

Крім того, маркування даних зображень – ще один суттєвий фактор витрат на збір даних. У таких завданнях, як медична візуалізація або супутникові знімки, маркування вимагає знання предметної області, що означає оплату праці фахівців, таких як лікарі або геопросторові аналітики, для коментування наборів даних. Дифузійні моделі можуть генерувати мічені синтетичні зображення, прив'язуючи їх до певних категорій або атрибутів. Наприклад, дифузійна модель, навчена генерувати рентгенівські знімки легень, може створювати зображення, позначені як «нормальні» або «аномальні», таким чином зменшуючи потребу в ручному маркуванні кожного окремого зображення. Автоматизоване генерування маркованих зображень допомагає скоротити високі витрати, пов'язані з анотуванням експертних даних.

3. ЗБАГАЧЕННЯ НАБОРУ ДАНИХ ЗА ДОПОМОГОЮ ДИФУЗІЙНИХ МОДЕЛЕЙ

Розглянемо приклад генерації даних за допомогою дифузійної моделі. Оберемо одну з найбільш популярних моделей – Stable Diffusion v1.5. Це модель латентної дифузії, ініціалізована з попередньої контрольної точки, і додатково налаштована протягом 595 тис. кроків на зображеннях 512x512 [1].

Також ми будемо використовувати підхід перетворення зображення до зображення, де на додаток до інструкції (як в класичних моделях дифузії) також можна передати початкове зображення як відправну точку для процесу дифузії. Початкове зображення кодується в латентний простір і до нього додається шум. Потім модель латентної дифузії бере інструкцію і зашумлене латентне зображення, прогнозує доданий шум і видаляє передбачений шум з початкового латентного зображення, щоб отримати нове латентне зображення. Нарешті, декодер декодує нове латентне зображення назад у зображення [2].

Щоб використовувати цей пайплайн для обробки зображень, спочатку потрібно підготувати початкове зображення, яке буде передано до пайплайну. Потім необхіно передати інструкцію для моделі і зображення підготовлене раніше, щоб згенерувати нове зображення.

Рисунок 1. Приклад генерації зображення (справа) з інструкцією та початковим зображенням (зліва)

В якості набіру даних було обрано датасет на якому буде проводитися дослідження: DAC-SDC 2022 (Design Automation Conference System Design Contest 2022 Dataset) [3]. Цей датасет використовувався в змаганні 2022 року, метою якого було розробити та впровадити нові алгоритми, засновані на виявленні об'єктів на зображеннях, отриманих з безпілотних літальних апаратів (БПЛА).

З усіх наявних класів, оберемо три найбільш репрезентативних: person, car, riding. Візьмемо зменшену версію датасету з 700 зображеннями де для кожного класу проблизно належать по 200 зображень. Натренуємо одну з найбільш популярних state of the art моделей YOLO11 [4] на оригінальному датасеті:

Таблиця 1. Метрики ефективості YOLO11 на початковому датасеті

Клас	mAP50	mAP50-95
car	0.936	0.555
person	0.944	0.537
riding	0.872	0.567

Далі оберемо клас car і для кожного зображення з класу згенеруємо нове зображення за допомогою дифузійної моделі. Отримаємо 218 згенерованих зображень. Проте перед тим як об’єднувати ці зображення з нашим початковим датасетом необхідно спочатку позначити для моделі де саме на зображенні знаходиться обраний клас (car). Для маркування скористаємося сервісом roboflow [5], який значно полегшує цей процес. В результаті об'єднуємо отримані згенеровані та помарковані дані з початковим датасетом та тренуємо модель Yolo на збагаченому датасеті:

Таблиця 2. Метрики ефективості YOLO11 на збагаченому датасеті

Клас	mAP50	mAP50-95
car	0.871	0.566
person	0.946	0.63
riding	0.896	0.545

В результаті бачимо покращення для класу car в метриці mAP50-95 проте і погіршення в метриці mAP50. Для покращення результатів пропонується провести дослідження впливу інших параметрів моделі дифузії на результати та генерувати зображення в співвідношенні 1:2

або 1:3 згенерованих зображень. Також в подальшому планується генерувати зображення для всіх класів.

4. ВИСНОВКИ

В роботі було досліджено використання дифузійних моделей для збагачення даних, підкреслено їхній трансформаційний потенціал у створенні високоякісних, різноманітних синтетичних даних. Дифузійні моделі мають явні переваги над традиційними методами доповнення, створюючи абсолютно нові вибірки даних з випадкового шуму, що дозволяє створювати багатші та різноманітніші навчальні набори даних.

Також було описано ключові переваги використання дифузійних моделей для доповнення даних, такі як їх здатність вирішувати проблеми незбалансованості даних, покращувати узагальнення моделі та зменшувати витрати, пов'язані зі збором даних. Генеруючи синтетичні дані, які враховують складні варіації освітлення, кута і текстури, дифузійні моделі можуть підвищити надійність і продуктивність моделей машинного навчання.

Нарешті, було запропоновано комплексний план дослідження для порівняння результатів обраної моделі на початковому датасеті та датасеті збагаченому даними з дифузійної моделі. Для покращення результатів було запропоновано провести дослідження впливу параметрів моделі дифузії на результати та генерувати зображення в співвідношенні 1:2 або 1:3 згенерованих зображень. Також було запропоновано генерувати зображення для всіх класів датасету.

Дослідження має на меті надати цінну інформацію про те, як дифузійні моделі та передові інструменти генерації синтетичних даних можуть бути використані для покращення завдань машинного навчання.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Hugging Face Diffusers [Електронний ресурс]. — Режим доступу до ресурсу: https://huggingface.co/docs/diffusers/en/using-diffusers/img2img (дата звернення: 23.10.2024)
2. Hugging Face Image-to-Image [Електронний ресурс]. — Режим доступу до ресурсу: https://huggingface.co/tasks/image-to-image (дата звернення: 23.10.2024)
3. Xu, Xiaowei & Zhang, Xinyi & Yu, Bei & Rowen, Christopher & Hu, Jingtong & Shi, Yiyu. (2019). DAC-SDC Low Power Object Detection Challenge for UAV Applications. 10.1109/TPAMI.2019.2932429.
4. Ultralytics YOLO11 source code [Електронний ресурс]. — Режим доступу до ресурсу: https://github.com/ultralytics/ultralytics (дата звернення: 15.10.2024)
5. Train a YOLOv11 Object Detection Model on a Custom Dataset [Електронний ресурс]. — Режим доступу до ресурсу: https://blog.roboflow.com/yolov11-how-to-train-custom-data/ (дата звернення: 23.10.2024)

СЕНСОРНІ МЕРЕЖІ ДЛЯ КОНТРОЛЮ ЗА ЖИТТЄДІЯЛЬНІСТЮ ЛЮДИНИ

Немировська В.О.[1], Гіоргізова-Гай В.Ш.[2]

Київський політехнічний інститут імені Ігоря Сікорського, Київ, Україна

[1] nemivika@gmail.com, [2] high.victoria@lll.kpi.ua

Метою дослідження є проєктування та розробка натільної сенсорної мережі для моніторингу життєвих показників людини в екстремальних умовах з можливістю застосування в Україні. Результатом є спроєктована система, а також її прототип на стадії першої ітерації розробки, що охоплює основний необхідний функціонал. Наукова новизна полягає у розробці натільної сенсорної мережі спрямовану саме на застосування у військовій сфері, дослідницьких експедиціях та екстремальному туризмі, із застосуванням технологій сенсорних мереж, туманних обчислень та супутникового зв'язку.

Ключові слова: натільна сенсорна мережа, моніторинг життєдіяльності, Bluetooth LE, туманні обчислення, екстремальні умови та військове застосування.

1. ВСТУП

Контроль за життєвими показниками людини в екстремальних умовах є актуальним завданням у зв'язку з необхідністю підвищення рівня безпеки військових, що перебувають у зонах бойових дій в Україні, а також дослідників та екстремальних туристів, що діють у складних та небезпечних середовищах. Наприклад, у військових операціях сенсорні мережі здатні збільшити шанси на виживання військовослужбовців, учасників дослідницьких експедицій, що діють у важкодоступних зонах (зонах відчуження, пустелях, високогірних районах тощо), забезпечуючи завчасне виявлення критичних ситуацій та своєчасне надання медичної допомоги. Застосування сенсорних мереж у даній сфері також може суттєво підвищити ефективність операцій, дозволяючи центрам обробки даних (ЦОД) оперативно отримувати загальну інформацію про стан особового складу та приймати рішення на основі зібраних даних.

Враховуючи специфіку поставленої задачі, система має відповідати таким вимогам: автономність та портативність, що включає в себе зручність носіння, енергоефективність, форм-фактор та вагу; безпека даних, що передбачає ідентифікацію датчиків та шифрування даних при передаванні в Інтернет; надійність, тобто забезпечення працездатності при виході з ладу певних компонентів, можливість перекомплектації та покриття мережі; економічність, що включає вибір оптимальних компонентів та простоту реалізації системи.

На даний час не було знайдено прямих аналогів систем, які б комплексно вирішували ці завдання використовуючи натільні сенсорні мережі та відповідали б основним поставленим вимогам, тому запропоновано власну розробку натільної сенсорної мережі, яку описано в даній статті.

2. АНАЛІЗ ПУБЛІКАЦІЙ

2.1. Огляд аналогів

Існує багато досліджень та реалізацій натільних сенсорних мереж для контролю життєво важливих показників людини, представлених в наукових публікаціях.

Більшість систем призначені для спостереження за станом хворих та здорових людей у стаціонарних умовах міста, що обумовлює склад компонентів цих систем та особливості їх реалізації. Серед систем такого призначення можна виділити сенсори, одяг, ремені і програмне забезпечення Zephyr та OmniSense від Medtronic [1] для детального моніторингу фізіологічних показників під час тренувань професійних спортсменів та військових; смарт одяг від Hexoskin [2] для відстеження кардіореспіраторної функції, сну та фізичної активності; розробка Гарвардського університету CodeBlue [3] для відстеження медичних показників пацієнтів в режимі реального часу та оперативного реагування на зміни у їх стані. Явним недоліком останньої системи є передача даних у відкритому вигляді.

Існують також потужні розвинуті системи для застосування в екстремальних умовах. Наприклад, медичний гнучкий пластир PATCH (Persistent Access to Tactical Casualty Health) від Vivonics [4] може приймати, зберігати та розповсюджувати інформацію про виконану допомогу пацієнту або бійцю і призначений для збору, обміну та керування даними про поранених у бойових діях, для вирішення помилок у зв'язку між медиками та іншими особами, які займаються наданням допомоги. Натільна сенсорна мережа Lifeguard розроблена групою дослідників NASA та Стенфордського університету для застосування в умовах космічного середовища та екстремальних наземних умовах [5, 6]. Ці системи є досить складними, багатофункціональними та коштовними.

Тому незважаючи на значну кількість розглянутих систем, серед них нема таких, які б задовольняли вимогам застосування в суворих, небезпечних умовах та поєднували в собі надійність, економічність, компактність і автономність для визначення місцезнаходження і своєчасного надання допомоги людині в критичному стані. І це зумовлює актуальність подальших досліджень, представлених у даній роботі.

2.2. Методика визначення критичного стану

Для визначення критичного стану організму людини необхідно використовувати вже перевірену та визнану систему оцінювання стану. Шкала оцінки стану організму людини для раннього попередження NEWS (National Early Warning Score) була розроблена Королівським коледжем лікарів (The Royal College of Physicians) наприкінці 1990-х [7], а в 2017 р. було опубліковано оновлену систему оцінювання NEWS 2. Дана система впроваджена на національному рівні в Великій Британії, а також використовується в багатьох інших країнах світу. В ній для кожної життєво важливої ознаки визначається нормальний діапазон – діапазон значень відповідної характеристики організму здорової людини в нормальному стані. Значення за межами цього діапазону оцінюються відповідно до величини відхилення від нормального. Балове зважування відображає тяжкість фізіологічного стану людини. Таблиці оцінювання NEWS 2 наведено на рис. 1 та рис. 2 [8].

Physiological parameter	Score 3	2	1	0	1	2	3
Respiration rate (per minute)	≤8		9–11	12–20		21–24	≥25
SpO_2 Scale 1 (%)	≤91	92–93	94–95	≥96			
SpO_2 Scale 2 (%)	≤83	84–85	86–87	88–92 ≥93 on air	93–94 on oxygen	95–96 on oxygen	≥97 on oxygen
Air or oxygen?		Oxygen		Air			
Systolic blood pressure (mmHg)	≤90	91–100	101–110	111–219			≥220
Pulse (per minute)	≤40		41–50	51–90	91–110	111–130	≥131
Consciousness				Alert			CVPU
Temperature (°C)	≤35.0		35.1–36.0	36.1–38.0	38.1–39.0	≥39.1	

Рисунок 1. Нормальний діапазон життєвих показників та оцінювання відхилень від нього за NEWS 2

NEW score	Clinical risk	Response
Aggregate score 0–4	Low	Ward-based response
Red score Score of 3 in any individual parameter	Low–medium	Urgent ward-based response*
Aggregate score 5–6	Medium	Key threshold for urgent response*
Aggregate score 7 or more	High	Urgent or emergency response**

Рисунок 2. Балове оцінювання фізіологічного стану людини за NEWS 2

Згідно з наведених таблиць серед важливих життєвих показників виділяють частоту дихання, рівень насиченості крові киснем (SpO_2), систолічний кров'яний тиск, частота серцевих скорочень за хвилину та температуру тіла. Моніторинг зазначених показників дозволяє оперативно оцінити стан здоров'я людини і своєчасно виявляти критичні відхилення, що необхідно взяти до уваги при подальшому проєктуванні.

3. ПРОЄКТУВАННЯ НАТІЛЬНОЇ СЕНСОРНОЇ МЕРЕЖІ

3.1. Проєктування логічної топології

Проєктування логічної топології натільної сенсорної мережі потребує ретельного планування та уваги до деталей, обумовлених специфікою потреб користувачів. Тому основними задачами, що постають при проєктуванні є забезпечення успішного прийняття пакетів зібраних даних для подальшого аналізу та визначення критичного стану людини, мінімізація трафіку та втрати пакетів, забезпечення енергоефективності системи, надійного захисту сенсорної мережі, стабільного з'єднання з Інтернетом та зниження обчислювального навантаження на кінцевому вузлі – ЦОД.

Всі ці задачі вирішуються в рамках концепції туманних обчислень (Fog Computing), яка передбачає проведення попередньої обробки даних якомога ближче до джерел їх надходження перед пересиланням на хмарний сервер. Попередня обробка даних на краю сенсорної мережі включає зокрема, агрегацію, форматування та зберігання зібраних даних, виконання обчислень з метою визначення інформації, яку варто передавати на хмарний ЦОД.

Загальну топологічну схему сенсорної мережі наведено на рис. 3. Показники з сенсорних вузлів через локальну мережу надходять до IoT-шлюзу. На шлюзі дані агрегуються,

структуруються та проводиться їх аналіз з метою визначення критичного стану людини. Після фіксації критичного стану людини шлюз формує повідомлення, яке включає відповідні показники датчиків і геолокації та відправляє його в мережу в зашифрованому вигляді. Передача повідомлення повторюється з певною періодичністю поки не буде отримано зворотній сигнал з сервера. Це здійснюється з метою підвищення надійності прийому даних та спрощення подальшого пошуку потерпілого. За допомогою модулів супутникового зв'язку оброблені дані передаються до найближчого супутника. Супутники, в свою чергу, передають отримані дані до відповідного наземного терміналу, який може передати дані на ЦОД. До сервера у ЦОД можна виконати запит на отримання доступу до перегляду даних про осіб, за якими ведеться спостереження, і в разі виявлення критичного стану особи, визначити її місцезнаходження та виконати відповідні дії щодо порятунку людини.

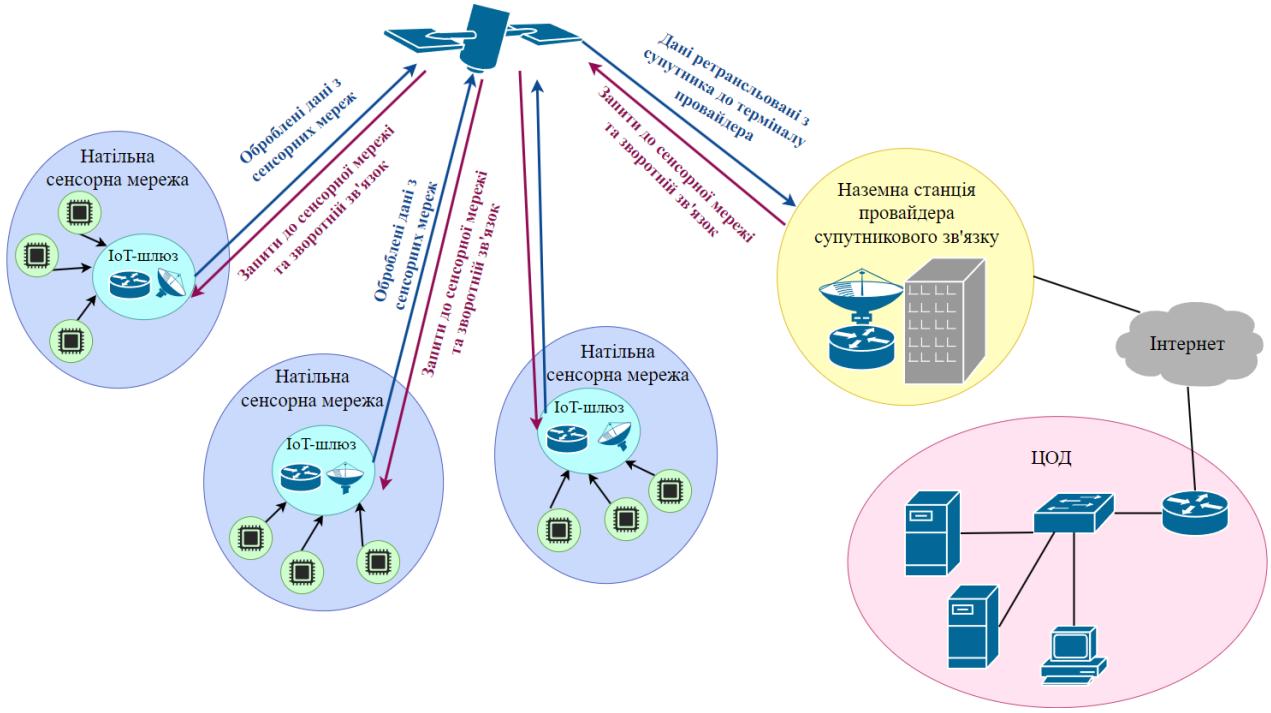


Рисунок 3. Логічна топологія сенсорної мережі

Для натільної мережі було обрано мережеву технологію Bluetooth LE. Перевагою технології перед іншими технологіями малого радіусу дії, таких як ZigBee, Z-Wave, Thread, є менше споживання електроенергії, широка підтримка вендорами, низька ціна модулів та чіпсетів і відкритість стандарту.

Для передачі даних по мережі Інтернет від аналізуючого модуля-шлюзу до ЦОД, обрано технологію супутникового зв'язку оскільки, по-перше, супутникові мережі забезпечують глобальне покриття навіть у віддалених та важкодоступних місцях. В експедиціях та екстремальному туризмі люди зазвичай перебувають саме в таких регіонах, а в зонах бойових дій є висока ймовірність пошкоджень наземної інфраструктури. По-друге, супутникові мережі забезпечують високу надійність передачі даних. Вони є стійкими до глушіння, перехоплення та фізичних пошкоджень, що є особливо важливим у військовій галузі. Провайдером супутникового зв'язку обрано Iridium, оскільки він надає глобальне покриття, а супутники знаходяться на низькій навколоземній орбіті, що забезпечує високу якість і швидкість зв'язку. Широкий спектр послуг та тарифів Iridium дозволяє обирати варіанти під конкретні завдання, оптимізуючи його вартість.

3.2. Вибір компонентів шлюзу

Виходячи з побудованої логічної топології, натільна мережа має два типи компонентів, а саме, вузол аналізуючого модуля-шлюзу та сенсорні вузли.

Для реалізації шлюзу пропонується застосувати модуль супутникового зв'язку RockBLOCK 9603N [9], що розроблений на базі модуля Iridium 9603 з додатково інтегрованими інтерфейсами для підключення зовнішніх пристроїв. Для визначення місцезнаходження людини пропонується застосувати GPS модуль Matek M10Q-5883 [10], що має UART інтерфейс для отримання даних географічних координат (GNSS). В якості джерела живлення для шлюзу обрано літієво-іонні акумулятори формату 18650 на 3500 мА·год з функціоналом зарядки літієвого акумулятора. Для виконання аналізу та комунікації з сенсорами пропонується обрати мікроконтролер ESP32-C3 SuperMini, який підтримує Bluetooth LE, має достатню обчислювальну потужність та об'єм пам'яті, чотири режими живлення, а також вбудовані апаратні криптографічні акселератори [11]. Для здійснення конфігурацій користувачем додано кнопку. Принципову схему підключення шлюзу наведено на рис. 4.

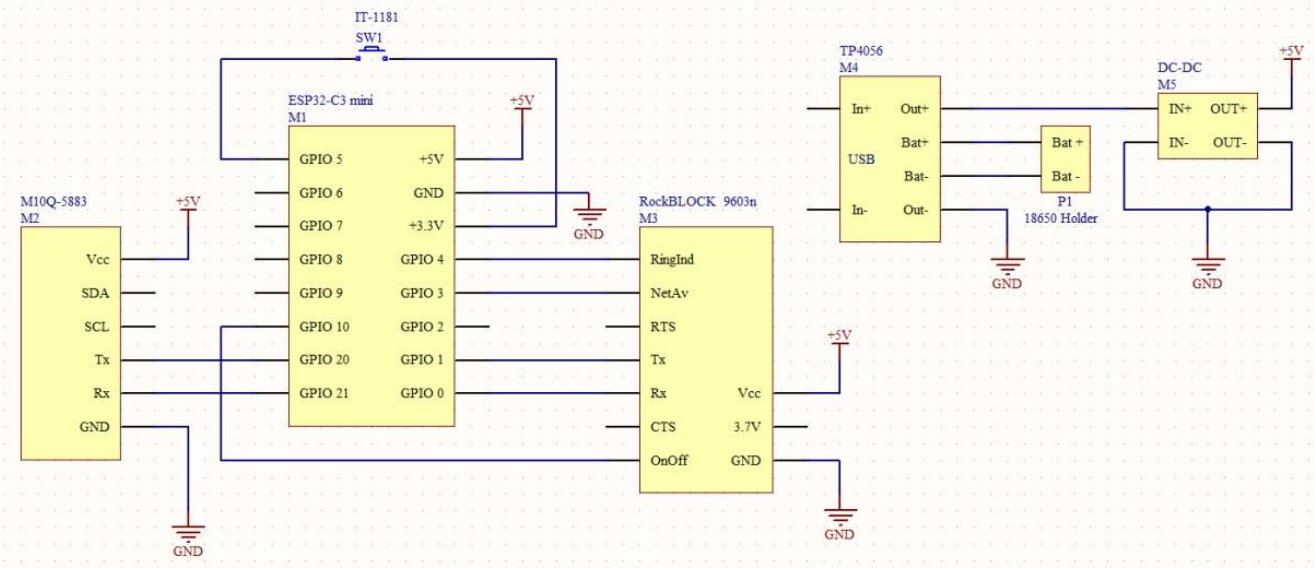


Рисунок 4. Принципова схема аналізуючого модуля-шлюза

3.3. Вибір компонентів сенсорів

На основі раніше зазначеної системи NEWS 2, для визначення критичного фізіологічного стану людини в розроблюваному прототипі натільної сенсорної мережі обрано такі показники як: рівень насиченості крові киснем, систолічний кров'яний тиск, частота серцевих скорочень за хвилину (пульс) та температуру тіла.

Пропонується набір датчиків, який включає в себе: пульсоксиметр (для вимірювання пульс та рівень насичення киснем), термометр (вимірювання температури тіла людини) та гіроскоп з акселерометром (для відстеження рухів тіла, його положення в просторі). З цього переліку в проекті вилучено тонометр, який мав би фіксувати значення систолічного тиску, проте він може бути незручним в умовах постійного руху.

Будова сенсорів для мережі є подібною. На рис. 5 як приклад показано принципову схему сенсора вимірювання температури. Він складається з датчика температури MAX30205 [12], мікроконтролера ESP32-C3 SuperMini, літієвої батарейки-таблетки CR3032 та кнопки.

Аналогічно сенсор пульсоксиметра складається з датчика пульсоксиметра MAX30102, мікроконтролера ESP32-C3, літієвої батарейки-таблетки CR3032 та кнопки. Для сенсора

визначення положення тіла використовується плата з інерційним вимірювальним модулем BMI160, і так само ESP32-C3, батарейка CR3032 та кнопка.

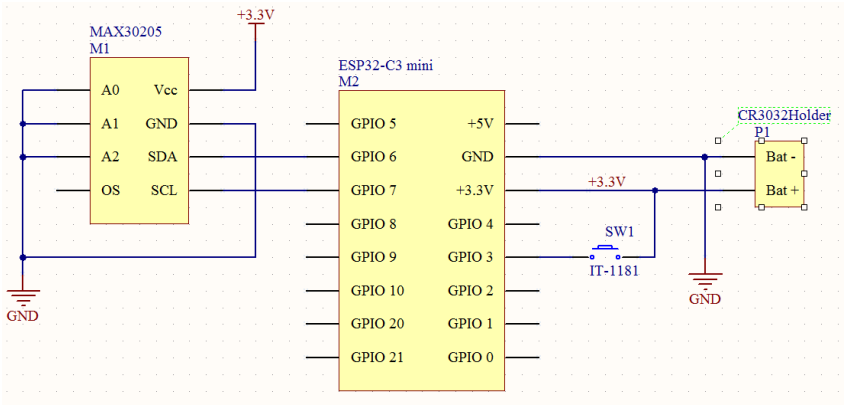


Рисунок 5. Принципова схема сенсора вимірювання температури

3.4. Загальна робота системи

Для підготовки системи до роботи потрібно виконати окрему процедуру реєстрації сенсорів на шлюзі, а шлюзу на сенсорах. Після успішного виконання реєстрації в список зареєстрованих пристроїв сенсора додається шлюз, куди він надсилатиме виміряні показники, а в список шлюзу додаються сенсори, з яких він буде приймати дані. Робота мережі синхронізується за таймером. Всі пристрої можуть перебувати в двох режимах: приймання/передача даних та глибокий сон. Після виконання реєстрації, сенсор переходить в режим періодичного направленого рекламування, а шлюз – в режим періодичного сканування радіо каналів для отримання пакетів від сенсорів зі списку зареєстрованих. При отриманні пакету шлюз записує передані дані в енергонезалежну пам’ять. Перед тим як заснути, контролер, на базі отриманих даних, виконує аналіз критичності стану людини та у разі його виявлення формує повідомлення, що надалі буде передаватися на ЦОД.

Розроблений алгоритм визначення критичного стану людини базується на системі показників NEWS 2, але має певні відмінності. Сенсорна мережа виконує вимірювання не по всім життєвим ознакам, зазначеним в NEWS 2, а тільки по трьом: частоті серцевих скорочень, насиченості киснем та температурі тіла. Додатково система може визначати положення тіла та фіксувати набір не характерних рухів, що також враховується у визначені критичного стану людини. Саму шкалу балів значень показників (рис. 1), залишимо незмінною, а для показників з інерційного пристрою скористаємося шкалою з табл. 1.

Таблиця 1. Шкала оцінювання показників інерційного вимірювального пристрою

Вимірювана характеристика	**Бали**		**Вимірювана характеристика**	**Бали**			
	Не зафіксовано	**Зафіксовано**		**Не зафіксовано**	**1 хв**	**2 хв**	**3 хв**
Удар	0	1	Положення догори ногами	0	1	2	3
Вільне падіння	0	1	Відсутність руху	0	1	2	3

Позначимо характеристику, вимірювану сенсором як SD_i (Sensor Data) де $i = 1, \dots, N$, N – кількість вимірювальних показників. Функція $f(SD_i) = S_i$ визначає кількість відведених балів відповідному життєвому показнику у зіставлені зі шкалою визначеною системою NEWS 2, де S_i (Score) – отриманий бал i-го показника. Отже сумарно оцінити стан людини LR (Life Rate) можна за формулою 1.

$$LR = \sum_{i=1}^{N} f(SD_i) = \sum_{i=1}^{N} S_i \quad (1)$$

Оскільки інерційний вимірювальний пристрій дає 4 показники, спочатку потрібно обчислити SD_i за формулою 2, а потім використати в формулі 1. Це забезпечує рівноцінний внесок у розрахунок стану при чому такі значущі показники, як довге перебування догори ногами чи нерухомість, більше впливають на визначення критичного стану, ніж одиничні події, такі як падіння чи удар.

$$SD_{BodyPosition} = \max(SD^{HighG}_{BodyPosition}, SD^{FreeFall}_{BodyPosition}, SD^{UpsideDown}_{BodyPosition}, SD^{NoMotion}_{BodyPosition}), \quad (2)$$

де $SD_{BodyPosition}$ – показники положення тіла, а $SD^{HighG}_{BodyPosition}$, $SD^{FreeFall}_{BodyPosition}$, $SD^{UpsideDown}_{BodyPosition}$, $SD^{NoMotion}_{BodyPosition}$, характеристики удару, вільного падіння, положення догори ногами та відсутності руху, відповідно.

Додатково варто врахувати можливості розрядження батарей або інших проблем з підключенням окремих сенсорів мережі. В такому випадку дані від них не будуть отримані, що може привести до помилкової оцінки стану людини. Щоб врахувати такі ситуації пропонується
формула 3.

$$LR_N = \frac{LR}{\sum_{j=1}^{M} \max(f(SD_j))} = \frac{LR}{\sum_{j=1}^{M} \max(S_j)}, \quad (3)$$

де LR_N – нормалізоване значення оцінки стану людини, LR – стан людини, визначений за формулою 1, $\max(f(SD_j)) = \max(S_j)$ – максимальна кількість балів, що може бути отримана за шкалою відповідного j-го показника. Тут $j = 0, \ldots, M$, де M – кількість справно працюючих вузлів, що підключені до шлюзу сенсорної мережі. Це нормалізоване значення може вказувати на фізіологічний стан людини відповідно до Табл. 2.

Таблиця 2. Критичність стану людини за отриманою нормалізованою оцінкою

Діапазон нормалізованого значення оцінки стану людини	**Критичність фізіологічного стану людини**
[0.0 … 0.4)	Низький рівень загрози життю
[0.4 … 0.7)	Помірний рівень загрози життю
[0.7 … 1.0]	Дуже високий рівень загрози життю

У разі визначення критичного стану людини шлюз формує повідомлення, яке включає наступні дані: ступінь критичності фізіологічного стану людини, життєві показники (пульс, відсоток насичення киснем, температура), масив з 4-ох значень подій на інерційному вимірювальному пристрої та дані про місцезнаходження (широта та довгота).

Температуру тіла можна зберігати та передавати в тому самому форматі, в якому вона приходить з датчика температури, її розмір становитиме 2 байти. Пульс та відсоток насичення киснем представляються невеликими цілими числами, тому вони також можуть бути записані одним байтом. Дані про положення та діяльність тіла визначаються чотирма станами, кожен з яких може набувати чотирьох значень, що характеризують протяжність фіксування даного стану. Для кодування цієї інформації можна використати 4 поля по·2 біти, що становить один байт (рис. 6). Довгота і широта мають бути визначені з точністю до мікроградусів, тому розмір кожного показника становить 4 байти. Описана структура полів кінцевого повідомлення показана на рис. 7.

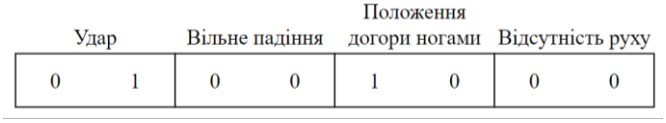


Рисунок 6. Кодування інформації з сенсора визначення руху та положення тіла

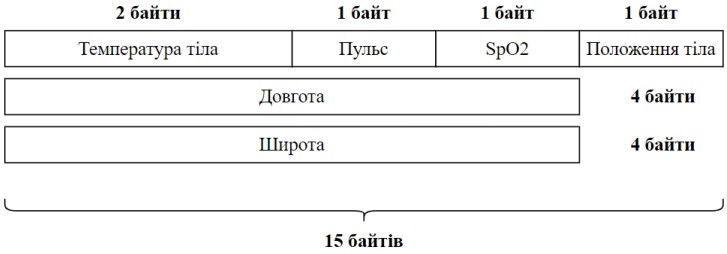


Рисунок 7. Структура полів повідомлення

3.5. Реалізація прототипу

Для перевірки працездатності основних рішень в побудові запропонованої натільної сенсорної мережі було реалізовано спрощений прототип, який складається з сенсора вимірювання температури та шлюзу. На прототипі було перевірено правильність побудови принципових схем, механізм реєстрації та відключення сенсора від шлюзу, організацію передачі даних з сенсора на шлюз по Bluetooth LE, алгоритм обробки даних на шлюзі та формування повідомлення.

До початку роботи з системою користувач має зареєструвати сенсори для їх ідентифікації аналізуючим модулем-шлюзом. Для входу в режим реєстрації необхідно затиснути кнопку протягом 1 – 5с. Ідентифікатором входу в цей режим буде світіння синього світлодіода. Встановивши сенсор та шлюз в режим реєстрації, необхідно піднести їх близько один до одного таким чином, щоб рівень потужності прийнятого сигналу на аналізуючому модулі був не менше встановленого порогового. За виконання цих умов буде встановлено з’єднання, що підтвердиться швидким мерехтінням синього світлодіода. Вихід з режиму реєстрації виконується при повторному натисканні кнопки протягом 1–5 с, після чого світлодіоди вимикаються.

Для роз’єднання сенсору зі шлюзом необхідно спочатку виконати перехід в режим видалення пристрою шляхом затисненням кнопки не менше 5с. Свідченням про вхід в цей режим є світіння синього світлодіода. Аналогічно до реєстрації, при близькому положенні пристроїв один до одного, відбувається видалення зі списку зареєстрованих. Ідентифікатором успішного виконання даної операції є повільне мерехтіння синього світлодіода. Повторне тривале натискання на кнопку (від 5 с) виконує вихід з режиму видалення.

При успішній реєстрації шлюзу на сенсорі температури, сенсор переходить в стан періодичного рекламування. Контролер сенсора періодично пробуджується та засинає за таймером. Коли контролер прокидається, він пробуджує датчик, датчик виконує замір, контролер зчитує його показники через інтерфейс I2C, після чого датчик одразу засинає. В MAX30205 [12] показник температури представляється в двох байтах (рис. 8). На контролері сенсора значення температури обраховуються шляхом складання степені двійки у відповідних бітах зі значенням 1. Отримавши значення температури контролер формує пакет даних і надсилає його на аналізуючий модуль-шлюз.

UPPER BYTE								LOWER BYTE							
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
S	MSB 64°C	32°C	16°C	8°C	4°C	2°C	1°C	0.5°C	0.25°C	0.125°C	0.0625°C	0.03125°C	0.015625°C	0.0078125°C	0.00390625°C
	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}

Рисунок 8. Представлення значення температури в MAX30205

При отримані даних температури, шлюз виконує аналіз за зазначеним алгоритмом визначення критичності стану людини, після чого формує структуру повідомлення.

3.6. Тестування енергоспоживання

Забезпечення енергоефективності та автономності є важливими вимогами для натільної мережі. Тому було проведено тестування енергоспоживання на аналізуючому модулі та на сенсорі вимірювання температури. Значення струму фіксувалися при перебуванні пристроїв в режимах глибокого сну та сканування/рекламування (табл. 3).

Таблиця 3. Результати замірів споживаного струму пристроями мережі

Пристрій	Режим	Номер експерименту					Середнє
		1	2	3	4	5	
Аналізуючий модуль	Глибокий сон, мкА	90,8	91,0	91,2	90,7	91,1	90,96
	Сканування, мА	85,3	84,4	86,3	86,0	86,2	85,64
Сенсор вимірювання температури	Глибокий сон, мкА	94,1	94,0	93,9	94,8	94,5	94,26
	Рекламування, мА	88,0	88,4	88,1	88,7	88,3	88,3

Середнє споживання струму сенсором вимірювання температури в Табл. 3 є трохи більшим ніж аналізуючим модулем. Це пов’язано з тим, що окрім самого контролера, струм іде ще і на плату датчика, котрий хоч і перебуває більшість часу в стані сну, але все одно має певне енергоспоживання.

Враховуючи, що аналізуючий модуль-шлюз оснащений акумулятором ємністю 3500 мА·год, а сенсор вимірювання температури – батарейкою CR3032 ємністю 500 мА·год, можемо розрахувати час роботи пристроїв.

Оскільки шлюз зазвичай перебуває в режимі періодичного сканування, то при виконанні сканування протягом 3 с та сну 5 хв, час його роботи до повного розряджання акумулятора буде складати 155 днів. Однак, варто враховувати, що при під’єднанні модуля супутникового зв’язку енергоспоживання зросте, і час роботи модуля-шлюзу зменшиться. Сенсор температури в основному перебуває в режимі періодичного рекламування. При часі активності у 3 с та сну 5 хв час роботи сенсора до розряду батареї буде складати 21 день.

Отже, без підзарядки акумулятора та заміни батарейок сенсорна мережа зможе працювати близько 20 днів. Такий результат можна вважати достатнім для визнання системи автономною.

4. ВИСНОВКИ

В даній роботі було запропоновано проєкт натільної сенсорної мережі для моніторингу життєво важливих показників людини у військовій галузі, експедиціях та екстремальному туризмі.

На основі аналізу публікацій, присвяченим аналогічним системам та оцінки стану організму людини для раннього попередження NEWS 2, запропоновано проект натільної сенсорної мережі.

Розроблене програмне забезпечення реалізує функціонал зчитування інформації з датчика температури, обміну даними між сенсором вимірювання температури та аналізуючим

модулем-шлюзом, налаштування натільної сенсорної мережі і аналіз критичності стану людини на основі модифікації системи показників NEWS 2.

Реалізація прототипу показала, що побудована на його основі система здатна задовольнити такі базові вимоги як: економічність, енергоефективність, портативність, невеликий розмір та легка вага, робота тільки із зареєстрованими пристроями, забезпечення працездатності при виході з ладу певних сенсорів. Аналізуючий модуль та сенсор успішно виконують реєстрацію та встановлюють з'єднання. Шлюз успішно приймає дані з сенсора, ідентифікує випадки виходу переданих значень поза норму та розраховує показник критичності стану людини. Також реалізовано зручний механізм налаштування пристроїв за допомогою простого натискання кнопки.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Zephyr Performance Systems, Medtronic. URL: https://www.zephyranywhere.com/system/overview (дата звернення: 01.11.2024).

2. The Hexoskin Smart Clothing, Hexoskin. URL: https://hexoskin.com/ (дата звернення: 01.11.2024).

3. Technical Report TR-08-05, Division of Engineering and Applied Sciences, Harvard University, 2005. URL: https://dash.harvard.edu/handle/1/24829604 (дата звернення: 01.11.2024).

4. Novel System for Persistent Access to Tactical Casualty Health (PATCH), Vivonics Inc. URL: https://www.vivonics.com/technologies//patch (дата звернення: 01.11.2024).

5. Petrenko A. I., Approaches for WSN (Wireless Sensor Networks) Standardization and their Interoperability in Combining into a Global Network. 2023.

6. LifeGuard Vital Signs Monitoring System URL: https://nlsp.nasa.gov/view/lsdapub/lsda_hardware/IDP-LSDA_HARDWARE-0000000000000495 (дата звернення: 01.11.2024).

7. Royal College of Physicians. National Early Warning Score (NEWS): Standardising the assessment of acuteillness severity in the NHS. Report of a working party. London: RCP, 2012. URL: https://archive-rcplondon.zedcloud.co.uk/file/32/download (дата звернення: 01.11.2024).

8. Royal College of Physicians. National Early Warning Score (NEWS) 2: Standardising the assessment of acute-illness severity in the NHS. Updated report of a working party. London: RCP, 2017. URL: https://archive-rcplondon.zedcloud.co.uk/file/8636/download (дата звернення: 01.11.2024).

9. RockBLOCK 9603 Developer Guide URL: https://www.mouser.com/datasheet/2/813/RockBLOCK-9603-Developers-Guide_1-1223968.pdf?srsltid=AfmBOorR5IBlUMN7CTeZRVOVpPCRy6KJqBiiTsSON4CRY4YswKiIIjj7 (дата звернення: 03.11.2024).

10. MatekSys GNSS & COMPASS, M10Q-5883 URL: https://www.mateksys.com/?portfolio=m10q-5883#tab-id-2 (дата звернення: 03.11.2024).

11. ESP32-C3 Series Datasheet Version 1.7 - Espressif Systems, 2024. URL: https://www.espressif.com/sites/default/files/documentation/esp32-c3-mini-1_datasheet_en.pdf (дата звернення: 03.11.2024).

12. Human Body Temperature Sensor MAX30205 Datasheet – Maxim Integrated Products, Inc., 2016. URL: https://www.analog.com/media/en/technical-documentation/data-sheets/max30205.pdf (дата звернення: 03.11.2024).

АНАЛІЗ ТА ОПТИМІЗАЦІЯ МЕТОДУ РЕНДЕРИНГУ RAY MARCHING У 3Д ГРАФІЦІ

Платонов М.О.[1], Безносик О.Ю.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] platonov.nikita@lll.kpi.ua, [2] beznosyk.oleksandr@lll.kpi.ua [0000-0003-2775-6070]

Метою даного дослідження є аналіз та оптимізація методу рендерингу комп'ютерної тривимірної графіки ray marching. Наукова новизна даної роботи полягає у вдосконаленні механізму прорисовки комп'ютерних зображень тривимірного простору за допомогою методу маршрування променів. Практична цінність дослідження полягає у оптимізації побудови математичної моделі світу, що робить цей метод більш доступним для широкого використання без потреби у надмірних потужностях для створення та дослідження математично вірних моделей віртуального світу.

Ключові слова: ray marching, рендеринг, комп'ютерна графіка, тривимірна графіка.

1. ВСТУП

Сучасні комп'ютерні графічні технології швидко еволюціонують, забезпечуючи надзвичайно реалістичне відтворення віртуальних світів на моніторі кожного комп'ютера. Основною задачею цих технологій є переведення віртуальної 3D сцени на двовимірний монітор за найкоротший час, оскільки частота оновлення становить в середньому 60 Герц. Існують різні типи рендерингу, які дозволяють досягти цієї мети. Наразі вони використовуються майже всюди, оскільки виведення зображення на екран є базовою функцією користувацького комп'ютера. Також через суттєвий зріст потужності пристроїв почали застосовуватись нові методи рендерингу, оскільки раніше, коли вони були винайдені, їх можна було розглядати лише на папері як принцип або ідею.

Одним із найефективніших підходів до рендерингу складних 3D сцен є методи ray tracing, які використовуються для моделювання поширення світла у віртуальному середовищі. Проте, класичне ray tracing має свої обмеження в умовах складної геометрії та динамічних сцен, що викликає необхідність пошуку нових методів.

Одним із перспективних напрямків у цій галузі є ray marching – метод, який використовується для рендерингу об'єктів з використанням імпліцитних поверхонь, визначених через функції відстаней (Signed Distance Fields, SDF). Цей підхід дозволяє працювати з комплексними поверхнями та сценами, де класичні методи відстеження променів можуть бути менш ефективними. Завдяки своїй гнучкості та можливості точного моделювання складних сцен, ray marching набуває все більшого поширення в таких областях, як процедурне моделювання, візуалізація медичних даних та розробка комп'ютерних ігор. І, незважаючи на доволі велике споживання обчислювальних ресурсів, цей метод все одно є досить поширеним у використанні.

2. ДОСЛІДЖЕННЯ ТА АНАЛІЗ МЕТОДУ RAY MARCHING

2.1. Signed Distance Field (SDF)

Основою методу рендерингу ray marching є функції визначення відстаней між точкою та об’єктом у просторі [1]. Ці функції відомі як Signed Distance Functions (SDF) і повертають мінімальну відстань від точки до найближчої поверхні об’єкта. Якщо значення від’ємне, точка розташована всередині об’єкта, якщо додатне — поза ним. Використання Signed Distance Field дозволяє легко визначати межі об'єктів у просторі, а також розраховувати відстань, на яку необхідно "просунутися" вздовж променю для уникнення перетину об'єкта. SDF має кілька переваг у рендерингу:

- Точність та ефективність: Використання SDF дозволяє оптимізувати трасування променів, що значно прискорює процес рендерингу. Оскільки метод обчислює мінімальну відстань, промені рухаються стрибками, уникаючи непотрібних розрахунків.
- Легке визначення складних форм: SDF дозволяє визначати як прості, так і складні форми через прості математичні формули, що спрощує роботу з геометрією.
- Композиція об'єктів: За допомогою SDF можна створювати нові об'єкти шляхом комбінації, об’єднання та віднімання геометричних форм. Це значно розширює можливості моделювання.

Типові приклади Signed Distance Functions для об’єктів включають сфери, куби, циліндри та інші прості форми.

Розглянемо математичну модель пошуку SDF. Отже, нехай у нас є точка в просторі P(x,y), а SDF(P) – відстань від точки P до найближчої поверхні об’єкта. Тоді загальна формула функції відстані буде виглядати наступним чином [2]:

$$SDF(P) = min_{Q \in S} \|P - Q\|,$$

де Q – будь-яка точка на поверхні об’єкта S, а $\|P - Q\|$ – це евклідова відстань між точками P і Q.

Саме через це SDF є дуже універсальним способом знаходження відстані. Використовуючи формули примітивних фігур, можна знайти де та як саме точка відноситься до об’єкту на сцені.

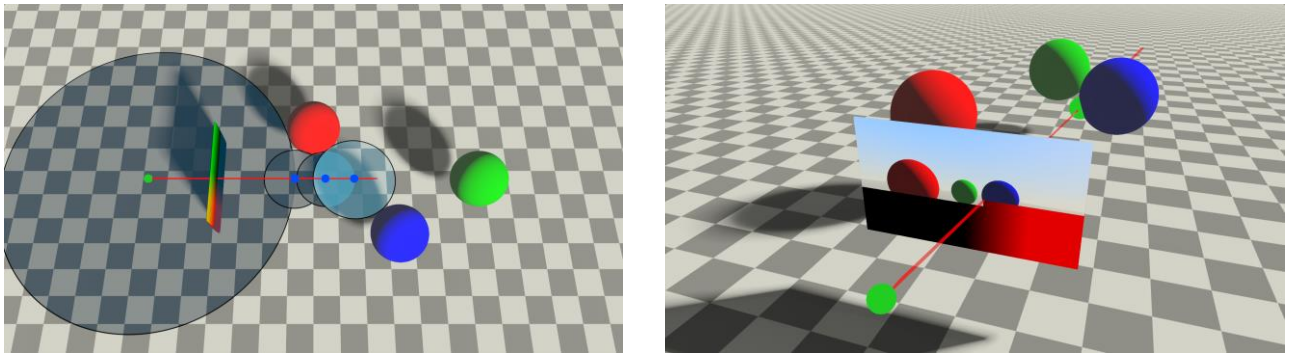
2.2. Аналіз роботи алгоритму ray marching

Метод ray marching прорисовує зображення за допомогою методу визначення відстаней (Signed Distacne Field – SDF). На відміну від методу трасування променів, який вираховує конкретну точку перетину променю та об’єкту, ray marching вираховує відстань від променів до об’єкту за допомогою SDF, який не перетинає сам об’єкт. Цей метод полягає в ітеративному русі вздовж променю, запущеного з камери, до тих пір, поки не буде досягнуто поверхні об’єкта або максимуму дозволених ітерацій. Процес ray marching для відображення об'єктів на екрані можна описати такими кроками (рис. 1, а) [3]:

- Ініціалізація променю: Кожен піксель на екрані відповідає окремому променю, який випускається з точки огляду камери. Кут запуску кожного променю обчислюється на основі положення пікселя та параметрів камери (позиція, орієнтація та поле зору).
- Крокування (marching) уздовж променю: На кожному кроці визначається відстань до найближчого об’єкта за допомогою функції Signed Distance Function. Ця відстань вказує на мінімальну довжину кроку, на яку можна "пройти" вперед уздовж променю, гарантуючи, що на цьому кроці променю не відбудеться перетину з поверхнею об'єкта.
- Перевірка на досягнення поверхні об’єкта: Якщо відстань до об’єкта, повернена SDF, стає меншою за певний поріг (наприклад, epsilon), це означає, що промінь досяг

поверхні об'єкта з бажаною точністю, і можна завершити marching. У цьому випадку піксель отримує колір, що відповідає властивостям поверхні об’єкта в цій точці (з урахуванням освітлення, кольору, текстури тощо).

- Завершення ітерацій або вихід за межі сцени: Якщо під час крокування променю не вдається знайти точку перетину з жодним об'єктом після певної кількості ітерацій, або якщо відстань від променю стає надто великою (вихід за межі сцени), процес ray marching завершується. У цьому випадку вважається, що промінь не перетнув об’єктів, і піксель може отримати колір фону або значення, що позначає відсутність об'єктів.
- Визначення кольору та освітлення: Після досягнення поверхні об’єкта розраховується освітлення для визначення кольору пікселя. Зазвичай для цього використовується модель освітлення (наприклад, моделі Фонга або Бліна-Фонга), яка враховує різні компоненти освітлення, такі як дифузне, відбите та навколишнє світло (рис. 1, б). Також можна додатково обчислити нормалі поверхні, застосувавши градієнти Signed Distance Function.



а) б)

Рисунок 1, а) приклад роботи одного кроку ray marching [4],
б) ітеративний процес рендерингу і застосування кольору [4]

3. ОПТИМІЗАЦІЇ МЕТОДУ RAY MARCHING

3.1. Аналіз недоліків традиційного ray marching

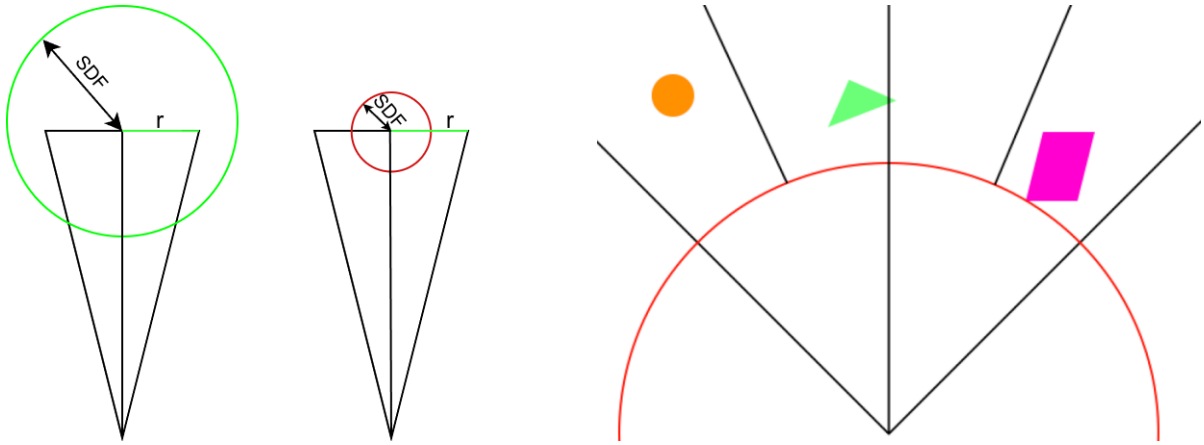
Основний недолік алгоритму ray marching — це велика кількість обчислень, необхідних для кожного пікселя. Оскільки кожен промінь "крокує" через сцену до тих пір, поки не знайде поверхню об’єкта або не вийде за межі сцени, метод вимагає багато ітерацій для кожного променю. Це може займати значну кількість ресурсів, особливо для складних сцен з безліччю об’єктів і тонких деталей. Кожен крок вимагає обчислення Signed Distance Function, а також часто додаткових розрахунків для ефектів освітлення, тіней та інших візуальних характеристик.

Окрім обчислювальної інтенсивності, алгоритм також страждає від труднощів із точністю. Якщо поріг точності (epsilon) обрано неправильно, це може призвести до помилок — промені можуть "пройти крізь" поверхні або не досягти їх зовсім. В результаті, для забезпечення високої якості зображення потрібно проводити ще більше ітерацій або використовувати адаптивні алгоритми, що ще більше збільшує обчислювальні витрати. Також промені обчислюються паралельно на графічному процесорі, таким чином, вони не обмінюються інформацією, через що треба робити нові обчислення для кожного променю окремо.

3.2. Розробка оптимізованого методу ray marching

Основний метод ray marching полягає у випусканні променю з кожного пікселя екрану, який потім ітеративно обраховує всі можливі зіткнення з об'єктами на сцені. Це дуже ресурсоємна операція, і при цьому багато сусідніх променів мають дуже невелику різницю між кінцевими обрахунками, оскільки проходять майже такий самий шлях і тому мають дуже схожі SDF. Таким чином, можна, замість того, щоб випускати промінь з кожного пікселю, випускати конус, який буде відображати зображення для декількох пікселів. Кількість конусів буде набагато менша ніж кількість променів для кожного пікселя. Логіка обчислення SDF для конуса буде відрізнятись, оскільки змінився сам принцип — змінюється правило, за яким конус продовжує марширування. Конус буде продовжувати марширування у разі, якщо відстань (SDF) від центра конусу до об'єкта буде більшою, ніж радіус конусу (рис. 2, а). Далі вже для обчислення інших пікселів розділимо конус на два, і зробімо замір SDF до найближчого об'єкту. Новий алгоритм буде включати такі дії:

1. На першому етапі вираховуємо SDF до найближчого об'єкту на сцені. Зазвичай він завжди більший ніж радіус конусу, тому продовжуємо ітерацію.
2. Далі вираховуємо SDF від центра конусу до найближчого об'єкта, дивимось, чи він менший за радіус конуса. Якщо менший, то закінчуємо цю ітерацію і ділимо конус на два (рис. 2, б). Знову проводимо SDF від вже нових конусів до найближчих об'єктів і перевіряємо, чи відстань між ними більше ніж радіус конусу.
3. Якщо під час розрахунку SDF тільки збільшується, то це означає, що далі більше немає об'єктів, які треба рендерити, тому можна цей конус відкинути і не робити більше обчислень.



а) б)

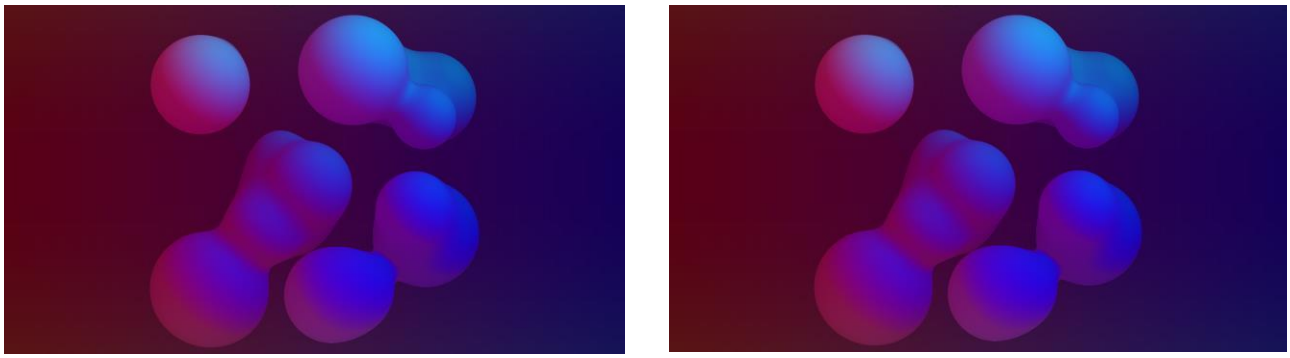
Рисунок 2, а) зліва: продовжуємо марширування конусів, справа: зупиняємось, б) приклад розділення конусу на два

Алгоритм виконується до моменту, коли буде досягнуто максимальну кількість ітерацій. Таким чином, можна обмежити дальність прорисовки об'єктів (інакше алгоритм буде працювати нескінченно).

3.3. Аналіз продуктивності розробленого методу

Порівняння двох методів буде відбуватися за кількістю кадрів за секунду та враховувати, наскільки відрізняються зображення. У програмі на мові програмування C++ за допомогою графічної бібліотеки OpenGL реалізуємо просту сцену зі сферами роздільною здатністю 1920

на 1080 пікселів за допомогою алгоритму ray marching (рис. 3, а) та розробленого методу конусів (рис. 3, б). У випадку з методом традиційного ray marching було отримано показник 48 кадрів за секунду, у той час як метод конусів зміг зробити 65 кадрів за секунду. При цьому якість зображення методу конусів майже не відрізняється від ray marching, оскільки ці методи використовують одну і ту ж систему SDF.



а) б)

Рисунок 3, а) рендеринг сфер за допомогою ray marching,
б) рендеринг сфер за допомогою методу конусів

4. ВИСНОВКИ

Ray marching — це потужний метод для рендерингу складних тривимірних сцен, що використовує Signed Distance Fields для точного визначення відстаней до об’єктів. Завдяки цьому методу можливо створювати детальні зображення з реалістичним освітленням та тінями, але він має значний недолік — велику кількість обчислень. Ray marching чудово підходить для генерації високоякісних зображень, але може бути надто повільним для інтерактивних додатків та потребує оптимізації для ефективного використання.

Оскільки ray marching має доволі серйозний недолік, який полягає у тому, що кожен піксель вимагає багато ітерацій для визначення найближчої поверхні, що робить алгоритм обчислювально затратним, особливо для складних сцен, було розроблено метод, завдяки якому ітерація йде не променями з кожного пікселя, а конусами. Це дозволяє не прораховувати промені, які не перетинаються з об’єктами, тим самим сильно зменшуючи затрати для рендеру кадру.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Inigo Quilez. Inigo Quilez :: computer graphics, mathematics, shaders, fractals, demoscene and more. URL: https://iquilezles.org/articles/raymarchingdf/ (дата звернення: 11.11.2024).

2. Hart J. C. Sphere tracing: a geometric method for the antialiased ray tracing of implicit surfaces. The Visual Computer. 1996. Т. 12, № 10. С. 527–545. URL: https://doi.org/10.1007/s003710050084 (дата звернення: 20.10.2024)

3. kishimisu. An introduction to Raymarching, 2023. YouTube. URL: https://www.youtube.com/watch?v=khblXafu7iA (дата звернення: 11.11.2024).

4. Shadertoy. Shadertoy BETA. URL: https://www.shadertoy.com/view/wlSGWy (дата звернення: 11.11.2024).

АДАПТАЦІЯ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ ДО НИЗЬКО-РЕСУРСНИХ МОВ ТА ДІАЛЕКТІВ

Поплавський В.О.[1], Кислий Р.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] poplavskyi.vladislav@lll.kpi.ua, [2] kvrware@gmail.com

Метою даного дослідження є адаптація великих мовних моделей (LLM) до діалектів та низько-реусрних мов, на прикладі гуцульського. Наукова новизна роботи полягає у розробці методології комбінування сучасних технологій та нестандартних підходів до їх використання, що дозволить LLM коректно працювати з текстами на діалектах, які не мають значних мовних ресурсів, таких як маленьки корпуси або словники. Це дослідження вперше використовує підхід генерації синтетичних даних, виокремлення правил з використанням reasoning моделей, використання Retrieval-Augmented Generation (RAG) для покращення перекладу та його подальшого використання у навчанні LLM. Практична цінність дослідження полягає в потенціалі адаптованих мовних моделей сприяти роботі з діалектами та розвитку української мови в цифровій сфері, зокрема для створення додатків і сервісів, що враховують регіональні особливості мови.

Ключові слова: великі мовні моделі, адаптація до діалектів, гуцульський діалект, синтетичні дані, збереження мовної різноманітності, reasoning model, корпус, NLP.

1. ВСТУП

У сучасному світі обробка природної мови набуває важливого значення в різних сферах – від технологій штучного інтелекту до цифрових комунікацій. Великі мовні моделі (LLM), такі як ChatGPT, BERT та інші, стали потужними інструментами в обробці текстів, що охоплюють різноманітні мови та задачі. Проте, більшість LLM оптимізовані для основних мов, що мають значні мовні ресурси, великі корпуси текстів та словники, тоді як діалекти та низько-ресурсні мови, зокрема українські діалекти, як наприклад гуцульський, залишаються мало вивченими. Це створює значний розрив у можливостях забезпечення якісної обробки тексту для цих мовних різновидів. Це обмеження ускладнює створення ефективних систем для розпізнавання та перекладу діалектів. Низько-ресурсні мови та діалекти є важливими для збереження культурної спадщини та створення адаптованих цифрових сервісів, які підтримують лінгвістичне розмаїття. Використання сучасних більших мовних моделей з reasoning, методів файн-тюну,

Таким чином, дослідження адаптації великих мовних моделей до низько-ресурсних мов, зокрема до українських діалектів, є актуальним напрямом розвитку обробки природної мови, який може мати як наукову, так і практичну цінність.

2. МЕТОДОЛОГІЯ

2.1. Підготовка та збір даних

Дані для адаптації великих мовних моделей до гуцульського діалекту були зібрані з різних джерел. Основним джерелом став роман «Дідо Іванчик», виданий двомовним форматом – текст гуцульською та його переклад українською мовою. Цей роман дозволив створити паралельний корпус з 9,852 пар речень. Додатково, з книг Люби-Параскевії Стринадюк «У нас, гуцулів» та її статей на платформі Zbruc було зібрано 281 та 1232 речення відповідно, які також були включені до корпусу для глибшого контексту та забезпечення більшої репрезентативності гуцульського діалекту. Вся зібрана інформація ретельно оброблялася, видалялися зайві символи, уточнювалася форма запису слів та їхнє розташування у корпусі, щоб забезпечити однорідність структури даних

2.2. Відновлення правил перекладу

Для автоматичної генерації правил перетворення між гуцульським діалектом та стандартною українською мовою було застосовано новітню модель від OpenAI chagpt o1, підхід у основі Chain-of-Thought (CoT) (Reasoning), який дозволяє моделі глибше аналізувати структуру мови, виділяючи послідовність думок, симулюючі реальний процес мислення з створенням, аналізом, критикою та відкиданням моделей та гіпотез, які призводять до остаточного рішення. Також модель має збільшений контекст на вході та для відповіді. За допомогою цього підходу модель визначає правила перекладу, які можуть бути використані для узгодження діалектного тексту зі стандартною українською мовою. Спочатку діалектний текст подається разом з еталонним перекладом, на основі якого модель формує правила замін. Далі ці правила тестуються на нових прикладах, щоб оцінити їхню точність та повноту. Можливе використання декількох ітерацій самостійного аналізу, щоб уточнити та адаптувати правила. Також можливо додавати ймовірнісний підхід з відбором правил, які найчастіше зустрічаються, та використовувати декілька різних чи “батчів” які перетинаються для удосконалення правил, перевірки, і покращення, моделлю самої себе для того, наскільки реальними є ці правила.

2.3. Побудова RAG-системи

Retrieval-Augmented Generation (RAG) використовується для додавання в контекст перекладу реальних вирівняних схожих речень, які можуть надати приклади використання правил та покращити переклад, симулюючі в цьому процесі реальний процес перекладу людиною, яка лише вчить мову. Система шукає найбільш релевантні українські речення, які відповідають смисловому контексту діалектного тексту. В подальшому цей набір даних можна так само доповнювати при появі нових реальних парних даних, для покращення цього процесу.

3. ГЕНЕРАЦІЯ СИНТЕТИЧНОГО ДАТАСЕТУ

Загальний підхід показано на рис. 1. Використання описаного підходу для генерації синтетичного датасету для діалектного перекладу є основним етапом, який дозволяє значно розширити обсяг наявних даних для тренування моделей, адаптованих до гуцульського діалекту. Даний процес включає декілька підходів, таких як пошук схожих структур за допомогою RAG, заміна слів із використанням технологій Named Entity Recognition (NER) та морфологічного аналізу при наявності словнику, а також використання інструкційних правил і системи агентів для уточнення порядку слів у реченнях. Далі наведено основні етапи цього процесу.

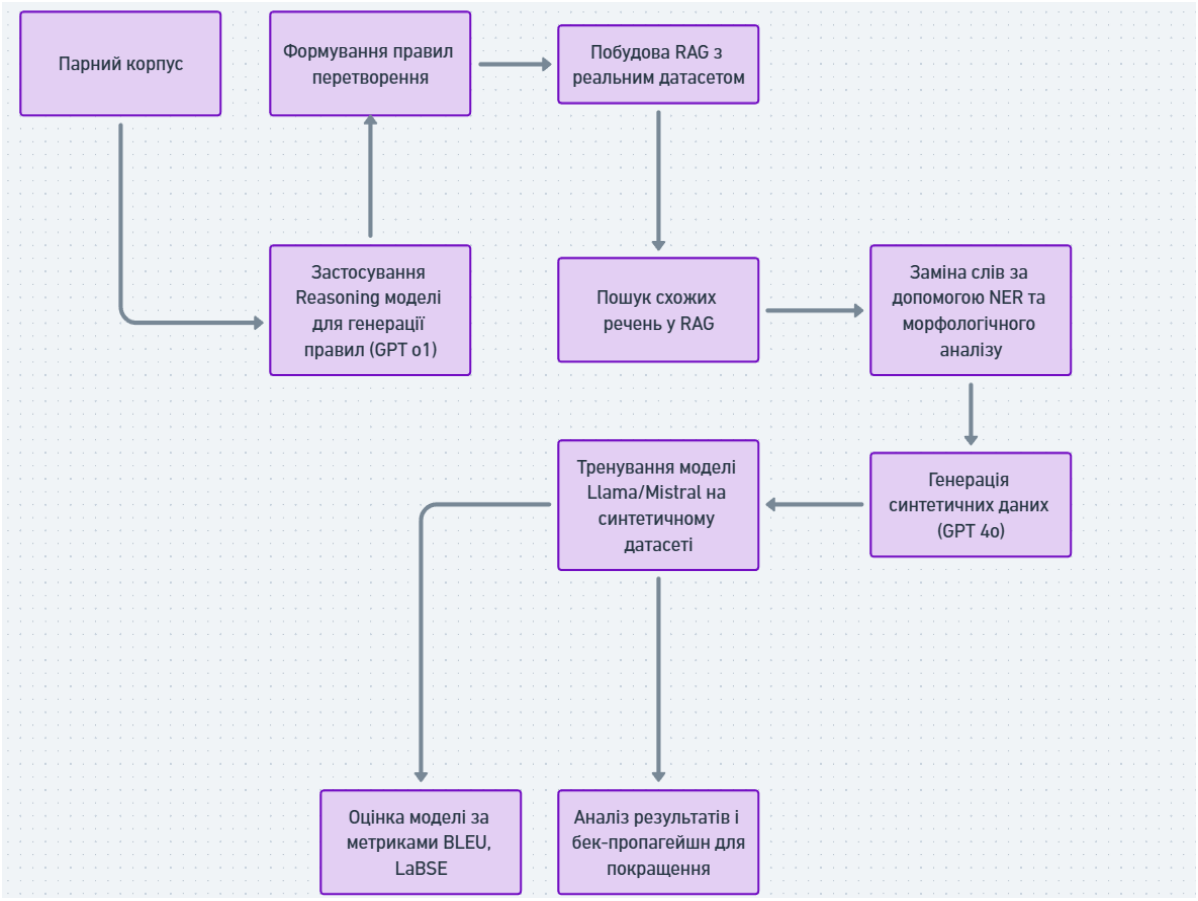


Рисунок 1. Загальний підхід

3.1. Заміна слів з використанням NER та морфологічного аналізу

Для забезпечення відповідності лексичних форм у синтетичному корпусі має сенс застосовувати підхід заміни слів із використанням Named Entity Recognition (NER) і морфологічного аналізу. Це дозволяє правильно визначати значення діалектних/українських термінів і їхню морфологічну форму, щоб уникнути неправильного узгодження у перекладених реченнях.

Спочатку виділяються ключові діалектні слова та імена власні за допомогою NER. Це дозволяє зберегти значення власних назв або унікальних діалектних термінів у перекладі, уникнувши неправильної інтерпретації многозначних слів.

Для точного узгодження відмінків, чисел та інших граматичних категорій здійснюється морфологічний аналіз. Це дозволяє визначити правильні форми слів та забезпечити їхнє коректне використання у точечних змінах в суфіксах/коренях/закінченнях нових реченнях, що створює більш природний вигляд синтетичного тексту. Використовуючи правила трансформації та словник, які були отримані на попередньому етапі, здійснюється автоматична заміна українських слів на діалектні.

3.2. Розширення корпусу за допомогою інструкційних правил та розгалуження обов’язків агентів

Щоб забезпечити вірний порядок слів у діалекті, можна до системи генерації синтетичних даних додати окремого агента, який буде працювати зі своїм контекстом зміни порядку слів за інструкціями які були так само витягнути більш великою моделлю, чи дотренувати окрему модель.

3.3. Використання синтетичного датасету для тренування моделей

Створений синтетичний корпус на основі RAG і системи агентів надалі може бути використаний для тренування меншої моделі, наприклад, моделей Mistral або Llama, які здатні зберігати важливі особливості діалектів та забезпечувати точний переклад на стандартну українську мову. Це дозволить масштабувати процес, адаптуючи модель до реальних умов роботи з діалектами, і забезпечити її продуктивність в умовах обмежених ресурсів.

3.4. Отримані правила

Наведемо приклад шаблону промпту, який був отриманий в результаті використання gpt o1.

req = '"Here are Grammatical Rules for Converting Ukrainian Text into the Hutsul Dialect Below is a comprehensive set of grammatical rules to help you convert Ukrainian text into the Hutsul dialect. They focus on phonetic changes, grammatical transformations, and syntactic patterns characteristic of the Hutsul dialect, with a particular emphasis on Word Order. Note: Apply specific rules such as changing "и" to "і" or softening consonants by adding "ь" only when you are certain they are appropriate in the context. 1. Vowel Changes 1.1. "є" instead of "а" or "я" (when stressed) "а" or "я" become "є" when stressed. Examples: "як" → "єк" "ягода" → "єгода" "яблуко" → "єблуко" "яйце" → "єйце" "ярмарок" → "єрмарок" "теля" → "телє" "шапка" → "шєпка" "капелюх" → "кєпелюх" "м'ята" → "мнєта" "колядник" → "колідник" "є" instead of "и" in some words: Examples: "йдеш" → "єдеш" "йшли" → "єшли" 1.2. "е" or sometimes "і" instead of "и" (when stressed) "и" becomes "е" or less commonly "і" when stressed. Examples: "абись" → "абес" "абисьти" → "абести" "жито" → "жето" "вим'я" → "вім'ї" 1.3. "и" instead of "і" "і" becomes "и". Examples: "Іван" → "Иван" "іду" → "иду" "із" → "из" "зійшла" → "зийшла" 1.4. "и" or "і" instead of "я" "я" becomes "и" or "і". Examples: "найся" → "найси" "як ся маєте" → "як сі маєте" "він знаходився" → "він находив сі" "я забився" → "я забивсі" 1.5. "у" instead of "ю" "ю" becomes "у". Examples: "сюди" → "суда" "вівцю" → "вівцу" "кішницю" → "кішницу" 1.6. "ві-" instead of "ви-" in prefixes The prefix "ви-" becomes "ві-". Examples: "витратити" → "вітратити" "виповісти" → "віповісти" 2. Consonant Changes 2.1. "ґ" instead of "д" (not always, only if confident) Examples: "дівка" → "ґівка" "дірка" → "ґєрка" "дідо" → "ґедьо" "дякую" → "ґєкую" "неділя" → "неґіля" "понеділок" → "понєґівок" "сидів" → "сиґів" "поділа" → "поґіла" 2.2. "ть" instead of "к" "к" becomes "ть" in certain words. Example: "донька" → "доньтя" 2.3. "д" instead of "ґ" "ґ" becomes "д". Example: "леґінь" → "ледінь" 2.4. "к" instead of "т" before "є", "і", sometimes at word endings "т" becomes "к" before "є", "і", or at the end of words. Examples: "християни" → "крескєни" "тяжко" → "кєшко" "повитягати" → "повікєгати" "святі" → "свєкі" "тіло" → "кіло" "по світу" → "по свікі" "смерть" → "смеркь" 2.5. Softened Sibilants (Sh, Ch sounds) Softening of sibilant consonants by adding a soft sign "ь". Examples: "чого" → "чьо" "нічого" → "нічьо" "челядь" → "чєлєдь" "шапка" → "шєпка" "чари" → "чєри" 2.6. "ц" instead of "ч" and vice versa "ч" becomes "ц", and "ц" becomes "ч". Examples: "чи" → "ци" "цибух" → "чибук" 2.7. Hard "с", "ц", "з" instead of softened ones Softened consonants "сь", "зь", "ць" become hard "с", "з", "ц". Examples: "щось" → "шос" "десь" → "дес" "на вулицю" → "на вулицу" "крізь" → "кріз" 2.8. "нн", "н" instead of "дн", "тн", "лн" "дн", "тн", "лн" become "нн" or "н". Examples: "передній" → "перенний" "мельник" → "менник" or "меник" 3. Use of Particles and Conjunctions 3.1. "і" to "тай" or "та й" The conjunction "і" is replaced with "тай" or "та й". Example: "Він і вона" → "Він тай вна" 3.2. "та" becomes "тай" "та" changes to "тай". Example: "Більше та краще" → "Бирше тай красше" 3.3. "й" to "тай" "й" becomes "тай". Example: "Він й вона" → "Він тай вна" 4. Negation 4.1. "не" to "ни" "не" becomes "ни". Example: "Не знаю" → "Ни знаю" 4.2. Double Negation Double negation is used for emphasis. Example: "Ніколи не бачив" → "Николи ни видів" 5. Pronouns and Possessives 5.1. "його" to "йиго" "його" becomes "йиго". Example: "Його книга" → "Йиго книга" 5.2. "мене" to "мині" "мене" becomes "мині". Example: "Він бачив мене" → "Він видів мині" 6. Reflexive Verbs 6.1. "ся" to "си" "ся" becomes "си". Example: "Сміється" → "Смієтси" 6.2. Placement of "си" "Си" is placed after the verb. Example: "Він думається" → "Він думаєтси" 7. Verb Endings and Conjugations 7.1. Third Person Singular Present Tense Verbs ending in "-є" or "-е" change to "-єт". Examples: "Має" → "Маєт" "Знає" → "Знаєт" "Ходить" → "Ходит" 7.2. Past Tense Verbs Past tense verbs often remain the same, but pronunciation may differ. Example: "Він сказав" → "Він сказав" (with dialectal pronunciation) 8. Prepositions 8.1. "від" to "вид" "від" becomes "вид". Example: "Від тебе" → "Вид тебе" 8.2. "у" to "в" "у" becomes "в" before consonants. Example: "У хаті" → "В хаті" 8.3. "до" becomes "д'" before vowels "до" becomes "д'" when followed by a word starting with a vowel. Example: "До Андрія" → "Д'Андрія" 9. Word Order and Syntax Word order in the Hutsul dialect can differ significantly from standard Ukrainian. The dialect often rearranges sentence components to emphasize certain elements, convey nuances, or align with traditional speech patterns. 9.1. Time and Place Expressions at the Beginning Rule: Time and place expressions are frequently placed at the beginning of the sentence to set the scene or emphasize when or where an action occurs.

Таким чином, розроблена методологія генерації синтетичного корпусу з використанням технологій RAG, NER, морфологічного аналізу та системи агентів для рефайнменту порядку слів забезпечує створення високоякісних синтетичних даних для адаптації великих мовних моделей до діалектних текстів використовуючи можливість моделей до генералізації правил та використання менш евристичних методів для зменшення простору для ілюзіонювання моделей.

4. ЕКСПЕРИМЕНТИ ТА РЕЗУЛЬТАТИ

Оцінка якості адаптованих моделей проводилася на основі низки метрик, що дозволяють глибше зрозуміти ступінь відповідності синтетичних даних реальним діалектним текстам:

Таблиця 2. Статистика по датасету

Метрика	Статистика
U-src	0.038
U-tgt	0.023
X	0.092

1. **BLEU (Bilingual Evaluation Understudy)**: Метрика, яка вимірює точність перекладу за допомогою порівняння n-грам перекладу з еталонними текстами. BLEU використовується для оцінки схожості з наявними текстами і допомагає визначити, наскільки близькі переклади до оригінального змісту діалекту.
2. **LaBSE (Language-agnostic BERT Sentence Embeddings)**: Модель, яка дозволяє оцінювати семантичну подібність між текстами, незалежно від мовних варіантів. LaBSE забезпечує кращу точність у визначенні смислових збігів, що є особливо важливим для діалектних текстів, де переклади можуть мати неточності на рівні лексики.
3. **Статистичні показники (згідно з Wieling et al., 2009)**:
    - **U-src (proportion of unaligned source characters)**: Пропорція невирівняних символів у діалектному тексті.
    - **U-tgt (proportion of unaligned target characters)**: Пропорція невирівняних символів у перекладеному тексті.
    - **V-C (vowel-to-consonant alignment)**: Пропорція вирівнянь голосних до приголосних (враховуючи виключення напівголосних, носових звуків тощо).
    - **X (crossing alignment pairs)**: Пропорція пар вирівнювань, що перетинаються.

Ці метрики дозволяють кількісно оцінити вирівнювання символів між діалектним та перекладним текстами і допомагають виявити проблемні місця для подальшої оптимізації.

4.1. Методологія покращення синтетичних даних

На основі отриманих метрик можна додатково впровадити ітеративний процес оптимізації, який дозволить адаптувати модель до реальних вимог діалектного тексту. Основні кроки методології покращення включають:

1. **Аналіз статистичних показників**: Після початкового запуску моделі аналізуються пропорції U-src, U-tgt, V-C та X для виявлення невідповідностей у вирівнюванні символів та корекції смислових зсувів.
2. **Ітеративний процес покращення**: Використовуючи результати початкової оцінки, проводиться бек-пропагейшн для корекції проблемних ділянок тексту. Це дозволяє моделі уточнювати правила вирівнювання та генерувати більш точні синтетичні приклади.

Цей комплексний підхід до експериментування та покращення моделі дозволяє побудувати високоякісний синтетичний корпус і забезпечити адаптацію великих мовних моделей до діалектних текстів з максимальною ефективністю.

5. ВИСНОВКИ

Дослідження продемонструвало ефективність розробленої методології адаптації великих мовних моделей до гуцульського діалекту та низькоресурсних мов. Застосування підходів на основі RAG, більших “розумних” мовних моделей для генерації синтетичних даних, NER та морфологічного аналізу дозволило створити синтетичний датасет, який істотно покращив якість перекладу діалектного тексту. Згенеровані правила трансформації, інструкційні підходи та система агентів для уточнення порядку слів забезпечили природну структуру синтетичних текстів, а також глибше розуміння контексту моделі.

Оцінка продуктивності показала високий рівень точності моделей, що підтверджують метрики BLEU, LaBSE, а також спеціальні статистичні показники, які демонструють покращення узгодженості та точності перекладу. Ці результати доводять доцільність використання запропонованої методології для роботи з діалектами та низькоресурсними мовами, що відкриває можливості для подальших досліджень у цій галузі.

Подальша робота може бути спрямована на розширення методології на інші діалекти, а також на інтеграцію додаткових технологій для покращення якості синтетичних даних та збору даних самих по собі. Це сприятиме збереженню культурного та мовного розмаїття, а також розвитку адаптованих мовних технологій у цифровому середовищі.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Held W., Ziems C., Yang D. TADA: Task-Agnostic Dialect Adapters for English. [Електронний ресурс] / Held W., Ziems C., Yang D. – arXiv preprint arXiv:2305.16651, 2023. – Режим доступу: https://arxiv.org/abs/2305.16651 – Дата доступу: 15.09.2024.

2. Liu Y., Held W., Yang D. DADA: Dialect Adaptation via Dynamic Aggregation of Linguistic Rules. [Електронний ресурс] / Liu Y., Held W., Yang D. – arXiv preprint arXiv:2305.13406, 2023. – Режим доступу: https://arxiv.org/abs/2305.13406 – Дата доступу: 20.09.2024.

3. Xiao Z., Held W., Liu Y., Yang D. Task-Agnostic Low-Rank Adapters for Unseen English Dialects. [Електронний ресурс] / Xiao Z., Held W., Liu Y., Yang D. – arXiv preprint arXiv:2311.00915, 2023. – Режим доступу: https://arxiv.org/abs/2311.00915 – Дата доступу: 25.09.2024.

4. Scherrer Y., Ljubešić N., Samardžić T. Automatic Normalisation of the Swiss German ArchiMob Corpus Using Character-Level Machine Translation. [Електронний ресурс] / Scherrer Y., Ljubešić N., Samardžić T. – arXiv preprint arXiv:1904.06565, 2019. – Режим доступу: https://arxiv.org/abs/1904.06565 – Дата доступу: 30.09.2024.

5. Ziems C., Held W., Yang D. VALUE: Evaluating Dialectal Bias of Pre-trained Language Models. [Електронний ресурс] / Ziems C., Held W., Yang D. – arXiv preprint arXiv:2305.16652, 2023. – Режим доступу: https://arxiv.org/abs/2305.16652 – Дата доступу: 05.10.2024.

ЗАСОБИ РОЗРОБКИ ЦИФРОВИХ ДВІЙНИКІВ ІЗ ЗАЛУЧЕННЯМ ТЕХНОЛОГІЙ IOT

Ратушнюк В.Є.[1], Булах Б.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] ratushniuk.vasyl@lll.kpi.ua, [2] bogdan.bulakh@gmail.com [0000-0001-5880-6101]

Метою даного дослідження є вивчення інструментів, які відіграють ключову роль у розробці цифрових двійників з інтеграцією IoT. Наукова новизна роботи полягає у розробці нових підходів до інтеграції цифрових двійників із технологіями IoT для забезпечення динамічного оновлення моделей у реальному часі, автоматизації управління фізичними об'єктами та підвищення точності прогнозів. Практична цінність дослідження полягає у можливості впровадження розроблених підходів до створення цифрових двійників на основі IoT для підвищення ефективності управління об'єктами та системами в різних галузях.

Ключові слова: цифровий двійник, IoT, платформи розробки цифрових двійників.

1. ВСТУП

Останніми роками концепція цифрових двійників або цифрових двійників привернула значну увагу в галузях завдяки її потенціалу революції в керуванні, моніторингу та оптимізації фізичних систем. Цифровий двійник – це віртуальна модель, яка точно повторює фізичний об’єкт, систему або процес у режимі реального часу, забезпечуючи покращене прийняття рішень, прогнозоване обслуговування та ефективність роботи. Ця технологія тісно пов’язана з розвитком і широким використанням Інтернету речей (IoT), оскільки пристрої IoT служать найважливішим мостом між фізичним світом і його цифровим аналогом.

Пристрої IoT, включаючи датчики та виконавчі механізми, постійно збирають дані в реальному часі з фізичних активів і передають їх на цифрові платформи-двійники, дозволяючи віртуальним моделям віддзеркалювати зміни, аналізувати продуктивність і прогнозувати майбутні результати. Ця конвергенція IoT і цифрових двійників стимулює інновації в різних секторах, включаючи виробництво, міське планування, охорону здоров’я та розумну інфраструктуру. У міру впровадження цифрових двійників на базі IoT у галузях потреба в розумінні інструментів, платформ і фреймворків, які полегшують їх створення та керування, стає надзвичайно важливою [1].

2. КОНЦЕПЦІЯ ЦИФРОВИХ ДВІЙНИКІВ З ВИКОРИСТАННЯМ IOT: ВІД ІНДУСТРІАЛЬНОЇ РЕВОЛЮЦІЇ ДО СУЧАСНОСТІ

Розвиток технологій завжди супроводжувався революційними змінами в промисловості та суспільстві. Четверта індустріальна революція, або Індустрія 4.0, знаменує нову еру, в якій цифрові технології, автоматизація та обмін даними стають основою для трансформації всіх сфер життя. У центрі цієї революції знаходяться такі інноваційні концепції, як цифрові двійники та Інтернет речей (IoT), які разом створюють нові можливості для управління фізичними об'єктами та процесами [2].

2.1 Цифрові двійники: основні принципи

Цифровий двійник – це віртуальна репрезентація фізичного об'єкта, системи або процесу, яка дозволяє відтворювати їхній стан, поведінку та функціонування в реальному часі. Ця технологія забезпечує постійний моніторинг, аналіз та прогнозування завдяки використанню великих обсягів даних, які надходять з фізичних об'єктів.

Реалізація цифрових двійників стала можливою завдяки стрімкому розвитку IoT, що забезпечує зв’язок між фізичними та цифровими світами. IoT-пристрої, такі як датчики, виконавчі механізми та розумні пристрої, збирають дані про стан об'єкта, передаючи їх на цифрові платформи. Ці дані дають змогу цифровому двійнику постійно оновлювати модель, аналізувати продуктивність та прогнозувати потенційні проблеми.

Окрім цього Інтернет речей є фундаментом для роботи цифрових двійників, оскільки забезпечує потік даних між фізичними об'єктами та їх віртуальними моделями. Цей зв'язок дозволяє двійникам не лише відображати поточний стан систем, але й активно впливати на них через механізми зворотного зв'язку. Наприклад, якщо датчик виявляє зниження продуктивності обладнання, цифровий двійник може запропонувати оптимізаційні рішення або навіть автоматично їх реалізувати.

2.2 Вплив на галузі

- Інтеграція цифрових двійників та IoT вже має значний вплив на різні галузі:
- Виробництво: Можливість прогнозувати збої, оптимізувати виробничі процеси та знижувати витрати на обслуговування.
- Міське планування: Створення розумних міст із динамічним управлінням інфраструктурою, енергетикою та транспортом.
- Охорона здоров’я: Відстеження стану пацієнтів та прогнозування розвитку захворювань.
- Енергетика: Оптимізація роботи енергомереж та інтеграція відновлюваних джерел енергії.

2.3 Актуальність теми

На сьогоднішній день концепція цифрових двійників з використанням IoT є надзвичайно актуальною. Вона відкриває нові горизонти для підвищення ефективності, зменшення витрат та покращення якості управління складними системами. У зв’язку зі стрімким зростанням обсягів даних, розвитку штучного інтелекту та підвищенням вимог до автоматизації, впровадження цифрових двійників стає не просто конкурентною перевагою, а необхідністю для багатьох підприємств та організацій.

Таким чином, дослідження та впровадження цифрових двійників у поєднанні з IoT є важливим кроком до створення більш інтелектуальних, адаптивних і стійких систем, що визначають майбутнє промисловості та суспільства [2].

3. ПЛАТФОРМИ РОЗРОБКИ ЦИФРОВИХ ДВІЙНИКІВ

3.1 Unity платформа

Unity – це потужна та універсальна 3D-платформа, яка широко використовується для розробки ігор, симуляцій та інтерактивного досвіду. Він став промисловим стандартом для створення в реальному часі, підтримуючи розробку гучних ігор, таких як Hearthstone, Pokémon GO і Escape from Tarkov. Його популярність пояснюється його надійними функціями, зручним інтерфейсом і гнучкістю, що робить його необхідним інструментом не лише для розробки ігор, але й для інших сфер, таких як віртуальна реальність, доповнена реальність і кінематографічне виробництво.

За своєю суттю Unity визначається як платформа для створення в реальному часі, що означає, що вона забезпечує миттєвий зворотний зв'язок і взаємодію у відповідь на вхідні дані користувача. Ця можливість кардинально змінює правила у сферах, які вимагають ітераційного проектування та швидкого створення прототипів. В ігровій індустрії зворотний зв'язок у реальному часі покращує досвід гравців, забезпечуючи безперебійну взаємодію з віртуальним середовищем. Окрім ігор, ця функція реального часу набула популярності в кіноіндустрії, де вона використовується для створення комп'ютерної анімації. У цьому контексті Unity дозволяє режисерам і аніматорам регулювати такі елементи, як ракурси камери, освітлення та рухи персонажів, на льоту, значно скорочуючи час і вартість виробництва. Здатність бачити зміни миттєво означає, що весь творчий процес стає більш гнучким і спільним.

В контексті цифрових двійників Unity пропонує значні переваги для їх розробки, поєднуючи взаємодію в реальному часі, розширену візуалізацію та повну інтеграцію з IoT та іншими джерелами даних. Його можливості в режимі реального часу гарантують, що цифрові двійники залишаються синхронізованими зі своїми фізичними аналогами, дозволяючи миттєво оновлювати та швидко реагувати на моделювання. Сумісність платформи з різними пристроями забезпечує доступ із персонального комп'ютеру і мобільних пристроїв, сприяючи віддаленому моніторингу та співпраці.

Завдяки вдосконаленій 3D-візуалізації Unity цифрові двійники можуть бути візуалізовані в дуже деталізованій і реалістичній манері, покращуючи розуміння та залучення користувачів. Крім того, Unity підтримує динамічне моделювання, що дозволяє користувачам тестувати різні сценарії та оптимізувати продуктивність системи, не порушуючи реальні операції. Інтеграція з пристроями IoT забезпечує безперервний потік даних, зберігаючи цифровий двійник точним і актуальним.

Крім того, багата ресурсами екосистема Unity, включаючи сховище активів і активну спільноту, прискорює розвиток, надаючи готові компоненти та підтримку. Ця комбінація функцій допомагає зменшити витрати, оптимізувати робочі процеси та покращити процес прийняття рішень, що робить Unity потужним інструментом для створення захоплюючих та інтерактивних цифрових подвійних рішень [3]. Приклад моделі та інтерфейсу зображено на рисунку 1:

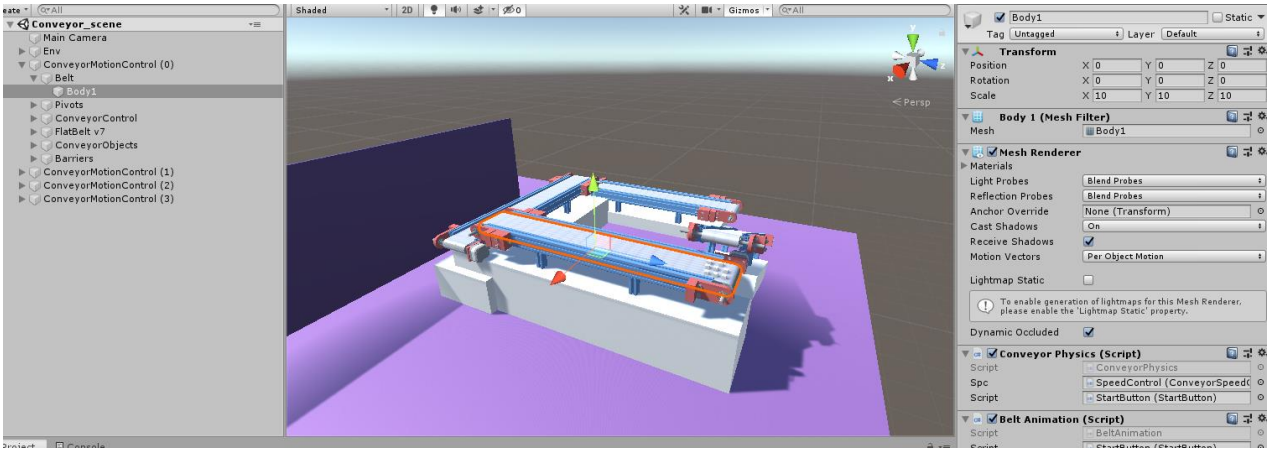


Рисунок 1. Приклад інтерфейсу та моделі конвеєру в платформі Unity

3.2 IBM Rhapsody

IBM Engineering Systems Design Rhapsody — це комплексне середовище моделювання, яке підтримує системну інженерію на основі моделей за допомогою діаграм SysML(Systems

Modeling Language) і UML(Unified Modeling Language). Це дозволяє проектувати, моделювати та перевіряти складні системи, сприяючи співпраці між командами інженерів та інтеграції з різними інструментами розробки. Інтеграція IBM Rhapsody з Unity дозволяє візуалізувати моделі SysML у 3D-середовищі, покращуючи розуміння складних системних архітектур. Цю інтеграцію можна досягти шляхом перетворення моделей SysML, розроблених у Rhapsody, у 3D-представлення в Unity. Такий підхід забезпечує віртуальне тестування та перевірку, забезпечуючи більш інтуїтивне розуміння поведінки системи цифрових двійників [4]. Приклад інтерфейсу даного середовища моделювання показано на рисунку 2:

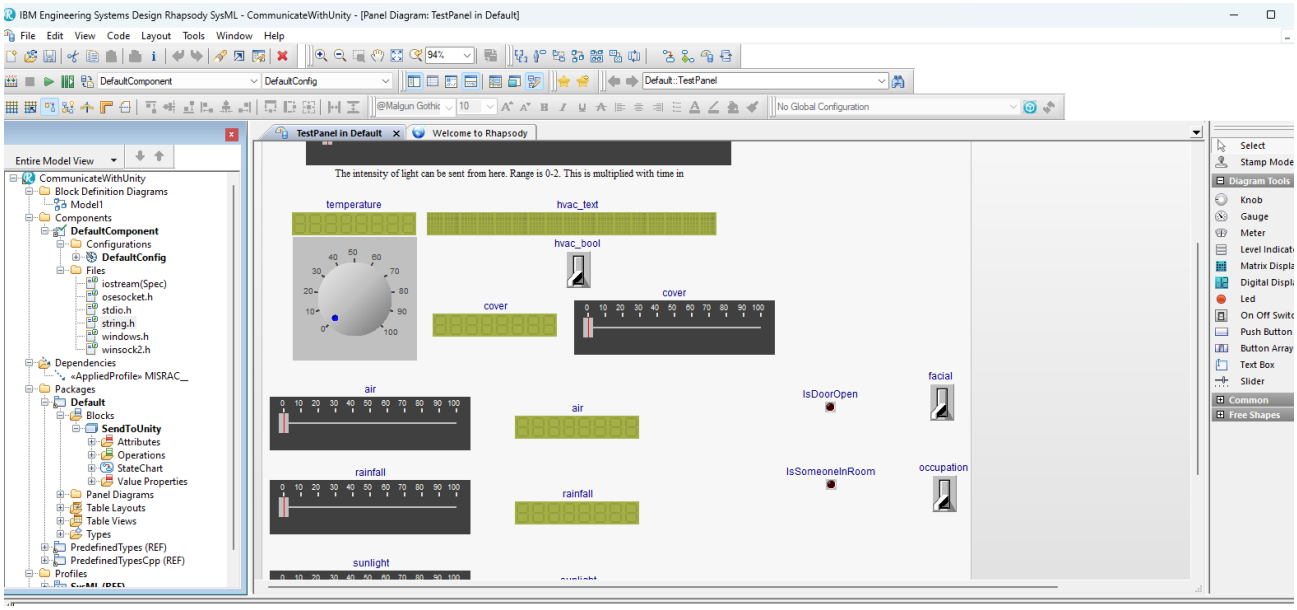


Рисунок 2. Приклад інтерфейсу IBM Rhapsody

4. ВИСНОВКИ

Розробка цифрових двійників, які використовують інтеграцію IoT і передових інструментів, таких як Unity та IBM Rhapsody, знаменує собою значний прогрес у сфері системного моделювання та симуляції. Цей підхід, заснований на принципах Четвертої промислової революції, змінює спосіб нашої взаємодії зі складними системами, пропонуючи розширені можливості моніторингу, оптимізації та прийняття рішень.

Unity служить потужною 3D-платформою в реальному часі, яка полегшує створення інтерактивних і візуально привабливих цифрових двійників. Його можливості візуалізації в реальному часі та кросплатформна підтримка дозволяють проводити динамічне моделювання та захоплюючий досвід, що робить його ідеальним для галузей промисловості, починаючи від виробництва та міського планування до охорони здоров'я та енергетики. Інтеграція Unity з IoT гарантує, що цифрові двійники постійно оновлюються реальними даними, забезпечуючи моніторинг у реальному часі та прогнозне обслуговування.

IBM Rhapsody, з іншого боку, забезпечує надійне середовище для проектування систем на основі моделей. Поєднання цих інструментів забезпечує безперебійний робочий процес для розробки цифрових двійників. Починаючи з точного системного моделювання в Rhapsody, перетворюючи ці моделі в Unity для 3D-візуалізації та інтегруючи дані IoT, організації можуть створити комплексні цифрові двійники. Ці двійники не тільки віддзеркалюють фізичні системи в режимі реального часу, але й дозволяють проводити широке тестування, симуляцію сценаріїв і оптимізацію, не порушуючи реальні операції.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. McKinsey & Company What is digital-twin technology? URL: https://www.mckinsey.com/featured-insights/mckinsey-explainers/what-is-digital-twin-technology (дата звернення: 09.11.2024).

2. Morozova O. Shchehlov V. Методи та технології розроблення цифрових двійників для гарантоздатних систем індустріального інтернету речей. URL: https://doi.org/10.26906/SUNZ.2022.4.127 (дата звернення: 10.11.2024).

3. Unity official documentation. URL: https://docs.unity3d.com/Manual/index.html (дата звернення: 10.11.2024).

4. IBM Rhapsody documentation. URL: https://www.ibm.com/products/systems-design-rhapsody. (дата звернення: 01.11.2024).

ПІДХОДИ ДО СТВОРЕННЯ ГОЛОСОВОГО ПОМІЧНИКА З ВИКОРИСТАННЯМ ІНСТРУМЕНТІВ OPENAI

Романенко С.В.[1], Булах Б.В.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] slav4ik77777@gmail.com, [2] bogdan.bulakh@gmail.com [0000-0001-5880-6101]

Метою даної роботи є розробка та впровадження голосового помічника з використанням сучасних інструментів OpenAI. У дослідженні розглядаються ключові аспекти обробки природної мови (NLP) і синтезу мовлення, а також аналізуються існуючі моделі генеративного штучного інтелекту. Основна увага приділяється архітектурі системи, яка інтегрує можливості OpenAI, та підходам до взаємодії з користувачем через голосові команди.

Ключові слова: OpenAI, обробка природної мови, синтез мовлення, голосовий помічник, штучний інтелект.

1. ВСТУП

У сучасному світі розвитку штучного інтелекту та технологій обробки природної мови (NLP) значного поширення набули голосові помічники, які забезпечують зручний і природний спосіб взаємодії користувачів з цифровими системами. З появою нових генеративних моделей, таких як GPT від OpenAI, стало можливим досягти нових висот у якості розпізнавання та синтезу мовлення, що дозволяє значно розширити можливості голосових асистентів.

Голосові помічники, такі як Siri, Alexa, Google Assistant, вже стали невід'ємною частиною повсякденного життя мільйонів користувачів. Проте з розвитком штучного інтелекту виникають нові виклики та можливості, особливо у контексті створення більш адаптивних та інтелектуальних систем, здатних обробляти складні запити та адаптуватися до різних мовних контекстів. Основною проблемою є необхідність розробки голосових помічників, які можуть не лише виконувати прості команди, але й взаємодіяти з користувачем на глибшому рівні.

Ця робота присвячена дослідженню підходів до створення голосового помічника з використанням інструментів OpenAI. Метою роботи є вивчення можливостей сучасних моделей генеративного штучного інтелекту, зокрема GPT, для інтеграції їх у голосового асистента, що здатний розпізнавати та генерувати мовлення з високою точністю і природністю. Поглиблене дослідження інструментів OpenAI дозволить оцінити їхню ефективність та відповідність вимогам до голосових помічників у різних сценаріях використання.

2. АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ ДО РОЗРОБКИ ГОЛОСОВИХ ПОМІЧНИКІВ

Обробка природної мови (NLP) є одним із ключових напрямів штучного інтелекту, що забезпечує ефективну взаємодію між комп'ютерами та людською мовою. Основна мета NLP полягає у тому, щоб комп'ютери могли розуміти, аналізувати та генерувати текст або мовлення, максимально наближене до природної людської комунікації.

Основні компоненти NLP

Процес обробки природної мови включає кілька важливих етапів [1, 2]:

- **Токенізація**: розбиття тексту на окремі слова або фрази для подальшого аналізу.
- **Визначення частин мови**: ідентифікація граматичних категорій слів, що допомагає зрозуміти їхню синтаксичну роль.
- **Синтаксичний аналіз**: побудова структури речення для виявлення взаємозв'язків між словами.
- **Лематизація та стемінг**: приведення слів до базових або кореневих форм для спрощення обробки.
- **Векторизація тексту**: перетворення текстових даних у числові формати для аналізу комп'ютером.
- **Розпізнавання сутностей**: виявлення значущих об'єктів у тексті, таких як імена, дати та місця.
- **Семантичний аналіз**: розуміння значення слів і речень у контексті.
- **Сентимент-аналіз**: визначення емоційного тону тексту, що важливо для аналізу відгуків та соціальних мереж.

Етапи процесу NLP

Процес обробки природної мови складається з послідовних етапів [1, 3]:

1. **Токенізація та нормалізація тексту**: перетворення тексту до нижнього регістру, видалення зайвих символів.
2. **Частиномовний аналіз**: визначення граматичних характеристик кожного слова.
3. **Синтаксичний і семантичний аналізи**: глибоке розуміння структури та змісту тексту.
4. **Векторизація тексту**: конвертація тексту у числовий формат для подальшого машинного аналізу.

Сучасні нейронні моделі, такі як Word2Vec або GPT, дозволяють передавати складні зв'язки між словами та виконувати більш просунуті завдання.

Роль NLP у голосових помічниках

NLP є фундаментом для створення голосових помічників, таких як Siri чи Alexa. Основні етапи їх роботи включають [1, 2]:

- **Розпізнавання мовлення (ASR)**: перетворення голосових команд у текстовий формат.
- **Аналіз тексту за допомогою NLP**: розуміння намірів користувача та виявлення ключових елементів запиту.
- **Генерація текстової відповіді**: формування релевантної відповіді на запит.
- **Синтез мовлення**: перетворення текстової відповіді у природне мовлення для зручності користувача.

Важливою особливістю є здатність систем навчатися на основі взаємодії з користувачем, що дозволяє адаптувати відповіді до його індивідуальних вподобань.

Еволюція NLP

Розвиток обробки природної мови пройшов кілька етапів [1, 3]:

- **Лінгвістичні правила та статистичні методи**: початкові підходи, такі як "мішок слів" та n-грамні моделі.
- **Приховані марковські моделі (HMM)**: удосконалення статистичних методів для кращого розуміння мовлення.

● **Нейронні мережі та трансформери**: сучасні моделі, як GPT, що революціонізували галузь завдяки здатності генерувати тексти, майже не відрізняються від людських.

Сучасні методи дозволили досягти високої точності у завданнях перекладу та розпізнавання мовлення, відкриваючи нові можливості для взаємодії людини з машиною.

Сучасні тенденції в NLP

Новітні досягнення в області NLP пов'язані з використанням трансформерних моделей [2, 3]:

● **BERT**: модель для двонаправленого аналізу контексту, що покращує розуміння змісту.

● **GPT**: генерація високоякісних текстів з врахуванням довготривалого контексту.

● **CLIP**: мультимодальні системи, що одночасно обробляють текст та зображення.

Завдяки цим моделям стало можливим автоматично створювати резюме текстів та генерувати новий контент, що знаходить застосування у журналістиці та юридичній практиці.

3. СИНТЕЗ МОВЛЕННЯ І ЙОГО ІНТЕГРАЦІЯ В ГОЛОСОВИЙ ПОМІЧНИК

Технологія синтезу мовлення

Процес синтезу мовлення складається з кількох етапів. Спершу текст проходить лінгвістичний аналіз, де визначаються його синтаксичні та семантичні характеристики. Це включає розпізнавання частин мови, пунктуації та інтонаційних пауз. Далі здійснюється фонетична транскрипція, яка перетворює текстові символи у послідовність фонем, відповідних звуків мовлення. Основними методами генерації звукового сигналу є [4, 5]:

● **Конкатенативний синтез**: поєднання записаних фрагментів реального мовлення.

● **Параметричний синтез**: використання статистичних моделей для генерації звуку.

● **Нейронні мережі**: сучасні підходи, як-от WaveNet чи Tacotron, які відтворюють голос з високою природністю.

Завдяки цим методам досягається високий рівень якості синтезованого мовлення, що важливо для природної взаємодії з користувачем.

Інтеграція OpenAI для синтезу мовлення

Використання технологій OpenAI відкриває нові горизонти у синтезі мовлення для голосових помічників. Моделі, такі як GPT-4, здатні генерувати текст з врахуванням контексту та індивідуальних особливостей користувача. Інтеграція цих моделей дозволяє [2]:

● **Покращити якість та релевантність відповідей**: помічник може надавати більш точні та корисні відповіді на запити.

● **Забезпечити адаптивність**: система вчиться з досвіду взаємодії з користувачем, підлаштовуючись під його потреби.

● **Розширити функціональність**: можливість обробки складніших запитів, включаючи контекстну розмову та емоційне забарвлення.

API OpenAI надає розробникам інструменти для легкої інтеграції цих можливостей у свої продукти, забезпечуючи конкурентну перевагу на ринку.

Вибір оптимальних моделей для різних сценаріїв використання

При розробці голосового помічника важливо обрати моделі, які найбільше відповідають конкретним завданням:

● **Для простих запитів**: легкі та швидкі моделі забезпечать швидку відповідь та низьке навантаження на систему.

- **Для складних діалогів**: потужні моделі, як GPT-4, здатні обробляти довгі контексти та забезпечувати глибоке розуміння намірів користувача.
- **Спеціалізовані сценарії**: у випадках, коли потрібна експертна інформація, варто використовувати моделі, натреновані на відповідних наборах даних.

Також необхідно враховувати мову спілкування та культурні особливості аудиторії, забезпечуючи підтримку багатомовності та локалізації.

4. РОЗРОБКА ПРОЕКТУ

Архітектура системи

Архітектура проекту побудована з урахуванням розділення на клієнтську та серверну частини, що забезпечує масштабованість та гнучкість системи. Структура проекту виглядає наступним чином:

```
project/
├── client/
│   ├── app.py
├── src/
│   ├── main.py
│   ├── models.py
│   ├── database.py
│   ├── tasks.py
│   └── routers.py
├── temp/
├── docker-compose.yml
├── Dockerfile
├── requirements.txt
└── .env
```

- **Клієнтська частина**: реалізована за допомогою tkinter для створення графічного інтерфейсу. Файл client/app.py містить код, що дозволяє записувати голосові команди користувача та надсилати їх на сервер для обробки.
- **Серверна частина**: побудована на основі FastAPI, забезпечує обробку запитів від клієнта, зберігання даних у базі даних за допомогою SQLAlchemy, та керування чергою завдань через RQ і Redis.
- **База даних**: використовується SQLite, що спрощує розгортання та тестування системи.
- **Черга завдань**: RQ використовується для асинхронної обробки завдань, таких як перетворення мовлення та генерація відповідей від AI.

Розробка інтерфейсу для взаємодії з користувачем

Інтерфейс користувача відіграє ключову роль у забезпеченні зручності та інтуїтивності взаємодії з голосовим помічником. Використовуючи tkinter, було створено простий та функціональний GUI (рис. 1), який дозволяє:

- **Запис голосових команд**: кнопка "Start Recording" активує мікрофон для запису голосу користувача.
- **Відображення статусу**: лейбли та повідомлення інформують користувача про поточний стан системи (запис, відправка, очікування відповіді).
- **Програвання відповідей**: після отримання відповіді від AI, система відтворює її у голосовому форматі.

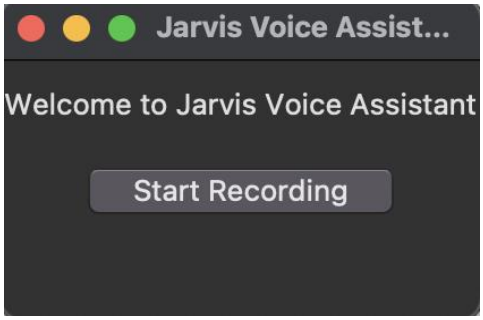


Рисунок 1. Інтерфейс користувача

Цей інтерфейс дозволяє користувачу легко взаємодіяти з системою, не потребуючи технічних знань про внутрішню реалізацію.

Інтеграція з OpenAI та обробка мовлення

Для забезпечення інтелектуальної відповіді використовується API OpenAI. Серверна частина приймає аудіофайл, перетворює його у текст за допомогою модуля speech_recognition, після чого передає запит до моделі GPT через OpenAI API:

```
def process_conversation(conversation_id: int):
    db: Session = SessionLocal()
    conversation = db.query(Conversation).filter(Conversation.id ==
conversation_id).first()

    if conversation:
        messages = [{"role": "user", "content": conversation.user_input}]
        completion = client.chat.completions.create(
            model="gpt-4",
            messages=messages
        )
        assistant_reply = completion.choices[0].message.content.strip()
        conversation.assistant_response = assistant_reply
        db.commit()
    db.close()
```

Отримана від AI відповідь конвертується у аудіо за допомогою модуля pyttsx3 або сервісу pyht, після чого надсилається клієнту для відтворення.

5. ВИСНОВКИ

Інтеграція сучасних технологій обробки природної мови, синтезу мовлення та моделей штучного інтелекту, таких як OpenAI, значно підвищує можливості голосових помічників. Це дозволяє створювати інтелектуальні системи з природною взаємодією, що адаптуються до потреб користувача. Розробка таких проектів вимагає ретельного планування архітектури та розробки зручного інтерфейсу, але в результаті забезпечує ефективні та конкурентоспроможні рішення в галузі штучного інтелекту.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Goodfellow, I., et al. (2016). "Deep Learning"
2. OpenAI API Documentation. Документація OpenAI: веб-сайт: URL: https://platform.openai.com/docs/api-reference/introduction
3. Vaswani, A., et al. (2017). "Attention Is All You Need"
4. Jurafsky, D., & Martin, J. H. (2020). Speech and Language Processing (3rd ed.)
5. Shen, J., Pang, R., Weiss, R. J., et al. (2018). Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions

НАВІГАЦІЯ ПО КОДУ ЗА ДОПОМОГОЮ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ: ДОСЛІДЖЕННЯ ТА РОЗРОБКА ЕФЕКТИВНИХ МЕТОДІВ ІНТЕЛЕКТУАЛЬНОГО ПОШУКУ ТА НАВІГАЦІЇ В ПРОГРАМНОМУ КОДІ

Теслюк В.Ю.[1], Кислий Р.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] tesliuk.volodymyr@lll.kpi.ua, [2] kvrware@gmail.com [0000-0002-8290-9917]

У роботі представлено розробку методології та алгоритмів для інтелектуального пошуку та навігації в програмному коді на основі великих мовних моделей. Запропонована система дозволяє значно полегшити процес орієнтування в коді, забезпечуючи швидкий та точний доступ до потрібних ділянок коду, зменшуючи час на його аналіз і розуміння. В основі розробки лежить застосування великих мовних моделей, що дозволяє враховувати контекст і семантичні зв'язки між різними елементами коду, сприяючи ефективній навігації та пошуку релевантних фрагментів.

Ключові слова: навігація по коду, інтелектуальний пошук, великі мовні моделі, трансформери, семантичний аналіз коду, алгоритми навігації, автоматизація коду, програмна інженерія, машинне навчання в програмуванні, системи рекомендацій для коду, обробка природної мови (NLP), інтелектуальна система пошуку, аналіз програмного забезпечення, контекстуальний пошук, ефективність навігації.

1. ВСТУП

У сучасному програмному забезпеченні, де обсяги коду зростають разом із його складністю, ефективна навігація та пошук по коду стають важливими для підвищення продуктивності розробників і підтримки якості програмних рішень. Програмісти часто стикаються з проблемою швидкого доступу до потрібних фрагментів коду або розуміння його структури, що ускладнює процес розробки та тестування. Традиційні методи навігації, такі як пошук за ключовими словами чи застосування статичного аналізу, не завжди здатні враховувати контекст коду, його залежності та семантичні зв'язки між елементами, що може призводити до неефективної роботи і навіть помилок у процесі розробки.

Завдяки сучасним досягненням у галузі обробки природної мови (NLP), великі мовні моделі (LLM) відкрили нові можливості для розуміння та навігації в програмному коді. Ці моделі здатні враховувати контекст і семантику коду, полегшуючи процес інтелектуального пошуку та надаючи користувачам рекомендації або підказки, що ґрунтуються на більш глибокому розумінні структури і логіки програмного забезпечення. Використання трансформерних моделей дозволяє автоматизувати пошук і спростити взаємодію з великим обсягом коду, що особливо актуально в умовах роботи з великими проектами.

Таким чином, актуальною задачею є розробка ефективної системи інтелектуального пошуку та навігації в програмному коді з використанням великих мовних моделей. Така система не лише прискорить доступ до необхідних фрагментів, але й сприятиме кращому

розумінню структури та логіки програмного коду, зменшуючи кількість помилок та підвищуючи продуктивність розробників.

2. ОГЛЯД ЛІТЕРАТУРИ ТА ІСНУЮЧИХ РІШЕНЬ

Статичний аналіз коду та Multiplier

Перед тим, як розглядати графові бази даних для аналізу коду, важливо зрозуміти основи статичного аналізу коду. Статичний аналіз коду - це процес аналізу програмного коду без його фактичного виконання. Це дозволяє виявляти потенційні помилки, вразливості та інші проблеми на ранніх стадіях розробки програмного забезпечення.

Multiplier – це інструмент статичного аналізу, який може значно підвищити продуктивність аудиту коду. Він працює, зберігаючи артефакти збірки коду в базу даних, яка потім стає доступною через API C++ або Python. Це дозволяє проводити більш точний та комплексний аналіз коду, виявляючи помилки, які могли б бути пропущені при ручному перегляді [1].

Переваги Multiplier:

- Зберігання артефактів збірки в базі даних дозволяє отримати повне уявлення про структуру коду, включаючи визначення функцій, змінні, типи даних та інші важливі деталі.
- Забезпечує постійний доступ до цієї інформації, що спрощує процес аудиту коду та дозволяє автоматизувати деякі його аспекти.
- Дозволяє створювати власні інструменти аналізу коду за допомогою API C++ або Python, розширюючи функціональність Multiplier.
- Приклади інструментів, які можна створити за допомогою Multiplier:
- Пошук викликів функцій всередині списків аргументів макросів.
- Виявлення можливих розбіжних представлень.
- Пошук структур даних, що містять самореференційні покажчики, такі як зв'язані списки та дерева.
- Виявлення "підозрілих" перетворень типів, що передаються аргументам функцій та повертаються з них.

Важливо зазначити: Хоча Multiplier є потужним інструментом статичного аналізу, він не замінює повністю ручний аудит коду. Ручний аудит все ще необхідний для виявлення більш складних помилок та вразливостей.

Графові бази даних для аналізу коду

Графові бази даних добре підходять для представлення структури коду, оскільки вони можуть ефективно моделювати складні взаємозв'язки між різними компонентами коду, такими як модулі, класи, функції та змінні. Деякі системи, такі як CodeBreaker, вже використовують графові бази даних для аналізу коду, демонструючи їх потенціал для покращення навігації по коду. CodeBreaker, наприклад, представляє дані у форматі RDF, де кожен модуль має свій власний граф RDF з документацією про класи, функції та методи модуля [2].

Великі мовні моделі (LLM) для обробки коду

LLM продемонстрували вражаючі можливості в розумінні та генерації коду, відкриваючи шлях до інтелектуальних методів навігації по коду.

CodexGraph – це приклад системи, яка використовує LLM для взаємодії з графовою базою даних коду, дозволяючи розробникам ставити запитання природною мовою та отримувати релевантні фрагменти коду. У CodexGraph основний агент LLM зосереджується на розумінні контексту та генерації запитів природною мовою, тоді як спеціалізований агент перекладу LLM перетворює ці запити на формальні запити до графу [3].

GraphCoder – ще одна система, яка використовує LLM для покращення пошуку коду на рівні репозиторія. GraphCoder використовує граф контексту коду на основі графового пошуку та мовної моделі для підвищення точності завершення коду. GraphCoder також використовує стратегію "записати, а потім перекласти", де первинний агент LLM фокусується на розумінні контексту та створенні запитів природною мовою, які потім передаються спеціалізованому агенту перекладу LLM для перетворення на формальні запити до графу [4].

RAG (Retrieval-Augmented Generation) системи

Системи RAG використовують LLM для отримання відповідної інформації з бази знань. LLM може аналізувати запити природною мовою, формувати відповідні запити до бази даних і отримувати релевантні фрагменти коду.

CodexGraph, наприклад, використовує стратегію "записати, а потім перекласти" для оптимізації взаємодії LLM з базою даних [3].

RepoHyper – це система RAG, яка використовує граф семантики репозиторія (RSG) для представлення глобального контексту та алгоритм пошуку "розширити та уточнити" для вибору відповідних фрагментів коду з RSG [5].

3. АРХІТЕКТУРА ТА ДИЗАЙН СИСТЕМИ

Етап 1: Побудова графової бази даних

На цьому етапі пропонується використовувати Multiplier та Neo4j для створення графової бази даних.

Multiplier буде використовуватися для аналізу програмного коду та вилучення артефактів збірки.

Neo4j – це популярна графова база даних, яка забезпечує ефективне зберігання та обробку графових даних [6].

Схема графової бази даних може включати наступні типи вузлів:

- Модулі: представляють окремі файли коду або бібліотеки.
- Класи: представляють об'єктно-орієнтовані структури даних.
- Функції: представляють блоки коду, які виконують певні дії.
- Змінні: представляють дані, що використовуються в коді.

Типи зв'язків можуть включати:

- Містить: вказує, що модуль містить клас або функцію.
- Наслідує: вказує, що клас успадковує властивості та методи іншого класу.
- Викликає: вказує, що функція викликає іншу функцію.

Запити Cypher можуть бути використані для пошуку інформації в графовій базі даних. Наприклад, наступний запит Cypher [6] може бути використаний для пошуку всіх функцій, які викликає певна функція:

```
MATCH (f:Function)-[:CALLS]->(g:Function)
WHERE f.name = "my_function"
RETURN g.name
```

Етап 2: Обробка з використанням LLM

На цьому етапі LLM будуть використані для обробки коду та покращення навігації.

LLM можуть генерувати ембедінги коду, які представляють семантичне значення коду у вигляді векторів. Ці ембедінги можуть бути використані для пошуку подібних фрагментів коду або для кластеризації коду.

RAG система може використовувати LLM для формування запитів до графової бази даних на основі природної мови. Наприклад, користувач може ввести запит "знайти всі функції, які обробляють файли CSV".

LLM може проаналізувати цей запит та сформувати відповідний запит Cypher для пошуку в графовій базі даних (рис. 1).

Для покращення точності пошуку можна використовувати такі стратегії:

- Розширення графу: передбачає додавання додаткових вузлів та зв'язків до графової бази даних, щоб розширити контекст пошуку. Наприклад, можна додати вузли, які представляють документацію коду або зовнішні бібліотеки.
- Передбачення зв'язків: передбачає використання LLM для передбачення нових зв'язків між вузлами в графовій базі даних. Наприклад, LLM може передбачити, що певна функція викликає іншу функцію, навіть якщо цей зв'язок явно не вказаний в коді.

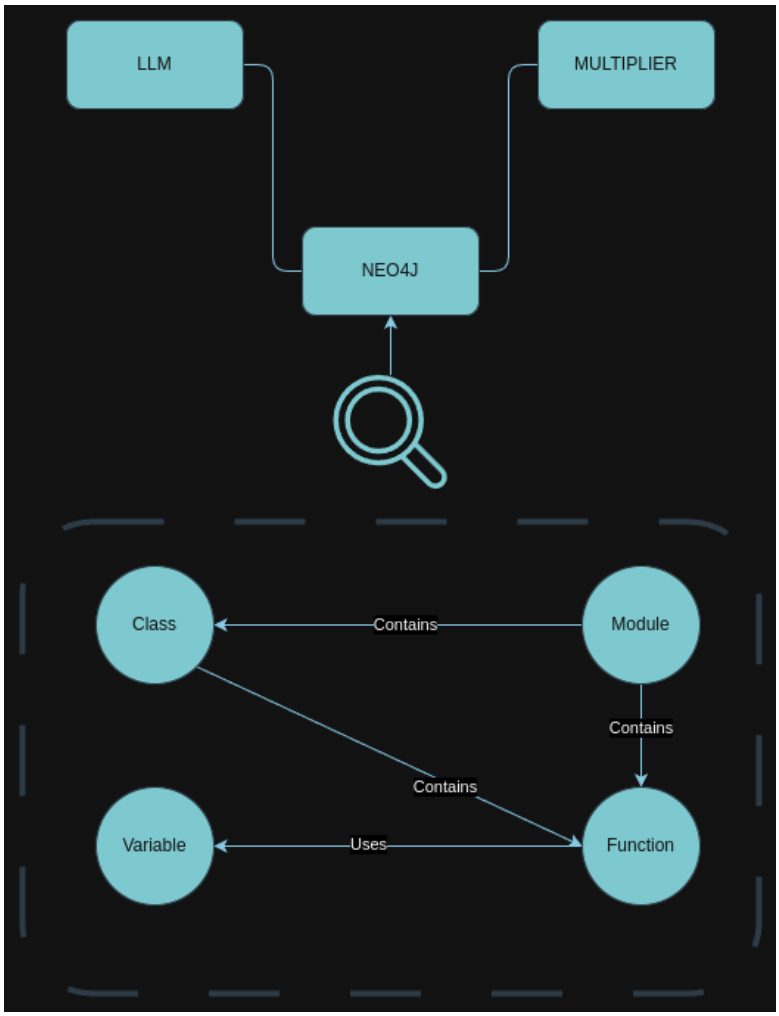


Рисунок 1. Високорівнева архітектура системи

4. ВИКОРИСТАННЯ СИСТЕМИ ДЛЯ НАВІГАЦІЇ ТА АНАЛІЗУ КОДУ

Даний концепт надає розробникам ефективні засоби для навігації по коду, аналізу його структури та автоматизації пошуку інформації. Основні можливості системи забезпечують

комплексний доступ до коду та сприяють інтелектуальному аналізу, що спрощує виконання наступних задач (рис. 2):

Швидкий доступ до структурних залежностей коду. Завдяки графовій базі даних, розробник може легко знаходити інформацію про взаємозв'язки між функціями, класами та модулями. Це особливо корисно для великих проектів, де важливо розуміти взаємодію компонентів.

Інтелектуальний пошук за допомогою природної мови. Система дозволяє розробникам формулювати запити у вигляді звичайних питань, таких як "знайти всі функції для роботи з JSON". Мовна модель автоматично трансформує запит в SQL або Cypher-запит, спрощуючи доступ до потрібних фрагментів без детального знання структурних запитів.

Автоматизоване генерування коду та підказок. На основі наявного коду та запитів, система може не лише знаходити відповідні ділянки, але й пропонувати варіанти для написання нового коду, шаблонів, або коду-аналогу, що значно пришвидшує процес розробки.

Кластеризація подібного коду для оптимізації. Система використовує семантичне групування для пошуку та класифікації фрагментів, які схожі за функціональністю або логікою. Це дає можливість легко виявляти повторювані елементи та оптимізувати код шляхом рефакторингу.

Інтерактивний аудит та аналіз коду. Завдяки постійному доступу до структури коду через API, розробники можуть швидко перевіряти цілісність та відповідність певних стандартів якості коду, знаходити потенційні вразливості та помилки без потреби в ручному аналізі.

Аналіз нових зв’язків між компонентами. Система може передбачити логічні взаємозв’язки між елементами, навіть якщо вони явно не зазначені, що дозволяє краще розуміти приховану структуру коду та сприяє ідентифікації критичних залежностей, корисних для рефакторингу чи усунення "вузьких місць".

Ці можливості системи значно підвищують ефективність роботи розробників, допомагають швидко орієнтуватися в кодових базах будь-якого розміру та забезпечують підтримку стандартів якості коду.

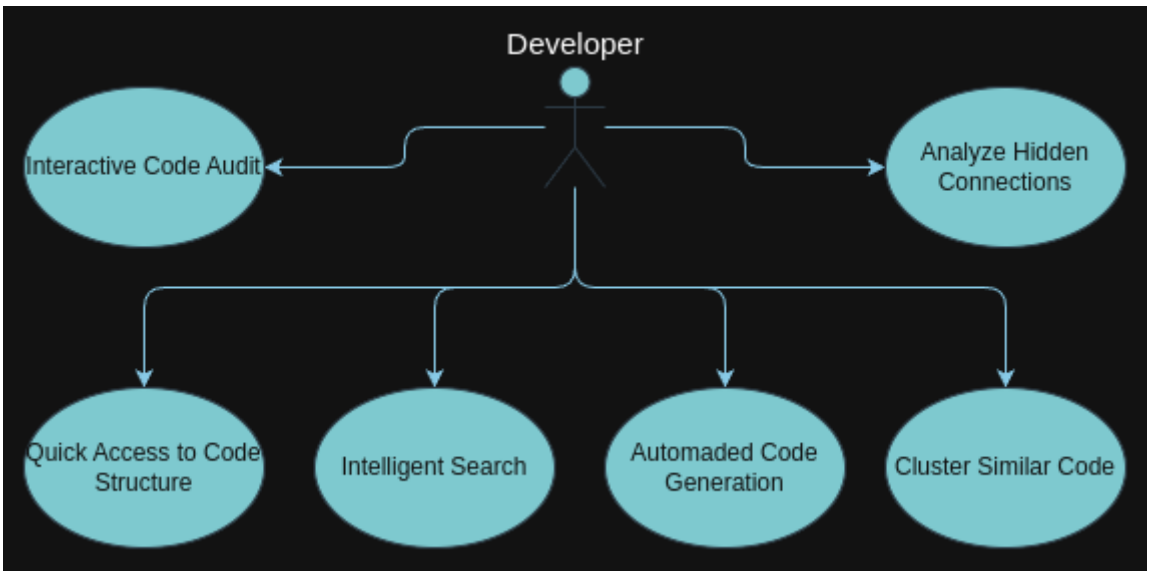


Рисунок 2. Діаграма варіантів використання системи

5. ВИСНОВКИ

У дослідженні було розглянуто сучасні підходи до навігації та аналізу програмного коду, які базуються на статичному аналізі, графових базах даних та великих мовних моделях (LLM). Було проаналізовано можливості та переваги використання статичного аналізу для структурованого доступу до коду, а також графових баз даних, які дозволяють моделювати складні взаємозв'язки між компонентами коду. Окрема увага приділялась можливостям LLM у покращенні процесу навігації та інтелектуального пошуку, що забезпечує швидкий доступ до релевантних фрагментів коду на основі запитів природною мовою.

Одним із ключових етапів стало використання Multiplier для створення графової бази даних, що забезпечує гнучкість і постійний доступ до структурних даних коду. Були розглянуті приклади застосування LLM для генерації коду та RAG-систем для інтелектуальної обробки запитів і пошуку інформації в кодовій базі, що відкриває нові можливості для автоматизації рутинних задач розробників.

Результати дослідження показали, що інтеграція статичного аналізу, графових нейронних мереж та LLM дозволяє створити ефективну систему для навігації, пошуку та аналізу коду. Запропонована система спрямована на підвищення продуктивності розробників, закликана забезпечити більш глибоке розуміння структури коду та спрощення виконання складних аналітичних задач, що робить її перспективним інструментом для роботи з великими кодовими базами.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Multiplier finds more bugs faster. [Електронний ресурс] – Режим доступу до ресурсу: https://github.com/trailofbits/multiplier

2. A Demonstration of CodeBreaker: A Machine Interpretable Knowledge Graph for Code. [Електронний ресурс] – Режим доступу до ресурсу: https://arxiv.org/pdf/2406.06822

3. CODEXGRAPH: Bridging Large Language Models and Code Repositories via Code Graph Databases. [Електронний ресурс] – Режим доступу до ресурсу: https://arxiv.org/pdf/2408.03910

4. GraphCoder: Enhancing Repository-Level Code Completion via Code Context Graph-based Retrieval and Language Model. [Електронний ресурс] – Режим доступу до ресурсу: https://arxiv.org/pdf/2406.07003

5. RepoHYPER: : Search-Expand-Refine on Semantic Graphs for Repository-Level Code Completion. [Електронний ресурс] – Режим доступу до ресурсу: https://arxiv.org/pdf/2403.06095

6. Neo4j. [Електронний ресурс] – Режим доступу до ресурсу: https://neo4j.com/

ДОСЛІДЖЕННЯ КОМП'ЮТЕРНОЇ СИМУЛЯЦІЇ ФІЗИЧНИХ ПРОЦЕСІВ ДЛЯ НЕСТАНДАРТНИХ МОДЕЛЕЙ ПРОСТОРУ

Титарчук В.І.[1], Безносик О.Ю.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] tytarchuk.vladyslav@lll.kpi.ua, [2] beznosyk.oleksandr@lll.kpi.ua [0000-0003-2775-6070]

Метою даного дослідження є аналіз та розробка нових методів фізичної симуляції в нестандартних моделях простору. Наукова новизна роботи полягає у створенні раніше не реалізованих методів симуляції, які якісно покращують адаптацію комп'ютерної симуляції фізичних процесів до нестандартних моделей простору. Практична цінність дослідження полягає у можливості симуляції складних процесів для наукових досліджень і комерційного ПЗ, зокрема 3D-редакторів і комп'ютерних ігор, що забезпечує унікальний користувацький досвід.

Ключові слова: нестандартні моделі простору, комп'ютерна симуляція, фізичні процеси, 3Д-графіка, комп'ютерні ігри.

1. ВСТУП

У сучасному світі комп'ютерне моделювання відіграє важливу роль у багатьох наукових та інженерних дисциплінах, таких як фізика, астрономія та інформатика, а також у розробці комерційного програмного забезпечення, такого як 3D-редактори та комп'ютерні ігри.

У традиційному фізичному моделюванні використовується стандартна модель простору, також відома як евклідовий простір, тобто тривимірний, однорідний і лінійний. Однак з точки зору наукових досліджень і з точки зору виробництва якісних комерційних продуктів має сенс моделювати неевклідовий простір і перебіг фізичних процесів у ньому.

Нестандартні просторові моделі, такі як криволінійні простори, простори з додатковими вимірами, простори зі змінною метрикою тощо, відкривають нові можливості для вивчення різних явищ і вирішення наукових питань. Також вони можуть запропонувати нові унікальні можливості для візуалізації даних і створення нестандартних середовищ для 3D-візуалізації та комп'ютерних ігор. Це пов'язано з тим, що стандартні фізичні та графічні рушії та алгоритми здебільшого призначені для роботи зі звичайними евклідовими просторами.

Враховуючи цю ситуацію, дослідження фізичних процесів у нестандартних просторових моделях в контексті комп'ютерного моделювання стає актуальним і важливим напрямком як для наукових розробок, так і для комерційного програмного забезпечення [1].

2. ДОСЛІДЖЕННЯ ТА АНАЛІЗ НЕСТАНДАРТНИХ МОДЕЛЕЙ ПРОСТОРУ

2.1. Аналіз існуючих нестандартних просторових моделей

В роботі розглядаються чотири найбільш розповсюджені типи нестандартних моделей простору, а саме:

- Багатовимірні простори – широкий клас просторових моделей, що загалом можна охарактеризувати як такі, що мають більшу кількість вимірів, ніж звичні для нас три.
- Топологічно нетривіальні простори – складні геометричні структури, основна особливість яких полягає в тому, що на малих масштабах вони схожі на звичний

евклідовий простір, але мають глобальну «топологію», що задає їхні властивості і значно відрізняється від звичних просторових моделей.

- Фрактальні простори – складні просторові моделі, структура яких може бути описана за допомогою фракталів – геометричних об’єктів, де кожна частина є подібною до цілого, і таким чином нескінченно прогресуючи до зменшення.
- Дискретні простори – це моделі, у яких простір розбивається на дискретні елементи або "кванти". У дискретному просторі кожна точка існує незалежно від інших, а перехід з кожної точки можливий лише в одну з певного числа «сусідніх» для неї точок.

Розглянуто принципи їх функціонування та здійснено класифікацію цих моделей на основі їхніх ключових характеристик, таких як топологія, симетрія, можливості локальних і глобальних змін простору, а також вплив на фізичні процеси.

Проведений аналіз дає можливість зробити висновки про доцільність подальшого вивчення тих чи інших типів моделей в контексті фізичної симуляції та дослідження перебігу фізичних процесів і явищ у них.

На основі результатів проведеного аналізу найбільш доцільною для вивчення було визначено модель топологічно нетривіальніх просторів, як таку, що єдина має частковий тип нестандартного простору і високу інноваційність, оскільки дана модель описує нестандартний простір з локальними елементами стандартного або певним його доповненням і привносить новизну в науку та має потенціал для подальших розробок.

2.2. Специфікація та доопрацювання обраної нестандартної моделі простору

Оскільки топологічно нетривіальні простори являють собою не одну конкретну просторову модель, а цілу доволі обширну групу моделей, постає питання про більш детальну їх специфікацію для подальшого практичного дослідження.

Однією з відомих, але мало досліджених реалізацій нестандартних просторів у прикладних комп'ютерних симуляціях є так звана портальна модель простору, яка забезпечує унікальне поєднання звичайної евклідової геометрії з нестандартними топологічними властивостями, що робить її особливо актуальною для вивчення у даній роботі.

Портали – це концептуальні об'єкти, які з'єднують різні точки простору через своєрідні «вікна», дозволяючи миттєве переміщення або візуальне спостереження між віддаленими точками простору. Найпростішим візуальним представленням порталу може слугувати пласке «вікно» певної форми, що з’єднано з таким же вікном у іншій, віддаленій точці простору. Перетинаючи площину порталу, об’єкт також переміщується у просторі, опиняючись у такому ж положенні відносно вихідного порталу, у якому він мав би бути відносно вхідного у класичній евклідовій моделі (рис. 1).

Портальна модель простору має цілий ряд унікальних характеристик, що роблять її вкрай цікавою для дослідження в контексті даної роботи:

- Портали є яскравим і унікальним прикладом топологічно нетривіального простору, змінюючи глобальну топологію простору, але при цьому зберігаючи локальну його евклідовість. Це дозволяє застосовувати класичні методи симуляції у тих областях, де не взаємодіють портали, зберігаючи простоту візуалізації і взаємодії з простором.
- Портали мають значний потенціал для створення унікальних візуальних ефектів у 3Д-графіці, але питання реалістичної симуляції фізичних процесів у портальній моделі простору майже не розглядається. Таким чином, вибір цієї моделі надає результатам роботи як практичну, так і наукову цінність.
- Під час переміщення через портал об'єкт повинен зберігати свої фізичні властивості, такі як імпульс, енергія та кутове прискорення, що потребує адаптації математичних формул, які описують фізичні взаємодії між базовими фізичними об’єктами, такими як тверді тіла

чи рідини, та власне порталами.

Таким чином, на основі цих факторів можна зробити висновок, що портальна модель простору цілком підпадає під вимоги та специфіку даної роботи, несе у собі практичний зміст та наукову новизну, а тому саме вона була обрана для подальшого дослідження перебігу складних фізичних процесів у ній у контексті комп'ютерної симуляції.

3. РОЗРОБКА МЕТОДІВ СИМУЛЯЦІЇ ФІЗИЧНИХ ПРОЦЕСІВ В ОБРАНІЙ НЕСТАНДАРТНІЙ МОДЕЛІ ПРОСТОРУ

3.1. Базова реалізація портальної моделі простору

Оскільки в рамках однієї роботи розглянути вкрай широкий перелік різноманітних фізичних процесів і явищ неможливо, в даній роботі було прийнято рішення зосередитися на симуляції рідин, які, з одного боку, часто застосовуються в комп’ютерній графіці та пов’язаних галузях, а, з іншого, методи їх адаптації до портальної моделі простору раніше не зустрічалися у жодного іншому проекті або відкритих джерелах.

Щоб забезпечити можливість такого дослідження, перш за все необхідно реалізувати базову симуляцію портальної моделі простору, що буде сумісна з найпростішими елементами фізичної симуляції, такими як рухомі тіла та віртуальні камери. Це можливо цілком успішно зробити на основі вже відомих методів адаптації фізичної симуляції до нестандартних моделей простору (рис. 2).

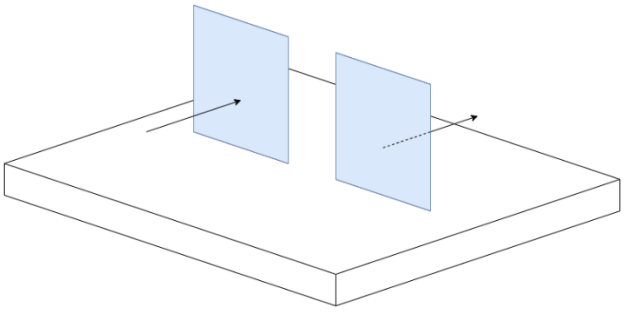


Рисунок 1. Базова схема роботи порталів у портальній моделі простору

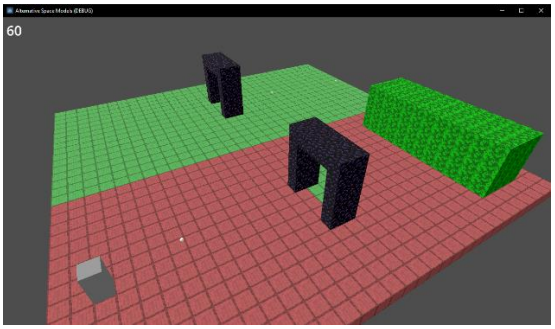


Рисунок 2. Коректно працююча тестова сцена з базовою реалізацією порталів

3.2. Симуляція рідин спрощена

Для того, щоб створити максимально реалістичний ефект поверхні рідини, зазвичай використовується спеціальна графічна програма – шейдер, що реалізовуватиме наступні функції:

- Анімація карти нормалей рідини для створення ефекту хвильок на її поверхні.
- Анімація вертексної сітки рідини для створення ефекту великих хвиль.
- Реалізація ефекту заломлення світла і викривлення об'єктів крізь поверхню рідини.

Така реалізація дозволяє отримати як стилізовану, так і доволі реалістичну поверхню води, таким чином забезпечуючи потреби розробників у різних ситуаціях, при цьому не використовуючи складну повноцінну симуляцію рідини [3].

Проте у випадку з портальною моделлю простору виникає проблема – суцільна поверхня рідини, що перетинає площини обох порталів, має помітний «розрив» по площині кожного з порталів (рис. 3, а).

Таким чином, основною проблемою наявних методів у адаптації спрощеної симуляції рідин до портальної моделі простору є виникнення «розривів» у єдиній площині рідини, у випадку перетину нею площини порталу. Щоб виправити це, необхідно змінити підхід до

організації взаємодії рідини і порталу, а також логіку роботи самої площини рідини таким чином, щоб край поверхні відразу за порталом був синхронізований з краєм поверхні відразу перед порталом.

Першим кроком в цьому буде розділення єдиної водної поверхні на декілька окремих – розміщених перед порталом і за порталом – таким чином, щоб їхні краї сходилися якраз на поверхні порталу. Це пояснюється тим, що при застосуванні єдиної поверхні портали загалом можуть знаходитися в будь-яких її точках, тобто мати випадкові UV-координати. А оскільки стан анімації водної поверхні залежить від UV-координат конкретної точки, це призводило б до задачі синхронізації випадкових точок. Після розділення ж однієї поверхні рідини на дві, суміщені краями по лінії поверхні порталу, проблема зводиться до задачі синхронізації анімації на проміжку UV-координат (0, 0 – 0, 1) з проміжком (1, 0 – 1, 1), що є набагато більш простою задачею.

Другим кроком буде застосування спеціальної синхронізуючої функції для обробки координат точки на поверхні рідини перед передачею їх у шейдер. Ця функція загалом може бути довільною, однак з метою синхронізації вищевказаних проміжків, вона має видавати однаковий результат при вхідному параметрі (0, X) і (1, X). Також, для спрощення, це може бути функція лише від одного аргументу, яка буде застосовуватися окремо для кожної координати.

Одним з найпростіших прикладів такої функції є наступна модульна функція:

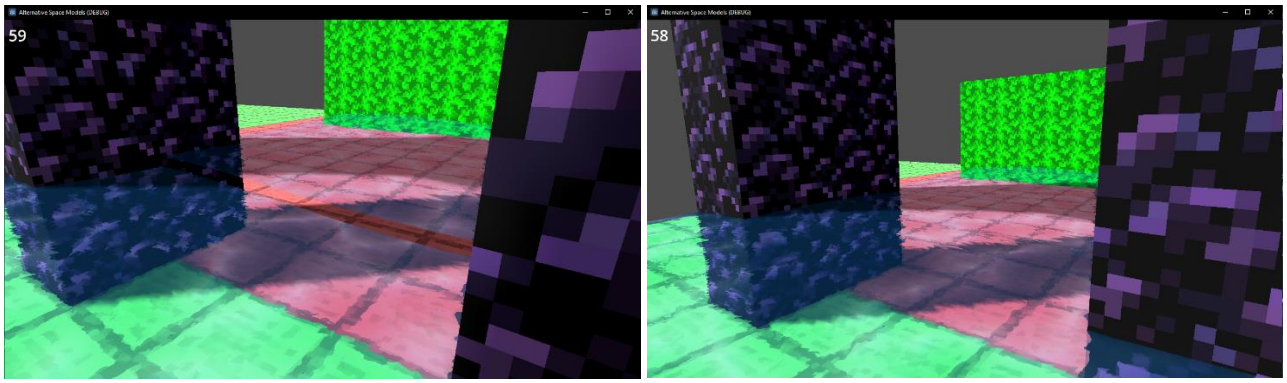
$$f(x) = abs(0.5 - x).$$

Проте така функція є занадто лінійною, що призведе до погіршення якості ефекту хвиль на усій поверхні рідини. Тому після деяких експериментів було підібрано більш оптимальну і складну синусоїдну функцію:

$$f(x) = sin(x * Pi * 2).$$

Така функція дозволяє одночасно і вирішити проблему з розривами поверхні рідини, і зберегти високу якість анімації хвиль на її поверхні. Причому, оскільки аналогічна косинусоїдна функція в такій ситуації також відповідає усім необхідним вимогам, це дає змогу застосовувати одну з них до U-координати поверхні рідини, а іншу – до V-координати, таким чином роблячи шаблон анімації поверхні рідини більш випадковим і реалістичним, що є явною перевагою з точки застосування даного методу у комерційній розробці ПЗ.

Також, в результаті практичної реалізації було підтверджено дієвість розробленого метода щодо якісної та ефективної адаптації спрощеної комп’ютерної симуляції рідин до портальної моделі простору (рис. 3, б).



а) б)

Рисунок 3, а) «Розрив» у пласкій поверхні рідини по площині порталу,
б) «Безшовне» проходження поверхні рідини через портал

3.3. Симуляція рідин повноцінна

Більш складною є реалізація повноцінної симуляції рідини. Існує декілька конкретних методів такої реалізації, що відрізняються між собою певними деталями, але найбільш розповсюдженим з них є так званий метод Гідродинаміки згладжених частинок (SPH) [4].

Для імплементації цього методу потрібно створити програмний код, що реалізуватиме симуляцію маси рідини на основі частинок. В результаті отримуємо доволі реалістичну симуляцію поведінки частинок, візуалізовану за допомогою маленьких сфер, логіка взаємодії яких між собою та з навколишнім середовищем відбувається за методом SPH і в цілому відповідає уявленню про поведінку рідин.

Проте при контакті з елементами неевклідового простору, а конкретно – з поверхнею порталу, виникає явна проблема у вигляді хаотичного «розриву» між масами рідини, що знаходяться попереду і позаду порталу (рис. 4, а).

Проблема з повноцінною симуляцією рідини є дещо більш складною і полягає в тому, що хоча візуально частинки рідини, розміщені безпосередньо перед і за порталом і здаються наближеними, насправді в реалізованій сцені вони знаходяться на значному віддалені одна від одної, що призводить до неможливості їх спільної обробки фізичним рушієм та шейдером і, як наслідок, до виникнення помітних розривів по площині порталу у масі рідини.

Отже, щоб вирішити цю проблему, необхідно реалізувати механізм, який дозволить частинкам рідни «підтримувати контакт» з іншими частинками через площину порталу.

Для цього пропонується наступний оригінальний метод:

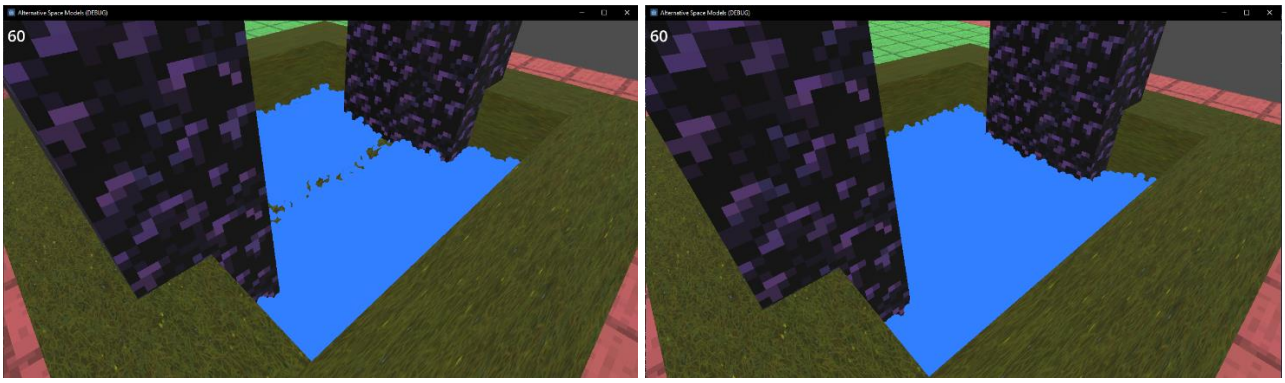
1. Оберемо певну невелику відстань, наприклад, рівну розміру однієї частинки рідини, та будемо відслідковувати частинки, що перебувають у цьому проміжку навколо порталу.

2. Коли кожна з частинок рідини перетинає портал, створимо її копію біля вхідного порталу, положення якої відносно вхідного порталу буде однаковим з положенням оригіналу відносно вихідного порталу.

3. Оновлюватимемо положення таких частинок-копій відповідно з оригіналами кожен кадр фізичної симуляції, поки вони не вийдуть за межі зони відслідковування.

Як видно зі схеми, при такому підході утворюється «зона перекриття» поруч з площиною порталу, в якій дублюючі сфери надійно покривають площину порталу, тим самим не допускаючи виникнення видимих розривів у масі рідини і, таким чином, дозволяючи вирішити проблему взаємодії рідини з портальним простором (рис. 5) [5].

Аналогічним чином реалізуємо розроблений метод на практиці, що дасть змогу перевірити його дієвість та впевнитись у коректному вирішенні проблеми (рис. 4, б) [6].



а) б)

Рисунок 4, а) «Розрив» у масі рідини по площині порталу,
б) Оновлена та коректна взаємодія рідини з порталом

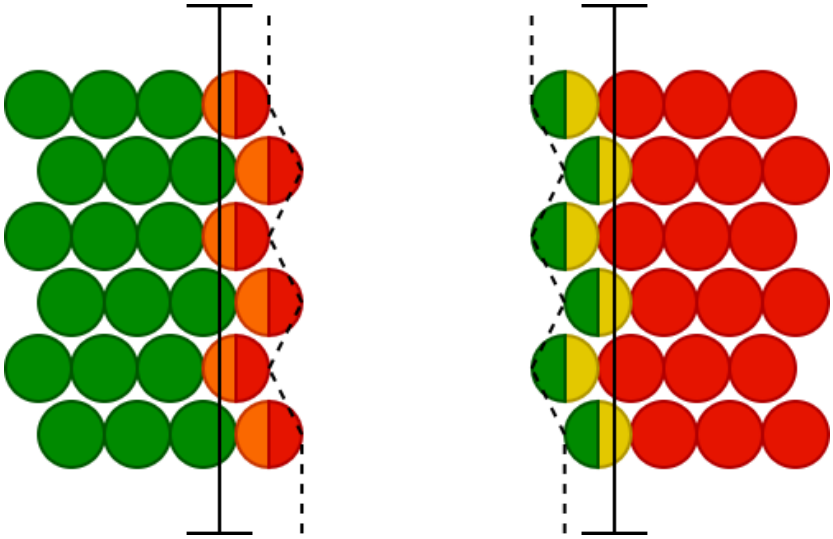


Рисунок 5. Оновлена схема взаємодії частинок з поєднаними порталами

4. ВИСНОВКИ

В роботі було проведено детальний аналіз різних нестандартних моделей простору, актуальних як у наукових дослідженнях, так і в прикладних сферах техніки, що дозволило обґрунтовано обрати модель для практичної реалізації. Також було досліджено загальноприйняті методи комп’ютерної симуляції фізичних явищ і проаналізовано їхню адаптацію для умов портальної геометрії простору.

Фінальним етапом роботи стала розробка нових методів адаптації комп’ютерної симуляції фізичних процесів і явищ до нестандартних моделей простору. Розроблені методи демонструють значно кращі результати у адаптації симуляції складних фізичних процесів і явищ до портальної моделі простору, ніж вже існуючі. Зокрема, розроблені методи дозволяють провести якісну адаптацію спрощеної та повноцінної симуляції рідин до портальної моделі простору, що раніше не було описано в жодній іншій роботі чи відкритому проекті.

Розроблені методи не лише забезпечують високу точність та продуктивність симуляції, а й відкривають нові можливості для комп'ютерного моделювання у прикладних сферах, де нестандартна геометрія простору може мати практичне значення, зокрема у розробці 3D-редакторів та ігрових рушіїв.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Greenberg M. J. Euclidean and non-Euclidean geometries. 4-те вид. New York : W.H. Freeman, 2008. 637 с.

2. Millington I. Game Physics Engine Development. CRC Press, 2010. URL: https://doi.org/10.1201/b13170 (дата звернення: 18.10.2024).

3. OpenGL 4 Shading Language Cookbook: Build high-quality, real-time 3D graphics with OpenGL 4.6, GLSL 4.6 and C++17, 3rd Edition. Packt Publishing, 2018. 472 с.

4. Liu G. R., Liu M. B. Smoothed Particle Hydrodynamics. *World Scientific*, 2003. URL: https://doi.org/10.1142/5340 (дата звернення: 25.10.2024).

5. Real-time fluid simulation with adaptive SPH / H. Yan та ін. *Computer Animation and Virtual Worlds*. 2009. Т. 20, № 2-3. С. 417–426. URL: https://doi.org/10.1002/cav.300 (дата звернення: 07.10.2024).

6. Stellingwerf R. F. Impact modeling with smooth particle hydrodynamics. Los Alamos, NM : Los Alamos National Laboratory, 1993. 12 с.

ОЦІНКА СУМІСНОСТІ КАНДИДАТА ТА КОМПАНІЇ НА ОСНОВІ СЕМАНТИЧНОГО АНАЛІЗУ ІСТОРІЇ НАЙМІВ ТА РЕЗЮМЕ

Товстенко Н.А.[1], Казаков В.В.[2], Письменний І.О.[3]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] tovstenko.nazarii@lll.kpi.ua, [2] kazakov.vladyslav@lll.kpi.ua,
[3] ihor.pismennyy@gmail.com [0000-0001-7648-2593]

У роботі описано розробку методології та алгоритмів для оцінки сумісності кандидатів і компаній на основі семантичного аналізу текстових даних резюме та історії наймів. Запропонована система дозволяє автоматизувати процес підбору кадрів, підвищуючи його точність і зменшуючи суб'єктивність під час аналізу резюме. В основі розробки лежить застосування моделей на основі трансформерів для обробки тексту, що дозволяє враховувати контекст і семантичну подібність між вакансіями та компетенціями кандидатів.

Ключові слова: підбір персоналу, семантичний аналіз, трансформерні моделі, аналіз ринку праці, автоматизація HR.

1. ВСТУП

На сучасному ринку праці технологічний розвиток та глобалізація створюють нові виклики для компаній і кандидатів. Пошук і підбір кваліфікованих кадрів стає важливою умовою для досягнення конкурентних переваг, а також для розвитку бізнесу та забезпечення стійкості компаній. Одночасно, зростаюча конкуренція серед кандидатів і вимоги до різноманітних навичок призводять до необхідності більш точного та ефективного відбору персоналу. Однак традиційні методи відбору, такі як ручне сортування резюме або стандартні інтерв'ю, мають ряд обмежень, включаючи високу суб'єктивність і обмежену здатність аналізувати великі обсяги текстових даних.

Використання сучасних методів машинного навчання, зокрема семантичного аналізу та обробки природної мови (NLP), відкриває нові можливості для автоматизації процесів відбору персоналу. Завдяки трансформерним моделям, таким як BERT, RoBERTa та інші, можна автоматично розпізнавати релевантні навички кандидата, навіть якщо вони описані різними термінами або фразами. Це дозволяє знизити ризик упущення перспективних кандидатів через неточне використання мови та значно підвищити точність процесу підбору персоналу. Таким чином, актуальність цієї роботи полягає у створенні ефективної системи автоматизованої оцінки сумісності, що враховує семантичні зв'язки між навичками, описаними в резюме, та вимогами вакансій.

Метою роботи є розробка методології та алгоритмів, які забезпечують ефективну оцінку сумісності кандидата та компанії на основі аналізу текстових даних резюме та історії наймів, застосовуючи сучасні методи обробки природної мови. Основні завдання включають збір і передобробку даних, розробку системи для виокремлення ключових навичок і оцінку сумісності кандидатів з вакансіями на основі семантичного аналізу.

2. ДОСЛІДЖЕННЯ ПРОЦЕСУ НАЙМУ

2.1. Особливості сучасного процесу найму

Сучасний процес найму суттєво змінився в порівнянні з минулим десятиліттям. Одним із найважливіших трендів є значне зростання обсягу заявок, що поступають на вакансії, особливо в технологічних галузях та сферах, де попит на кваліфікованих працівників перевищує пропозицію. Це ускладнює роботу рекрутерів, адже обробка великої кількості інформації вручну є дуже витратною за часом. Крім того, традиційні методи відбору часто не дозволяють враховувати нюанси професійного досвіду та особистих якостей кандидатів, що може негативно впливати на якість остаточного вибору.

Рекрутери стикаються з кількома ключовими проблемами в процесі найму:

- Суб’єктивність оцінки: рішення про вибір кандидатів часто піддаються суб'єктивним факторам, що може знижувати точність оцінки відповідності кандидата вакансії.
- Недостатня ефективність: звичайні процеси ручної обробки резюме потребують значних ресурсів і часу, що ускладнює швидке заповнення вакансій.
- Можливі упередження: людський фактор нерідко призводить до упередженості, яка може вплинути на рівень об'єктивності оцінки кандидатів.
- Нестача автоматизації: у багатьох компаніях процеси найму залишаються ручними або лише частково автоматизованими, що призводить до збільшення витрат на підбір персоналу.

2.2. Технології для аналізу ринку праці

Технології, що забезпечують автоматизацію найму [1], мають значний вплив на процеси підбору персоналу. Із розвитком інструментів NLP, машинного навчання, великих мовних моделей та семантичного аналізу з’явилася можливість детальніше і точніше аналізувати текстові дані з резюме і супровідних документів кандидатів. До таких технологій відносяться:

- Алгоритми машинного навчання – дозволяють розпізнавати патерни в даних, що характеризують успішних кандидатів, а також застосовувати методи класифікації для відбору резюме, які відповідають визначеним критеріям.
- Обробка природної мови (NLP) – забезпечує аналіз текстових даних резюме та вакансій, включаючи визначення ключових слів, фраз і контексту.
- Семантичний пошук – технологія, що дозволяє знаходити контекстуально релевантні документи та порівнювати їх між собою. Цей підхід, заснований на алгоритмах пошуку смислової подібності, зокрема, векторних уявлень тексту (word embeddings) [2], дозволяє оцінювати не лише точні збіги, а й семантичні відповідності між текстами.
- Контекстуальні моделі – серед них, BERT, RoBERTa, GPT-3 тощо. Ці моделі здатні обробляти текстові дані, враховуючи контекст та інші фактори, що дозволяє інтерпретувати фрази та ключові слова в їх повному значенні.

2.3. Методи обробки текстових даних у рекрутингу

Автоматизація найму і технології, що її забезпечують, суттєво впливають на ефективність підбору персоналу. З розвитком інструментів обробки природної мови, машинного навчання, великих мовних моделей і семантичного аналізу з’явилися методи для точнішого аналізу текстових даних із резюме і вакансій. Серед цих технологій виділяються алгоритми машинного навчання, які допомагають виявляти шаблони у даних і класифікувати резюме; методи обробки природної мови (NLP), які забезпечують детальний аналіз текстів, від пошуку ключових слів до розуміння контексту; технології семантичного пошуку, що дозволяють оцінювати відповідність текстів не лише за збігом слів, а й за їхньою семантичною

подібністю; та контекстуальні моделі, здатні враховувати повний контекст текстових даних для точної інтерпретації фраз і понять.

2.4. Використання історії наймів для прогнозування успішності

Аналіз історії попередніх наймів дозволяє виявляти фактори, що впливають на успішність підбору кандидатів для певних вакансій. Такий аналіз включає кілька етапів: збір інформації про минулі найми, дослідження факторів успішності, моделювання прогнозів і подальше використання отриманих даних у процесі відбору. Цей підхід дозволяє компаніям виявляти закономірності, які позитивно впливають на продуктивність кандидатів і, таким чином, оптимізувати процес найму.

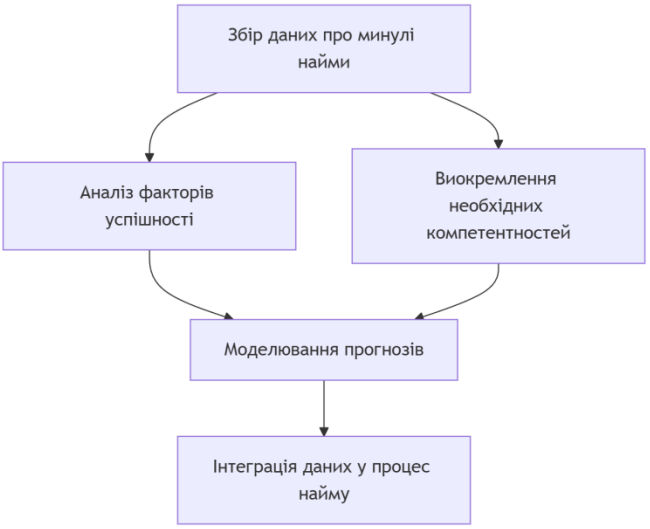


Рисунок 1. Діаграма процесу підбору персоналу з використанням історії попередніх наймів

3. РОЗРОБКА СИСТЕМИ ОЦІНКИ СУМІСНОСТІ КАНДИДАТА ТА КОМПАНІЇ НА ОСНОВІ СЕМАНТИЧНОГО АНАЛІЗУ ІСТОРІЇ НАЙМІВ ТА РЕЗЮМЕ

3.1. Збір та передобробка даних

Перший етап розробки системи передбачає створення великого корпусу текстових даних, що складаються з описів вакансій і резюме кандидатів [3]. Основні джерела включають популярні платформи для пошуку роботи, такі як Djinni [3], robota.ua, work.ua та DOU Робота. Ці ресурси дозволяють зібрати широкий спектр текстових даних про професійні вимоги, обов'язки та необхідні навички, що забезпечує різноманітність і актуальність корпусу.

Після збору даних проводиться їх детальна передобробка, яка включає кілька ключових кроків:

1. Очищення тексту: На цьому етапі видаляються зайві символи, такі як HTML-теги, зайві пробіли, спеціальні символи та непотрібні розділові знаки. Це дозволяє отримати "чисті" текстові дані, які легше піддаються аналізу.
2. Анонімізація даних [3]: Очистка тексту від персональних даних які можуть бути чутливими та охороняються законом.
3. Токенізація: Текст розділяється на окремі слова або фрази, що дозволяє системі більш ефективно обробляти інформацію. На цьому етапі використовуються спеціалізовані словники та алгоритми для розпізнавання складних термінів.
4. Лематизація та стемінг: Цей етап передбачає приведення слів до їх базових форм (лем).

Це знижує варіативність у тексті та забезпечує стандартизацію. Наприклад, слова "розробник", "розробляє" та "розробка" зводяться до основної форми "розробляти".

5. Видалення стоп-слів: У текстах можуть зустрічатися слова, що не несуть важливого змісту для аналізу (наприклад, "і", "або", "на"). Їх видаляють для зменшення обсягу даних і зосередження на важливих ключових словах.
6. Нормалізація числових даних: Деякі вакансії можуть містити числову інформацію, таку як кількість років досвіду, вимоги до знання певних технологій тощо. Ці дані виділяються окремо для подальшого використання у порівнянні.

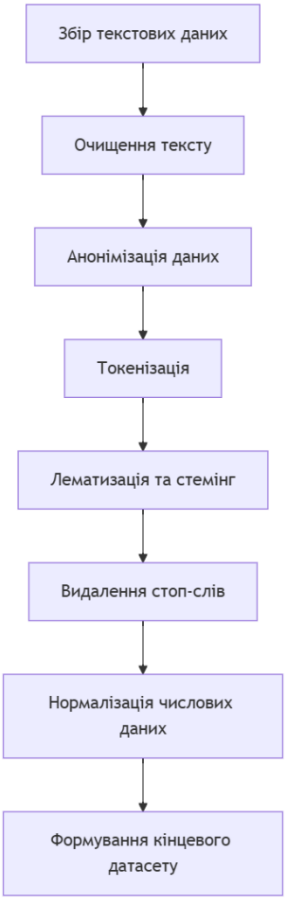


Рисунок 2. Візуалізація процесу збору та обробки даних

Після завершення цих етапів дані зберігаються у структурованій базі для подальшого аналізу. Така стандартизація значно полегшує наступні етапи обробки тексту та виокремлення релевантних навичок.

3.2. Побудова системи виокремлення навичок із опису вакансій та резюме

Для реалізації автоматичного виокремлення ключових компетенцій використовується підхід на основі трансформерних моделей, таких як BERT [4]. Такі моделі дозволяють враховувати контекст слів, що підвищує точність у виявленні навичок. Основні етапи роботи системи включають:

1. Навчання моделі: Використовується попередньо натренована модель BERT, адаптована для завдання виявлення навичок. Для цього модель додатково тренується на корпусі текстів з описами вакансій та резюме, що забезпечує здатність розпізнавати специфічні терміни.

2. Виявлення релевантних фраз: Модель використовує механізм "уваги" (attention), щоб визначати найбільш важливі частини тексту. Фрази, які відповідають певним типам навичок або компетенцій, виокремлюються для подальшого збереження.
3. Збереження результатів у базі даних: Виокремлені навички зберігаються у структурованій базі для подальшого порівняння та оцінки сумісності кандидатів з вакансіями.

3.3. Оцінка відповідності резюме вакансіям на основі виокремлених навичок

Після виокремлення ключових навичок основним завданням є оцінка відповідності кандидатів до вакансій. Система порівнює векторні представлення навичок кандидатів [5] і вимог вакансій, використовуючи, наприклад, метод косинусної схожості для визначення рівня відповідності. Це дозволяє не тільки враховувати точні збіги, а й оцінювати семантичну подібність між навичками [6]. Наприклад, якщо вакансія вимагає "управління проектами", а кандидат зазначає "координацію роботи команди", система може вважати ці навички схожими.

Для кожної пари "кандидат-вакансія" обчислюється інтегральний показник сумісності, який є середнім значенням схожості між виокремленими навичками. Цей показник дозволяє автоматизувати процес відбору, забезпечуючи швидкий і точний аналіз.

3.4. Оцінка ефективності розробленої системи

Для оцінки роботи системи використовуються кілька ключових метрик, що забезпечують кількісну оцінку точності й якості аналізу [7]:

- Точність (Precision): Визначає відсоток релевантних навичок, які система правильно розпізнала з-поміж усіх, що були виділені. Висока точність свідчить про здатність системи уникати зайвих, нерелевантних навичок.
- Повнота (Recall): Вимірює відсоток релевантних навичок, які система розпізнала з усіх наявних релевантних навичок у тексті. Висока повнота означає, що система виявляє більшу частину потрібних навичок, не пропускаючи важливих даних.
- F-score: Це інтегральна метрика, що об’єднує точність і повноту, забезпечуючи загальну оцінку ефективності. F-score особливо корисний, коли важливо знайти баланс між точністю та повнотою.
- Семантична точність: Ця метрика оцінює здатність системи розуміти значення навичок у контексті. Семантична точність вимірюється шляхом порівняння виділених навичок із синонімічними або подібними термінами у вимогах вакансії, що дозволяє виявити рівень відповідності не лише за текстовими збігами, але й за смисловою подібністю.

Рисунок 3. Візуалізація процесу оцінки точності системи

Під час тестування система проходить оцінку за цими метриками на контрольному наборі даних. Результати тестування дозволяють виявити сильні та слабкі сторони системи, а також провести порівняння з альтернативними методами, що використовують традиційний пошук за ключовими словами або базові моделі семантичної подібності.

4. ВИСНОВКИ

У дослідженні проведено детальний аналіз сучасних методів оцінки сумісності кандидатів із вимогами вакансій, який базуються на семантичному аналізі текстових даних. Було охоплено різні підходи, актуальні як для наукових досліджень, так і для практичного застосування в задачах підбору персоналу. Також проаналізовано ключові моделі обробки природної мови, включаючи статичні векторні моделі та контекстуальні трансформери, що дозволило обґрунтувати обрані моделі для практичної реалізації системи автоматизованого підбору персоналу.

Ключовим етапом дослідження стало збирання і підготовка великого корпусу текстових даних, що містять опис вакансій та резюме кандидатів. Зібраний корпус даних охоплює різноманітні галузі та професії, що забезпечує можливість розробки універсальної системи для оцінки сумісності. Застосовані методи очищення, лематизації, токенізації та нормалізації даних підвищують якість корпусу, зменшуючи шум і забезпечуючи єдність структури для подальшого аналізу.

Також було розроблено методологію оцінки сумісності кандидатів і вакансій, яка базується на порівнянні векторних представлень навичок. На основі зібраних даних розробляється система виокремлення навичок із текстів вакансій та резюме з використанням контекстуальних моделей, таких як BERT. Система потенційно здатна розпізнавати складні терміни та професійні синонімічні вирази, враховуючи контекст і значення слів. Це дозволяє автоматизувати здійснювати автоматичний пошук навичок і компетенцій, що значно підвищує точність порівняння навичок кандидатів із вимогами вакансій.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. New age recruitment trends and tools [Електронний ресурс] – Режим доступу до ресурсу: https://talenttitan.com/new-age-recruitment-trends-and-tools.

2. What are word embeddings? [Електронний ресурс] // 2024 – Режим доступу до ресурсу: https://www.ibm.com/topics/word-embeddings.

3. Drushchak N., Romanyshyn. M. Introducing the Djinni Recruitment Dataset: A Corpus of Anonymized CVs and Job Postings, 2024.

4. Devlin J., Chang M., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018.

5. Dar G., Geva M., Gupta A., Berant J. Analyzing Transformers in Embedding Space, 2022.

6. Input Embedding Sublayer in the Transformer Model [Електронний ресурс]. – 2024. – Режим доступу до ресурсу: https://medium.com/image-processing-with-python/input-embedding-sublayer-in-the-transformer-model-7346f160567d.

7. How do you evaluate the quality and accuracy of semantic annotation and extraction? [Електронний ресурс] – Режим доступу до ресурсу: https://www.linkedin.com/advice/0/how-do-you-evaluate-quality-accuracy-semantic.

ВИКОРИСТАННЯ LLM АГЕНТІВ В ПЕРШІЙ ЛАНЦІ МЕДИЧНОЇ ДОПОМОГИ

Тринус Н.В.[1], Кислий Р.В.[2]

[1,2]Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] trynus.nikita@lll.kpi.ua, [2] kvrware@gmail.com

Метою даного дослідження є розробка мультиагентної системі на базі великих мовних моделей, яка здатна виконувати роль першої ланки медичної допомоги. Наукова новизна роботи полягає у використанні мультиагентного підходу, який досягає більшої якості та покриває більшу кількість проблем з майже необмеженим способом розширення. Практична цінність дослідження полягає у використанні системи як першої ланки медичної допомоги замість телемедицини.

Ключові слова: агенти, мультиагентні системи, великі мовні моделі, медицина, перша ланка медичної допомоги, мультимодальність.

1. ВСТУП

Попит на доступну та ефективну первинну медичну допомогу ще ніколи не був таким актуальним. Традиційні методи, такі як особисті візити до лікаря, часто передбачають тривалий час очікування, значне навантаження на медичних працівників і матеріально-технічні проблеми для пацієнтів. Телемедицина, яку колись проголошували вирішенням цих проблем, виявилася обмеженою. Зазвичай вона не пропонує достатнього контексту для точної діагностики, що призводить до збільшення кількості повторних візитів і, зрештою, не в змозі зменшити системне навантаження на послуги охорони здоров'я.

Недавні досягнення у великих мовних моделях (LLM) [1] показали багатообіцяючі можливості для розширення першої ланки медичної допомоги шляхом автоматизації попереднього оцінювання пацієнтів, скорочення часу очікування та надання цінної підтримки перевантаженому медичному персоналу. LLM здатні аналізувати скарги пацієнтів, розпізнавати симптоми та запропонувати потенційні діагнози. Однак, незважаючи на їх високі вміння розуміння мови(семантики), LLM схильні до галюцинацій — генерування неправильної або оманливої інформації, коли надається недостатній контекст або зовсім його відсутність. Або LLM намагається дати відповідь на питання, на яке не знає відповіді. Це становить серйозну проблему під час застосування LLM у медицині, де точність і безпека пацієнтів є першочерговими.

У цій роботі досліджується розробка багатоагентної системи LLM, призначеної для першої ланки медичної допомоги. Запропонована система спрямована на подолання розриву між пацієнтами та постачальниками медичних послуг шляхом полегшення первинної оцінки та збору інформації. Завдяки інтеграції кількох агентів LLM із певними ролями система може мінімізувати ризик галюцинацій, використовуючи надійні механізми пошуку даних і контексту для підвищення надійності та безпеки.

2. ДОСЛІДЖЕННЯ ПІДХОДІВ ВИКОРИСТАННЯ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ У МЕДИЦИНІ

2.1. Розвиток спеціалізованих Великих Мовних Моделей у розумінні медицини.

Великі мовні моделі (LLM) продемонстрували значний потенціал у трансформації ландшафту медичної допомоги та телемедицини. Розвиток цих моделей можна простежити від універсальних LLM, таких як **GPT-3** [1] і **PaLM** [3], до спеціалізованих моделей, налаштованих на інструкції, адаптованих для медичних застосувань. Важливою віхою в цій еволюції є розробка **Med-PaLM**[2], передової моделі, розробленої для усунення недоліків попередніх LLM у застосуванні до охорони здоров'я (рис. 1).

Med-PaLM, варіація моделі PaLM із 540 мільярдами параметрів, було налаштовано з інструкціями з акцентом на узгодження з медичними знаннями. Датасет включає у себе спеціалізований набір даних і тестів, таких як MultiMedQA, що є складається з різноманітних наборів завдань із відповідями на медичні запитання, поєднуючи структуровані набори даних і нещодавно створений HealthSearchQA.

Example of USMLE-style question

A 32-year-old woman comes to the physician because of fatigue, breast tenderness, increased urinary frequency, and intermittent nausea for 2 weeks. Her last menstrual period was 7 weeks ago. She has a history of a seizure disorder treated with carbamazepine. Physical examination shows no abnormalities. A urine pregnancy test is positive. The child is at greatest risk of developing which of the following complications?

A. Renal dysplasia

B. Meningocele

C. Sensorineural hearing loss

D. Vaginal clear cell carcinoma

Рисунок 1. Приклад того, як виглядає датасет для тренування Med-PaLM

Слід зазначити, що Med-PaLM 2 досягає точності 86.5% на датасеті MedQA [4] (рис. 2).

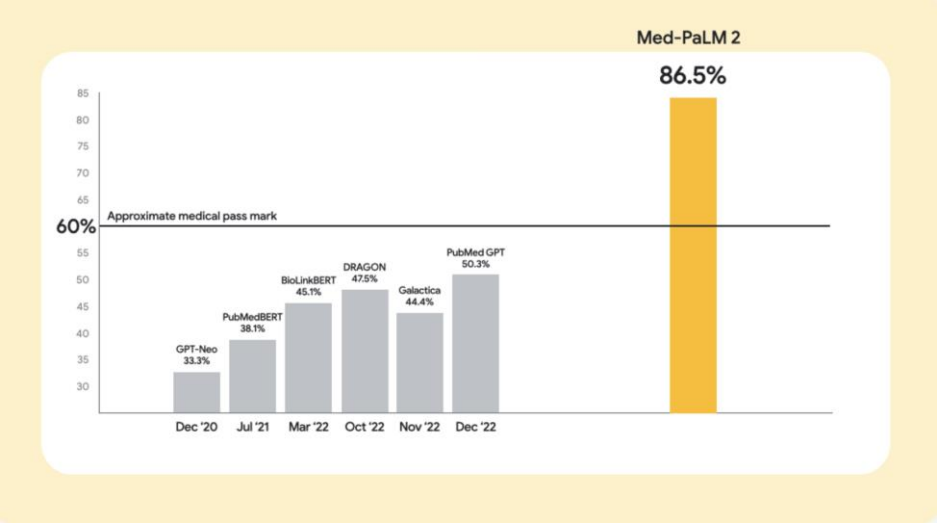


Рисунок 2. Порівняння Med-PaLM2 з іншими підходами [3]

Незважаючи на прогрес, можливості Med-PaLM все ще обмежені. Людські оцінки висвітлили постійні проблеми в таких сферах, як точність фактів, міркування та потенціал

генерування шкідливих порад. Хоча Med-PaLM тісно відповідає науковому консенсусу в понад 92% своїх відповідей, він все ще не відповідає якості прийняття рішень клініцистами.

Ці обмеження підкреслюють необхідність багатоагентної системи LLM у телемедицині, як запропоновано в цій дисертації. Не покладаючись лише на одну модель, а натомість використовуючи розподілений агентний підхід, стає можливим пом'якшити деякі з цих проблем. Кожен агент може спеціалізуватися на різних медичних завданнях — діагностиці, рекомендаціях щодо лікування, аналізі історії пацієнтів і виявленні аномалій — таким чином пропонуючи більш надійне та надійне рішення для надання медичної допомоги першого рівня.

2.2. Агентні підходи вирішення проблеми з використанням загальних Великих Мовних Моделей

Підхід openCHA [5] використовує великі мовні моделі (LLM) для розмовних агентів у медицині. Він усуває обмеження існуючих систем, забезпечуючи більш персоналізовані відповіді, інтегруючи зовнішні джерела даних і обробляючи мультимодальні дані для складних завдань охорони здоров'я. Фреймворк використовує оркестратор для збору та обробки зовнішньої інформації, покращення отримання знань, багатомовної підтримки та взаємодії з іншими платформами ШІ (рис. 3).

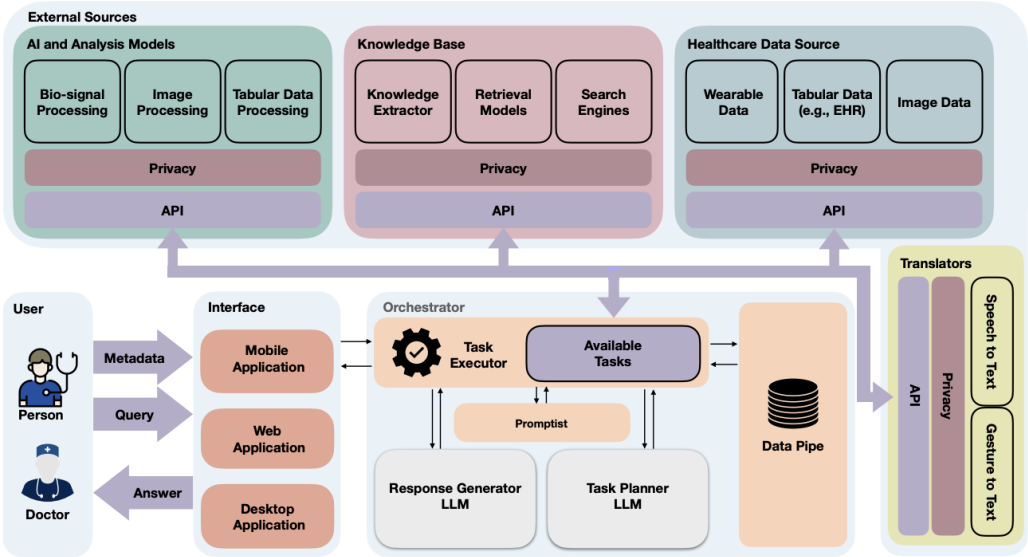


Рисунок 3. Архітектура системи openCHA [5]

На відміну від openCHA, підхід, що розглядається у даній роботи, запропонований у виходить за openCHA, пропонуючи глибше розуміння проблем клієнта за допомогою більш комплексного відстеження та аналізу даних. Використовуючи кілька спеціалізованих агентів, він охоплює ширший контекст, інтегрує ширший діапазон вхідних даних і забезпечує більш індивідуальну відповідь, що в кінцевому підсумку призводить до більш надійного медичного рішення.

MedAgents [6], яка використовує багатоагентний підхід на основі LLM для вирішення проблем у галузі медицини. MedAgents вирішує такі проблеми, як спеціальна термінологія та потреба в спеціальних знаннях, використовуючи рольову структуру, де LLM-агенти беруть участь у багатораундовій спільній дискусії. Процес передбачає збір думок від експертів у галузі, пропонування індивідуальних аналізів, їх узагальнення та повторення обговорень до досягнення консенсусу. Платформа не потребує навчання та наголошує на нульовому налаштуванні, демонструючи високу продуктивність у різних наборах медичних даних QA (рис. 4).

У той час як MedAgents демонструє покращення аргументації LLM через спільне обговорення, багатоагентна система у даній роботі пропонує кращий підхід, забезпечуючи глибше розуміння проблем, пов'язаних із клієнтом. Бо MedAgents не виділяють достатньо уваги насиченням контекстом LLM

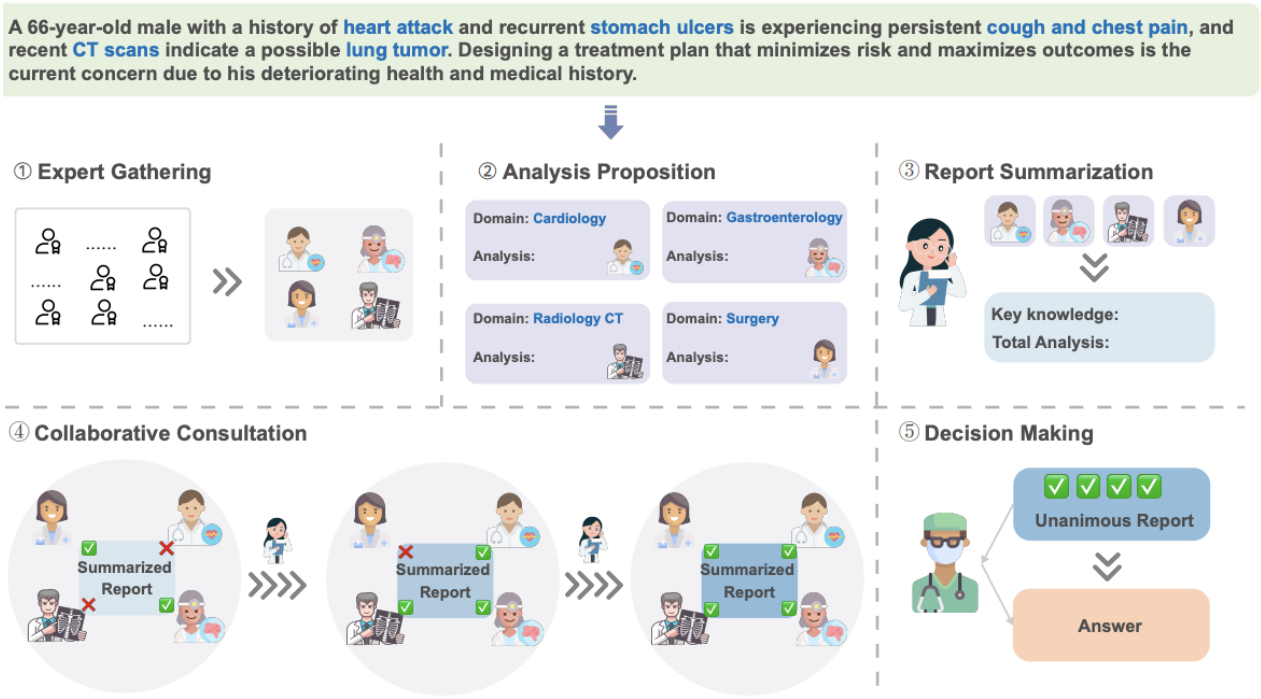


Рисунок 4. Архітектура MedAgents [6]

3. РОЗРОБКА ВЛАСНОЇ СИСТЕМИ ДЛЯ ПЕРШОЇ ЛАНКИ ДОПОМОГИ

3.1. Загальна архітектура

Враховуючи недоліки використання підходів Med-PaLM2 [3], openCHA [5] та MedAgents [6], було створено систему, яка об'єднує головні переваги кожного з методів, що дозволяє підвищити якість перфомансу.

Архітектура системи складається з наступних компонентів (рис. 5):

- RDBMS з EHR: зберігання даних пацієнтів для насичення контесту ширшою інформацією
- Med General info та PubMed: набір медичних даних для ширшого аналізу спеціалізованих тем
- MedQA(та інші схожі датасети): будь-які датасет з питанням - відповіддю даними.
- Search by symptoms: сервіс пошуку необхідної інформації
- 3rd party anomaly service: сервіс для ідентифікації аномалій таких як рентген знімки, серцебиття, та інші.
- Chat UI: спосіб використання системи за допомогою зручного інтерфейсу
- Orchestration Service & MultiAgent Service: сервіси для мультиагентної обробки запитів з певним управлінням для крайніх випадків.

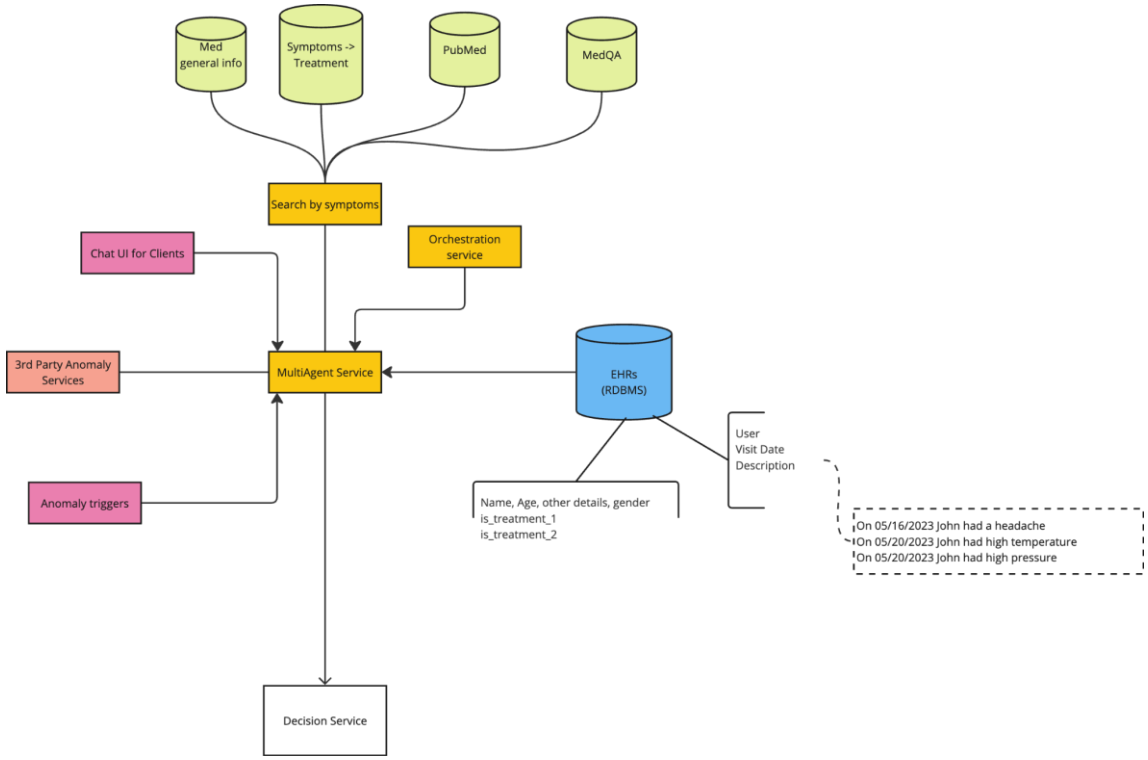


Рисунок 5. Загальна архітектура системи.

3.2. Ланцюг виконання та оркестрації агентів

Система функціонує за рахунок реалізації багатьох паралельних процесів, які взаємодіють між собою для забезпечення точності та оперативності. Нижче наведено опис основних процесів, що складають основу роботи системи:

1. Процес збору даних – відповідає за збирання всієї доступної інформації про стан пацієнта, включаючи історію хвороби, результати аналізів, попередні діагнози та скарги, а також будь-які аномальні симптоми, зафіксовані пацієнтом. Ці дані зберігаються у централізованій Базі Даних у форматі електронних медичних записів (EHR), що дає змогу швидко отримати доступ до інформації під час подальшої обробки.
2. Процес моніторингу аномалій – включає в себе постійне відстеження даних, які надходять з IoT девайсів, наприклад, розумних годинників, фітнес-трекерів, медичних сенсорів. Цей процес фокусується на виявленні критичних змін у показниках здоров'я, таких як пульс, рівень кисню в крові, температура тіла, і автоматично сигналізує про аномалії, якщо виявлено небезпечні відхилення від норми.
3. Процес роботи агенту-координатора – це ключовий компонент системи, що відповідає за ухвалення рішень. Агент оцінює зібрані дані та вирішує подальші кроки: чи необхідно розпочати пошук можливих діагнозів, попередити пацієнта про потенційні загрози здоров'ю, чи викликати сервіс для більш детального аналізу виявлених аномалій. Агент використовує машинне навчання та штучний інтелект для автоматизації цих процесів, що дає змогу зменшити навантаження на медичний персонал.
4. Пошуковий двигун – відповідає за здійснення багаторівневого пошуку всередині Бази Даних для швидкого знаходження релевантної інформації щодо діагностики та лікування. Він використовує різні алгоритми пошуку та індексації, щоб забезпечити високу швидкість та точність відповіді, навіть при великому обсязі даних. Двигун

здатен знаходити схожі випадки захворювань, аналізувати попередні методи лікування та пропонувати найкращі варіанти на основі статистики успіху.

5. Процес відбору релевантної інформації — після виконання пошуку, система відбирає найбільш релевантні дані та формує звіт, що містить ключову інформацію для подальшого ухвалення рішення лікарем або передавання її у відповідний сервіс. Цей процес оптимізований таким чином, щоб уникнути перевантаження даними, залишаючи тільки ті відомості, які мають безпосереднє значення для поточного стану пацієнта.
6. Завдяки такій архітектурі система забезпечує ефективне управління даними, швидкий аналіз та своєчасне прийняття рішень, що дозволяє знизити ризики для пацієнтів та підвищити якість медичної допомоги.

4. ВИСНОВКИ

Запропонована у даній роботі мультиагентна система на основі великих мовних моделей (LLM) демонструє суттєві переваги порівняно з існуючими підходами, такими як Med-PaLM2, openCHA та MedAgents. Незважаючи на досягнення цих рішень, їх основні недоліки полягають у недостатній точності, відсутності комплексного розуміння контексту та обмеженій гнучкості. Запропоноване рішення вирішує ці проблеми за допомогою розподіленого агентного підходу, який інтегрує кілька спеціалізованих агентів, кожен з яких виконує окрему роль: від діагностики та аналізу історії пацієнтів до ідентифікації аномалій за допомогою IoT девайсів. На відміну від Med-PaLM2, яка покладається на одну велику модель, нова архітектура значно знижує ризик галюцинацій завдяки оркестрації агентів, що забезпечує більш глибоке насичення контекстом та надійні механізми перевірки даних. Порівняно з openCHA, система виходить за рамки простої інтеграції даних, застосовуючи багаторівневий пошук та індивідуальну обробку запитів, що дозволяє значно покращити якість прийнятих рішень. Щодо MedAgents, розроблена система забезпечує більш глибоке розуміння проблем користувача та більш надійне виконання завдань завдяки комплексній обробці контексту та більшому охопленню медичних сценаріїв. Таким чином, запропонована мультиагентна система не тільки підвищує якість і точність надання первинної медичної допомоги, але й забезпечує більшу гнучкість та можливості для масштабування. Це робить її потужним інструментом для подолання обмежень традиційних підходів і виводить телемедицину на новий рівень, покращуючи доступність, надійність та ефективність медичних послуг.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Improving language understanding with unsupervised learning URL: https://openai.com/index/language-unsupervised/ (дата звернення: 11.11.2024).
2. Singhal, K., Azizi, S., Tu, T. et al. Large language models encode clinical knowledge. Nature 620, 172–180 (2023). URL: https://doi.org/10.1038/s41586-023-06291-2 (дата звернення: 11.11.2024).
3. Pathways Language Model (PaLM): Scaling to 540 Billion Parameters for Breakthrough Performance URL: https://research.google/blog/pathways-language-model-palm-scaling-to-540-billion-parameters-for-breakthrough-performance/ (дата звернення: 11.11.2024).
4. What Disease does this Patient Have? A Large-scale Open Domain Question Answering Dataset from Medical Exams URL: https://arxiv.org/abs/2009.13081 (дата звернення: 11.11.2024).
5. Conversational Health Agents: A Personalized LLM-Powered Agent Framework URL: https://arxiv.org/abs/2310.02374 (дата звернення: 11.11.2024).
6. MedAgents: Large Language Models as Collaborators for Zero-shot Medical Reasoning URL: https://arxiv.org/abs/2311.10537 (дата звернення: 11.11.2024)

АВТОМАТИЗАЦІЯ РОЗПІЗНАВАННЯ ТОЧОК ВІДМОВИ РОЗПОДІЛЕНОЇ СИСТЕМИ НА ОСНОВІ АНАЛІЗУ ТРАСУВАННЯ ЗАПИТІВ, ЛОГІВ ТА МЕТРИК

Хитрий М.С.[1], Письменний І.О.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] khytryi.mykoly@lll.kpi.ua , [2] ihor.pismennyy@gmail.com [0000-0001-7648-2593]

Зі зростанням складності розподілених комп'ютерних систем та обсягів даних, що оброблюються, особливої актуальності набуло питання автоматизації виявлення та аналізу точок відмови. Сучасні системи потребують рішень, які можуть швидко і точно діагностувати причини збоїв, обробляючи великі обсяги логів, метрик та трасувань. Дана робота присвячена проблемі автоматизації розпізнавання точок відмови в розподілених системах за допомогою великих мовних моделей (англ. Large Language Models, LLM), що дозволяють інтегрувати аналіз мультимодальних даних для підвищення надійності та швидкості реагування на інциденти.

Ключові слова: RCA, розподілені системи, LLM, метрики.

1. ВСТУП

У сучасному світі автоматизація системного моніторингу є критично важливим аспектом забезпечення надійності розподілених комп'ютерних систем, де різні компоненти взаємодіють між собою, обробляючи численні запити в режимі реального часу. Така архітектура ускладнює виявлення потенційних точок відмови, які можуть призвести до деградації продуктивності або повного збою системи. Традиційні методи моніторингу, такі як фіксовані порогові значення метрик або шаблони в логах, виявилися недостатньо ефективними для сучасних складних систем, оскільки вони не враховують динамічні взаємозв'язки між компонентами та різноманітні сценарії збоїв [1].

У відповідь на ці виклики все частіше використовуються підходи на основі машинного навчання, зокрема глибинного навчання, які дозволяють виявляти аномалії в поведінці системи на основі мультимодальних даних — логів, метрик та трасування запитів. Логи надають інформацію про події, метрики відображають продуктивність, а трасування запитів дозволяє відстежувати їхній шлях між компонентами системи. Інтеграція цих джерел даних створює більш точну модель, здатну не лише виявляти аномалії, а й локалізувати їхнє джерело.

Зі зростанням складності розподілених систем виявлення та діагностика точок відмови стає все більш тривалим і трудомістким процесом, що підвищує навантаження на інженерів, які відповідають за реагування на інциденти. Використання великих мовних моделей (LLM) та LLM-агентів відкриває нові можливості для автоматизації цього процесу, дозволяючи інтегрувати аналіз трасувань, логів та метрик в єдину систему. Це не лише прискорює процес виявлення точок відмови, але й дозволяє інженерам взаємодіяти з системою природною мовою, спрощуючи процес усунення проблем.

Отже, актуальною задачею є розробка інтегрованої системи автоматизації розпізнавання точок відмови з використанням LLM та LLM-агентів для аналізу трасування запитів, логів та

метрик. Така система підвищить ефективність реагування на інциденти, скоротить час простою і покращить загальну надійність розподілених систем [2].

2. ЗАПРОПОНОВАНЕ РІШЕННЯ НА БАЗІ LLM

В якості рішення пропонується запровадити окремий компонент у систему, в яку вже інтегровані інструменти для моніторингу. Його відповідальністю є обробка поточного інциденту, метрик, логів та трейсів, що буде зібрані в певному його часовому вікні.

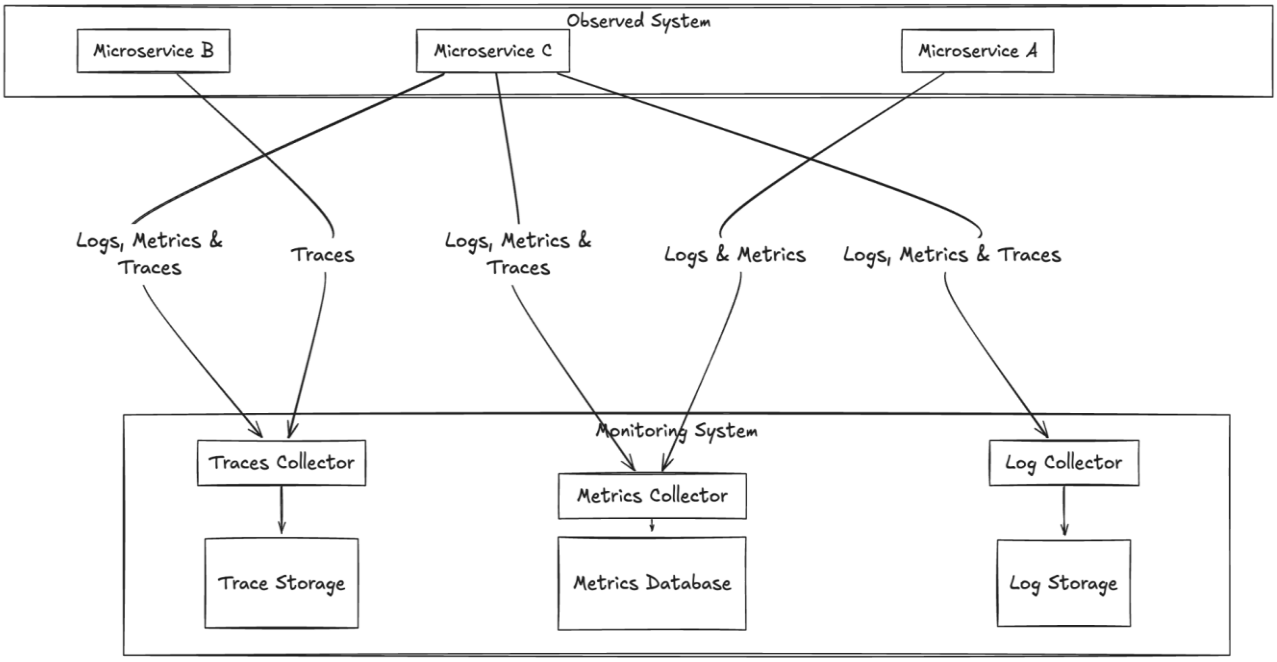


Рисунок 1. Схема системи, яка спостерігається

На рисунку 1 показано систему моніторингу, яка збирає дані від різних мікросервісів в межах розподіленої системи. Кожен мікросервіс генерує логи, метрики та трасування, які потім передаються до модулів збору у системі моніторингу. Система моніторингу включає колектори для трасування, метрик та логів, які отримують ці дані від мікросервісів та зберігають їх у відповідних сховищах. Зібрані дані будуть використовуватись для подальшого аналізу LLM-Agent в рамках процесу **RCA (Root Cause Analysis)** для виявлення точок відмови та визначення причин інцидентів.

На рис. 2 зображено процес роботи системи RCA, яка призначена для виявлення причин інцидентів у розподіленій системі. Спочатку відбувається збір даних, які обробляються агрегатором, і далі передаються в агент LLM для аналізу. Цей агент отримує доступ до зібраних даних та модулю аналізу інцидентів для подальшого дослідження.

На наступному етапі відбувається узагальнення інформації про інциденти. Далі використовується велика мовна модель (LLM), яка створює короткий звіт. Цей звіт допомагає сформувати векторне представлення інциденту, яке зберігається в базі даних векторів інцидентів.

Після генерації векторів відбувається пошук схожих інцидентів, використовуючи алгоритм пошуку найближчих сусідів. Це дозволяє знайти схожі випадки у базі даних, що може допомогти виявити загальні причини або патерни. На основі знайденої інформації система передає дані в LLM для додаткового RCA-аналізу, що призводить до прогнозу кореневої причини інциденту. Як фінальний крок, система формує пояснення і пропонує рекомендовані дії для вирішення проблеми або запобігання її повторенню в майбутньому.

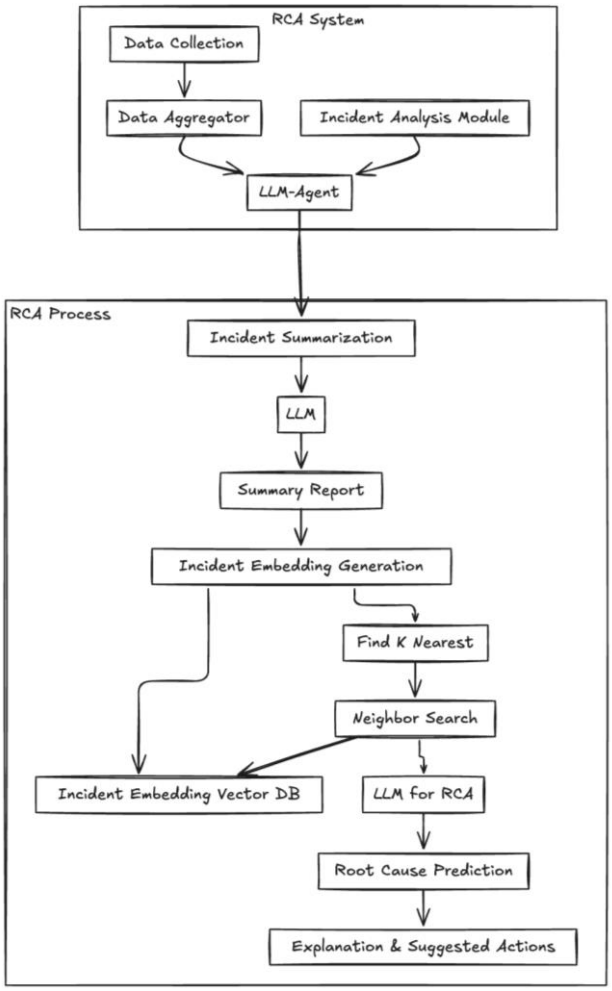


Рисунок 2. Компонент для виявлення точки відмови з допомогою LLM

3. ВИСНОВКИ

Запропоноване рішення демонструє, що великі мовні моделі (LLM) можуть ефективно використовуватися для автоматизації розпізнавання точок відмови в розподілених системах. Застосування LLM у цій сфері дозволяє обробляти великі обсяги мультимодальних даних (трасування запитів, логи, метрики), що дає змогу швидше і точніше виявляти аномалії та визначати їхні кореневі причини. Інтеграція LLM з методами машинного навчання для кластеризації та пошуку схожих інцидентів допомагає значно знизити навантаження на інженерів-чергових та мінімізувати людський фактор. Це рішення забезпечує контекстуальне розуміння складних збоїв, прискорює реакцію на інциденти та підвищує загальну надійність системи. Дослідження підтверджують, що використання LLM для автоматизації RCA є перспективним і може стати важливим компонентом майбутніх систем моніторингу й діагностики [3].

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Automatic Root Cause Analysis via Large Language Models for Cloud Incidents URL: https://arxiv.org/pdf/2305.15778

2. Empowering Practical Root Cause Analysis by Large Language Models for Cloud Incidents URL: https://yinfangchen.github.io/assets/pdf/llm_rca.pdf

3. Exploring LLM-based Agents for Root Cause Analysis URL: https://arxiv.org/pdf/2403.04123

БЕЗПЕКА БЛОКЧЕЙН СИСТЕМ В МІКРОСЕРВІСНІЙ АРХІТЕКТУРІ

Хуторянський Д.О.[1], Булах Б.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] khutorianskyi.daniil.@lll.kpi.ua , [2] bogdan.bulakh@gmail.com [0000-0001-5880-6101]

Сучасні інформаційні системи все частіше базуються на розподілених архітектурах, що поєднують переваги різних технологій. Блокчейн, завдяки своєму потенціалу в забезпеченні незмінності та прозорості даних, активно інтегрується в різноманітні застосунки, включаючи ті, що розроблені з використанням мікросервісної архітектури. Таке поєднання дозволяє створювати гнучкі та масштабовані системи, здатні ефективно обробляти великі обсяги даних. Особливої актуальності набуває питання безпеки, оскільки компрометація даних чи збій в роботі може мати серйозні наслідки. Дана робота присвячена дослідженню проблем безпеки блокчейн систем, інтегрованих у мікросервісну архітектуру, та розробці ефективних методів їх вирішення. В статті буде проведено аналіз специфічних загроз та вразливостей, характерних для таких систем, з урахуванням розподіленої природи мікросервісів та особливостей функціонування блокчейну. Результати дослідження спрямовані на підвищення надійності та безпеки розподілених систем на основі блокчейну та мікросервісів.

Ключові слова: блокчейн, мікросервісна архітектура, безпека.

1. ВСТУП

Інтеграція блокчейн технологій у мікросервісну архітектуру є одним із актуальних напрямків розвитку сучасних розподілених систем. Мікросервіси, завдяки своїй модульності та гнучкості, дозволяють створювати складні застосунки з високим рівнем масштабованості та відмовостійкості. Блокчейн, в свою чергу, забезпечує незмінність, прозорість та безпеку даних, що робить його привабливим для різноманітних застосувань, де важлива довіра та аудит інформації.

Однак, поєднання цих двох технологій породжує нові виклики в галузі безпеки. Розподілена природа мікросервісів створює множинні точки входу для потенційних атак, що ускладнює їх захист. Комунікація між мікросервісами та взаємодія з блокчейном потребують надійних механізмів захисту від несанкціонованого доступу, підміни даних та інших видів атак. Крім того, необхідно враховувати специфічні загрози, пов'язані з безпекою смарт-контрактів, управлінням ключами та доступом до даних в розподіленому реєстрі.

Дана робота присвячена комплексному дослідженню проблем безпеки блокчейн систем в мікросервісній архітектурі. В роботі будуть проаналізовані основні загрози та вразливості, характерні для таких систем, з урахуванням специфіки їх архітектури та функціонування. Особлива увага буде приділена розробці та аналізу ефективних методів захисту, що дозволяють підвищити безпеку та надійність розподілених застосунків на основі блокчейну та

мікросервісів. Результати дослідження можуть бути використані для створення безпечних і надійних систем в різних галузях, де застосовуються блокчейн технології.

2. ПРОБЛЕМИ БЕЗПЕКИ БЛОКЧЕЙН СИСТЕМ В МІКРОСЕРВІСНІЙ АРХІТЕКТУРІ

Інтеграція блокчейну та мікросервісів, хоч і обіцяє значні переваги, водночас породжує унікальні виклики в галузі безпеки. Окрім загроз, характерних для мікросервісної архітектури, таких як розподіленість та необхідність захисту комунікацій між сервісами, додаються ще й ризики, пов'язані з використанням блокчейн технології.

Однією з ключових проблем є забезпечення стійкості механізму консенсусу, який лежить в основі блокчейну. Різні алгоритми консенсусу (Proof-of-Work, Proof-of-Stake тощо) мають свої вразливості до атак, таких як атака 51%, сивільні атаки або атаки на доступність мережі [1]. Зловмисники можуть спробувати маніпулювати процесом досягнення консенсусу, щоб підробити транзакції, подвоїти витрати або порушити роботу мережі.

Безпека смарт-контрактів також є критично важливою. Помилки в коді смарт-контрактів або недостатній рівень аудиту можуть призвести до експлуатації вразливостей зловмисниками. Це може спричинити викрадення коштів, маніпуляції з даними або навіть повну зупинку роботи системи [2].

Ще одним важливим аспектом є захист комунікацій між мікросервісами та блокчейн системою. Використання незахищених протоколів або недостатньо надійних механізмів аутентифікації та авторизації може відкрити шлях для несанкціонованого доступу, підміни даних чи інших видів атак. Це особливо актуально при взаємодії з блокчейном, де будь-яка компрометація даних може мати серйозні наслідки, враховуючи незмінність записів в розподіленому реєстрі.

Необхідно також забезпечити надійне управління ключами та контроль доступу до даних в розподіленому реєстрі. Втрата або компрометація приватних ключів може призвести до незворотної втрати коштів чи даних. Крім того, важливо запобігти несанкціонованому доступу до конфіденційних даних, що зберігаються в блокчейні, забезпечуючи відповідні рівні авторизації та контролю доступу.

Враховуючи ці проблеми, забезпечення безпеки блокчейн систем в мікросервісній архітектурі вимагає комплексного підходу, що враховує специфіку обох технологій та їх взаємодію.

3. СТРАТЕГІЇ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ В БЛОКЧЕЙН-МІКРОСЕРВІСНИХ СИСТЕМАХ

Забезпечення безпеки блокчейн систем, інтегрованих у мікросервісну архітектуру, вимагає комплексного підходу, який враховує специфіку обох технологій. Необхідно застосовувати різноманітні стратегії та механізми захисту, спрямовані на протидію різним типам загроз, описаним у попередньому розділі.

Для забезпечення стійкості блокчейн мережі до атак необхідно ретельно обирати та налаштовувати алгоритм консенсусу. Наприклад, для систем з високими вимогами до безпеки може бути доцільним використання Proof-of-Work з достатньою обчислювальною потужністю мережі, щоб запобігти атакам 51%. У випадку Proof-of-Stake важливо забезпечити децентралізацію стейкінгу та запобігти концентрації монет в руках невеликої кількості учасників. Додаткові заходи безпеки можуть включати використання пулів стейкінгу, делегованого Proof-of-Stake та інших механізмів, що підвищують стійкість мережі до маніпуляцій.

Для мінімізації ризиків, пов'язаних з використанням смарт-контрактів, необхідно застосовувати комплексний підхід до їх розробки та впровадження. Це включає ретельний аудит коду, використання формальних методів верифікації, а також застосування best practices для написання безпечного коду. Важливо також враховувати потенційні вразливості, такі як реентерабельність, переповнення буфера та інші, та застосовувати відповідні заходи захисту. Наприклад, для запобігання атакам реентерабельності можна використовувати патерн "Checks-Effects-Interactions", а для захисту від переповнення буфера слід ретельно перевіряти вхідні дані та використовувати безпечні функції для роботи з пам'яттю [2].

Для забезпечення безпеки комунікацій між мікросервісами та блокчейн системою необхідно використовувати захищені протоколи, такі як TLS/SSL, для шифрування трафіку та забезпечення конфіденційності даних. Важливо також застосовувати надійні механізми аутентифікації та авторизації, щоб запобігти несанкціонованому доступу до системи. Це може включати використання токенів доступу, цифрових підписів та інших криптографічних методів. Крім того, слід забезпечити цілісність даних, що передаються між мікросервісами та блокчейном, щоб запобігти їх підміні або модифікації.

Надійне управління ключами є критично важливим для забезпечення безпеки блокчейн систем. Це включає використання безпечних методів генерації, зберігання та використання ключів. Наприклад, для зберігання приватних ключів можна використовувати апаратні гаманці або захищені хмарні сервіси. Важливо також застосовувати мультипідписи для підвищення безпеки транзакцій та запобігання несанкціонованому доступу до коштів. Крім того, необхідно ретельно контролювати доступ до даних в блокчейні, використовуючи рольовий контроль доступу та інші механізми авторизації.

Для забезпечення безпеки мікросервісів необхідно застосовувати різноманітні заходи захисту, такі як ізоляція мікросервісів, моніторинг та логірування, захист API та інші. Ізоляція мікросервісів дозволяє обмежити вплив потенційних атак на окремі сервіси, запобігаючи поширенню загрози на всю систему. Моніторинг та логірування дозволяють відстежувати активність мікросервісів та вчасно виявляти аномалії та потенційні загрози. Захист API мікросервісів забезпечується застосуванням механізмів аутентифікації, авторизації та валідації даних [3].

В цілому, забезпечення безпеки блокчейн-мікросервісних систем є складним завданням, яке вимагає комплексного підходу та застосування різноманітних стратегій та механізмів захисту. Важливо враховувати специфіку обох технологій та їх взаємодію, а також постійно моніторити та аналізувати нові загрози та вразливості.

4. ВИСНОВКИ

Дана робота досліджувала актуальну проблему забезпечення безпеки блокчейн систем в мікросервісній архітектурі. Було проаналізовано специфічні загрози та вразливості, пов'язані з розподіленою природою мікросервісів та особливостями блокчейну. Виявлено ключові виклики: захист механізму консенсусу, безпека смарт-контрактів, захист комунікацій та управління ключами. Запропоновано комплексний підхід до забезпечення безпеки, що включає ретельний вибір алгоритму консенсусу, аудит та верифікацію смарт-контрактів, використання захищених протоколів комунікації та надійне управління ключами.

Внеском роботи є детальний аналіз вразливостей та розробка рекомендацій щодо їх усунення. Запропоновані стратегії можуть бути використані для створення надійних розподілених систем. Результати мають практичне значення для розробників, дозволяючи їм враховувати потенційні ризики та застосовувати ефективні заходи захисту, що сприятиме підвищенню довіри до блокчейн технологій. Подальші дослідження можуть бути спрямовані

на вивчення нових загроз та розробку інноваційних методів захисту, а також на розробку стандартів безпеки для блокчейн-мікросервісних систем.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. A Survey on the Security of Blockchain Systems, URL: https://www.researchgate.net/publication/319249505_A_Survey_on_the_Security_of_Blockchain_Systems

2. A Microservice-enabled Architecture for Smart Surveillance using Blockchain Technology, URL: https://ieeexplore.ieee.org/abstract/document/8656968

3. BlendMAS: A Blockchain-Enabled Decentralized Microservices Architecture for Smart Public Safety, URL: https://ieeexplore.ieee.org/abstract/document/8946177

ВИКОНАННЯ РОЗПОДІЛЕНИХ ОБЧИСЛЕНЬ З ВІРТУАЛІЗАЦІЄЮ СЕРЕДОВИЩА КОРИСТУВАЧА

Шаблій В.С.[1], Булах Б.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] shabliy.volodymyr@lll.kpi.ua [0009-0004-5113-3572],
[2] bogdan.bulakh@gmail.com [0000-0001-5880-6101]

Метою даного дослідження є огляд та розробка методу віддалених розподілених обчислень який потребує мінімальних налаштувань з боку користувача та не потребує переписування існуючих застосунків під розподілене виконання. В результаті отримано рішення що може прискорити виконання існуючих розподілених програм що використовують процеси для обчислювально важких задач без їх адаптації під віддалене виконання.

Ключові слова: розподілені обчислення, перехоплення системних викликів, віртуалізація файлової системи

1. ВСТУП

У сучасному світі інформаційних технологій обчислювальні ресурси відіграють ключову роль у розвитку науки, бізнесу та промисловості. Зі зростанням обсягів даних та складності завдань виникає потреба в ефективному використанні віддалених ресурсів для прискорення різноманітних процесів, таких як збірка коду, моделювання та проведення складних розрахунків. Розподілені обчислення стали невід'ємною частиною багатьох систем, дозволяючи розподіляти навантаження між декількома машинами та оптимізувати використання ресурсів.

Однак впровадження розподілених обчислень часто супроводжується значними складнощами. Це може вимагати спеціальних налаштувань, адаптації програмного забезпечення під специфічні системи або навіть повного переписування коду. Такі підходи не лише затратні за часом, але й підвищують ризик виникнення помилок та несумісностей.

Постає питання: як, без внесення змін до існуючих програм, які вже здатні виконуватися розподілено на локальній машині за допомогою процесів, реалізувати їх виконання на декількох віддалених машинах з мінімальними конфігураційними зусиллями? Вирішення цієї проблеми відкриє можливості для більш ефективного використання розподілених систем без необхідності глибокого втручання в код або налаштування середовища.

У даній роботі розглянуто методи та технології, які дозволяють виконувати розподілені обчислення з використанням віртуалізації середовища користувача. Розроблено можливий підхід, який забезпечує зручне виконання програм на віддалених машинах, мінімізуючи необхідність додаткових налаштувань та змін у програмному забезпеченні.

2. ОГЛЯД МОЖЛИВИХ МЕТОДІВ

2.1. Перехоплення системних викликів

Уявімо задачу обробки великої кількості мультимедійних файлів, а саме відеофайлів, в операційній системі Linux. Для цього користувач може використати скрипт що для кожного файлу з директорії запускає новий процес з застосунком(накшталт ffmpeg [1]) що читає цей файл, оброблює його та записує в нову директорію.

В операційній системі лінукс створення нового процесу відбувається за допомогою двох системних викликів: fork(або vfork/clone) за яким слідує системний виклик execve. Fork дублює процес який його викликав повертаючи різні результати для двох процесів після чого один з них викликає execve що замінює пам’ять процесу на нову программу. Таке розподілення існує ще з часів Unix та з’явилось через те, що створення об’єднаного системного виклику fork-exec викликало зайву складність [2] системи.

Маючи можливість перехопити та замінити створення нового процесу на процес що буде тільки повертати результати можна відправити справжній процес на віддалене виконання.

Ptrace

Ptrace (від англ. process trace) – це системний виклик у Unix-подібних операційних системах, який надає можливість одному процесу (контролеру) спостерігати та змінювати виконання іншого процесу (цілі). Цей механізм широко використовується у відлагоджувачах, таких як gdb, для встановлення точок зупинки, відстеження змін регістрів та пам'яті, а також для перехоплення та модифікації системних викликів [3] (рис. 1).

Ідея з перехопленням та зміною системних викликів не є новою, наприклад ptrace можна застосувати для розподілення роботи з файловою системою [4, 5].

LD_PRELOAD

LD_PRELOAD – це змінна середовища, яка інструктує динамічний завантажувач (ld.so або ld-linux.so) завантажити вказані бібліотеки перед іншими під час запуску програми. Якщо в завантаженій бібліотеці є функції з такими ж іменами, як у стандартних бібліотеках, то програма буде використовувати саме їх. Таким чином якщо завантажити функції стандартної бібліотеки що відповідають за системні виклики то можна перехопити їх виконання.

Однак використання LD_PRELOAD має обмеження. Не всі програми можуть бути перехоплені таким чином. Програми, що статично зв'язані з бібліотеками, або ті, що використовують власні механізми завантаження динамічних бібліотек, не підтримують LD_PRELOAD. Це означає, що функції, які завантажуються специфічним способом, можуть уникнути перехоплення.

Зміна таблиці системних викликів

Ядро операційної системи Linux містить таблицю системних викликів, яка є масивом вказівників на функції-обробники системних викликів. Кожен запис у цій таблиці відповідає певному системному виклику, і коли програма викликає системний виклик, ядро звертається до відповідного обробника за індексом у таблиці [6].

Перехоплення системних викликів може бути здійснено шляхом модифікації цієї таблиці. Замінюючи вказівник на оригінальний обробник своїм власним, можна змінити поведінку системного виклику для всіх процесів у системі.

Проте модифікація ядра є складним та потенційно небезпечним процесом. Необережні дії можуть призвести до нестабільності системи, витоків пам'яті або навіть критичних помилок ядра. Розробка модулів ядра вимагає глибоких знань внутрішньої структури операційної системи та її механізмів.

2.2. Засоби для віртуалізації середовища користувача

Одного лиш перехоплення та відправлення системних викликів створення нових процесів на віддалену машину недостатньо для успішного виконання програми оскільки програма очікує такого ж середовища як і на локальній машині. Частіше всього середовище представляє собою набір файлів та бібліотек.

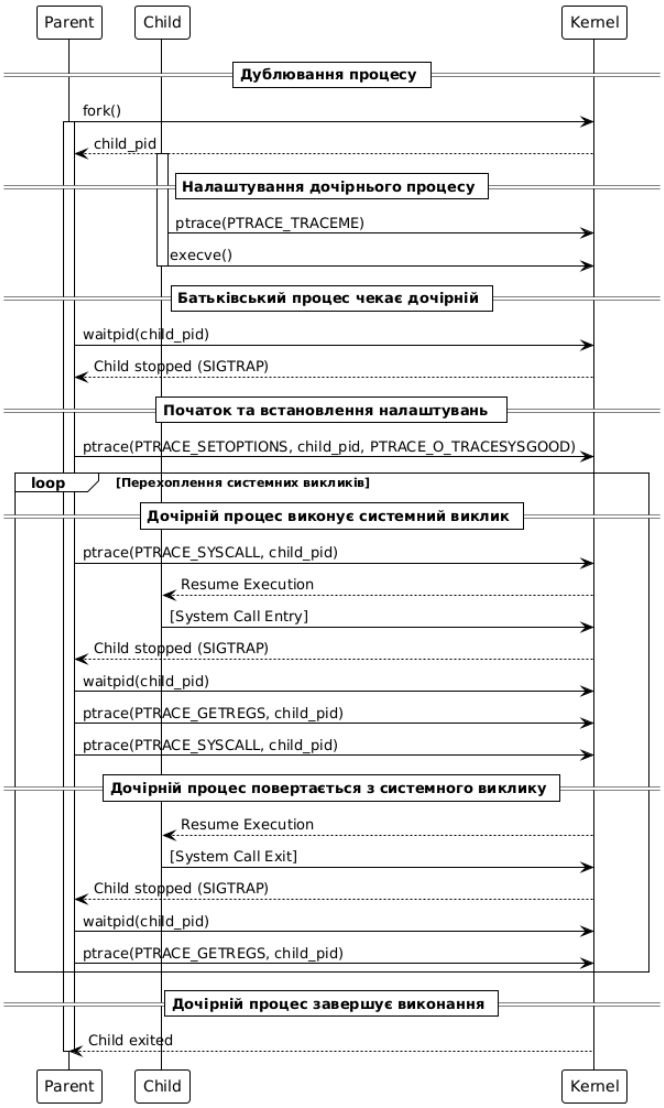


Рисунок 1. Діаграма послідовності відслідковування системних викликів за допомогою ptrace

FUSE

FUSE (Filesystem in User Space) є інтерфейсом, який дозволяє розробникам реалізовувати власні файлові системи в просторі користувача. Це досягається шляхом взаємодії між модулем ядра FUSE та користувацьким процесом, який реалізує логіку файлової системи. Коли програма виконує операції з файлами, такі як читання або запис, ці запити перенаправляються модулем FUSE до відповідного користувацького процесу для обробки.

Така архітектура дозволяє створювати віртуальні файлові системи, які можуть взаємодіяти з різними джерелами даних: віддаленими серверами, базами даних, іншими файловими системами тощо. Програми, які працюють з такою файловою системою, не потребують жодних змін і сприймають її як стандартну частину операційної системи.

Приклад успішного використання FUSE для взаємодії з віддаленою файловою системою можна побачити у [7].

Ptrace

Для того щоб забезпечити локальне середовище на віддаленій машині можна знову перехоплювати системні виклики вже віддаленого процесу проте цього разу нас цікавлять

системні виклики що працюють з файловою системою на кшталт openat, access, newfstatat та у разі доступу до файлової системи завантажувати необхідні файли та перенаправляти системні виклики до них

3. АРХІТЕКТУРА СИСТЕМИ

Розробка системи відбувалась за допомогою мови програмування Zig. Zig має багато засобів роботи з мовою програмування C що необхідно нам для виконання системних викликів таких як ptrace напряму через бібліотеки операційної системи. Також Zig робить складним ігнорування обробки помилок що в кінцевому результаті пришвидшує розробку завдяки зменшенню часу проведеного за відладкою застосунку. Строга типізація дозволяє перенести більшість помилок виконання на помилки компіляції [8].

Система поділена на два компоненти: перехоплювач та виконавець. Перехоплювач запускає дочірній процес та починає перехоплення системних викликів за допомогою Ptrace. Перехоплювач спостерігає за системними викликами fork та execve.

Спостереження ведеться одразу за декількома процесами: у разі fork до ptrace додається новий процес. Якщо відбувається виклик execve дочірній процес зупиняється перед виконанням виклику.

Отримуються аргументи виклику та якщо процес збирається виконувати программу яку ми збираємось розподіляти вирішується чи варто відправляти програму на віддалене обчислення. У разі якщо місця для локального виконання достатньо перехоплювач нічого не робить та дочірній процес продовжує своє виконання як зазвичай.

Якщо всі місця для локального виконання зайняті то замість початкової програми ми замінюємо шлях до нашої програми яка буде лише перенаправляти результати з віддаленого процесу. Заміна відбувається шляхом зміни регістрів перед виконанням системного виклику.

Оскільки шлях є вказівником на рядок у пам’яті ми маємо знайти місце де має знаходитись наш рядок з шляхом до нашої програми. Перехоплювач не має доступу до пам’яті дочірнього процесу оскільки вони знаходяться у різних адресних просторах.

Системні виклики ptrace мають обмежений доступ до пам’яті дочірнього процесу та можуть зчитувати та змінювати пам’ять по одному машинному слову за виклик. Рядок можна покласти в простір пам’яті стеку зчитавши регістри що вказують на нього [9].

Інший підхід який використаний в побудованій системі є використання спільної пам’яті. Для того щоб батьківський та дочірній процес встановили спільну пам’ять можна скористатись posix api для цього [10]. Проте ми не можемо модифікувати код дочірнього процесу.

Для того щоб встановити спільну пам’ять не змінюючи код дочірньої програми ми можемо скористатись LD_PRELOAD. Динамічні бібліотеки в операційній системі Linux можуть містити в собі функції ініціалізації та деініціалізації. Під час ініціалізації ми встановлюємо область пам’яті як спільну з батьківським процессом. В цю область пам’яті ми можемо покласти рядки з батьківського процесу які будуть доступні з дочірнього процесу.

Після того як ми замінили аргументи до системного виклику оригінальні аргументи відправляються по мережі до іншого компоненту системи: виконавця.

Виконавець очікує завдання. Він також запускає дочірні процеси під ptrace проте цього разу його цікавлять системні виклики що працюють з файловою системою. У разі якщо йде звернення до файлу якого не має в кеші файлів, перш ніж продовжити виконання, файл отримується по мережі. Відсутність файлів також кешується. Зміна та створення нових файлів запам’ятовується. Стандартні потоки вводу та виводу перенаправляються з віддаленого процесу на локальний. По завершенню виконання віддалений процес сигналізує локальному

про завершення. Нові та змінені файли відправляються назад. Таким чином створюється вигляд наче процес виконувався локально.

4. РЕЗУЛЬТАТИ

Роботу системи перевірено на невеликому python застосунку та двох комп'ютерах (таблиця 1) під'єднаних по мережі Ethernet 1000Mb/s. Застосунок обробляє відео файли в директорії за допомогою створення процесів ffmpeg для кожного файлу. Кількість одночасних процесів обмежена. Загалом 57 відео що займають 2,3 GiB.

Таблиця 1. Характеристики використаних комп'ютерів

	PC 1	**PC 2**
Процессор	AMD Ryzen 5 4600H 6 cores 2 threads	AMD Ryzen 7 7800X3D 8 cores 2 threads
Пам'ять	16 GiB DDR4	32 GiB DDR5
Диск	NVME 2200 МБ/с Read 1070 МБ/с Write	NVME 7000 МБ/с Read 6000 МБ/с Write
Час локального виконання	12 хв 50с	5 хв 31с

В результаті час виконання з розподіленням зайняв 4 хв 57с. З цього часу відправлення та отримання файлів з віддаленого комп'ютеру в суммі зайняло 25 с. Такі результати були отримані завдяки тому що задача обробки відеофайлів потребує багато обчислень та не багато роботи з файловою системою.

5. ВИСНОВКИ

У результаті проведеного дослідження було розроблено систему для виконання розподілених обчислень з мінімальними вимогами до налаштувань з боку користувача та без потреби в адаптації існуючих застосунків. Запропоноване рішення дозволяє здійснювати віддалене виконання програм шляхом перехоплення та модифікації системних викликів, віртуалізації середовища за допомогою перехоплення доступу до файлової мережі. Тестування показало скорочення часу виконання задач на двох комп'ютерах, що підтверджує ефективність і практичну цінність розробленого підходу для розподіленої обробки обчислювально-інтенсивних завдань

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Krishna R. V. Convert All Files Inside a Folder using FFmpeg (Batch Convert) [Електронний ресурс] / Rao Vijayanagar Krishna. – 2023. – Режим доступу до ресурсу: https://ottverse.com/convert-all-files-inside-folder-ffmpeg-batch-convert/.

2. Ritchie D. M. The Evolution of the Unix Time-sharing System / Ritchie Dennis, 1996.

3. Eaton P. Intercepting and modifying Linux system calls with ptrace [Електронний ресурс] / Phil Eaton. – 2023. – Режим доступу до ресурсу: https://notes.eatonphil.com/2023-10-01-intercepting-and-modifying-linux-system-calls-with-ptrace.html.

4. Thain D. Parrot: Transparent User-Level Middleware for Data-Intensive Computing / Thain Douglas, 2003.

5. Carrot: A Distributed Interposition Library for Networking and File Systems / Pondoc Christopher, 2024.

6. Intercept System Calls [Електронний ресурс] – Режим доступу до ресурсу: https://github.com/bawejakunal/Intercept-System-Calls.

7. Outrun [Електронний ресурс] – Режим доступу до ресурсу: https://github.com/Overv/outrun.

8. Zig Language Overview [Електронний ресурс]. – 2024. – Режим доступу до ресурсу: https://ziglang.org/learn/overview/.

9. Sánchez-Beato A. Modifying System Call Arguments With ptrace [Електронний ресурс] / Alfonso Sánchez-Beato. – 2017. – Режим доступу до ресурсу: https://www.alfonsobeato.net/c/modifying-system-call-arguments-with-ptrace/.

10. Kalin M. A guide to inter-process communication in Linux [Електронний ресурс] / Marty Kalin // opensource.com – Режим доступу до ресурсу: https://opensource.com/sites/default/files/gated-content/inter-process_communication_in_linux.pdf.

ДОСЛІДЖЕННЯ ВІДКРИТИХ ПЛАТФОРМ СЛУХОПРОТЕЗУВАННЯ ДЛЯ РОЗРОБКИ НАБОРУ ЧИСЛОВИХ АЛГОРИТМІВ, ОПТИМІЗОВАНИХ ДЛЯ ВИКОРИСТАННЯ В ЕНЕРГОЕФЕКТИВНИХ СИСТЕМАХ ЦИФРОВОЇ ОБРОБКИ СИГНАЛІВ

Шляхов Д.С.[1], Кирюша Б.А.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] shlyahov.danylo@lll.kpi.ua, [2] bogdankyrysha@gmail.com [0000-0003-2775-6070]

Метою цього дослідження є аналіз існуючих алгоритмів адаптивних фільтрів та їх реалізацій на відкритих платформах для обробки звуку, зокрема у сфері слухопротезування, а також визначення можливих шляхів оптимізації цих алгоритмів для покращення якості та швидкості обробки аудіосигналів. Це дослідження спрямоване на виявлення способів підвищення ефективності адаптивних фільтрів, таких як NLMS, шляхом удосконалення параметрів збіжності та стабільності, що може сприяти розробці більш точних і адаптивних слухових апаратів, здатних краще пристосовуватися до індивідуальних потреб користувача.

Ключові слова: адаптивні фільтри, цифрова обробка сигналів, слухопротезування, енергоефективні системи.

1. ВСТУП

Забезпечення якісної цифрової обробки звуку є важливою складовою в розробці сучасних слухових апаратів, особливо з урахуванням зростаючої потреби в мобільності та енергоефективності пристроїв. Слухові апарати потребують алгоритмів, здатних ефективно адаптуватися до змінних акустичних умов, таких як фоновий шум або різні просторові положення джерел звуку. Для цього необхідні цифрові фільтри, що можуть у реальному часі обробляти сигнал, зберігаючи його якість та стабільність. У зв'язку з цим відкриті платформи, як openMHA, стають важливим інструментом для наукових і прикладних досліджень, оскільки дозволяють розробникам експериментувати з різними фільтрами та алгоритмами без необхідності створювати власні програмні інструменти з нуля. Вони також сприяють швидкому впровадженню нових технологій, оскільки є доступними для широкого кола розробників і дослідників.

Важливою складовою цифрових слухових апаратів є адаптивні фільтри, такі як LMS (Least Mean Squares), NLMS (Normalized Least Mean Squares) та RLS (Recursive Least Squares). Ці фільтри здатні підлаштовуватися під змінні умови та забезпечувати високий рівень обробки звукових сигналів. Адаптивні фільтри дозволяють зменшувати фоновий шум, покращувати розбірливість мови та забезпечувати високу якість звуку. Однак, недоліком багатьох адаптивних фільтрів є їх обчислювальна складність, що призводить до підвищеного енергоспоживання, яке не завжди прийнятне для мобільних пристроїв.

Це дослідження спрямоване на аналіз адаптивних фільтрів, доступних у відкритих платформах, таких як openMHA, з метою оптимізації їх для використання в енергоефективних системах цифрової обробки сигналів.

2. ДОСЛІДЖЕННЯ ТА АНАЛІЗ АДАПТАТИВНИХ ФІЛЬТРІВ

2.1. FIR фільтр

Фільтр зі скінченною імпульсною характеристикою (FIR) – один із видів лінійних цифрових фільтрів, характерною особливістю якого є обмеженість за часом його імпульсної характеристики (з якогось моменту часу вона стає точно рівною нулю). Такий фільтр називають ще нерекурсивним через відсутність зворотного зв'язку.

FIR-фільтр визначається наступною функцією (1).

$$y[n] = \sum_{i=0}^{N} b_i * x[n-i]\,, \tag{1}$$

де x[n] – вхідний сигнал, y[n] – вихідний сигнал, N – порядок фільтру, b_i – коефіцієнт.

Виходячи з цієї функції можна сказати, що значення фільтру в будь-який момент часу є значенням відгуку на миттєве значення входу та сума всіх поступово затухаючих відгуків N попередніх значень входу. Виходячи з цієї функції з часом вихідний сигнал стане рівним нулю. На блоковій діаграмі (рис. 1) можна бачити затримку сигналу позначеною z^{-1} [1].

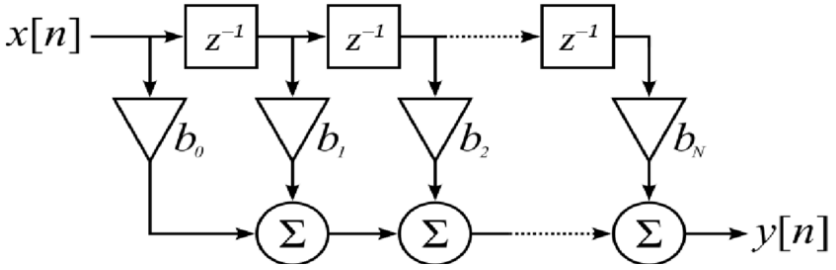


Рисунок 1. Блок діаграма FIR-фільтру

Властивості FIR-фільтрів:

- Не вимагають зворотнього зв’язку. Завдяки цьому похибка обчислень не накопичується.
- Стабільні, це означає що амплітуда сигналу ніколи не перевищує певне значення B, оскільки вихідний сигнал є сумою.
- Дозволяють досягнути лінійної фази використовуючи коефіцієнти симетричні відносно нуля. Лінійна фаза означає, що всі частотні компоненти вхідного сигналу будуть однаково зсунуті в часі. Це дозволяє легко перетворювати звуковий сигнал без спотворень.

2.2. Адаптивний фільтр LMS

Адаптивні фільтри в обробці звуку використовуються для динамічної корекції аудіосигналів у змінних акустичних умовах, забезпечуючи автоматичне підлаштування фільтра під поточні характеристики звуку та шуму. У контексті систем Master Hearing Aid (багатофункціональних слухових апаратів), адаптивні фільтри особливо важливі для покращення розбірливості мови та зменшення фонових шумів. Вони дозволяють в реальному часі оптимізувати обробку звуку, враховуючи індивідуальні потреби користувача та змінні акустичні умови, без необхідності ручного налаштування параметрів. Загальна робота адаптивного фільтру зображена на рис. 2.

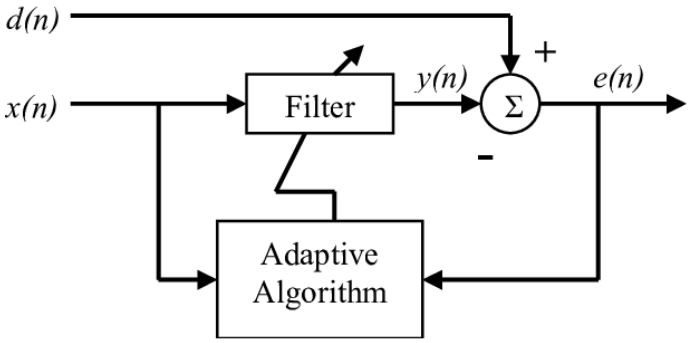


Рисунок 2. Блок схема роботи адаптивного фільтру.

LMS (least mean squares) – це клас алгоритмів, що намагаються відтворити бажаний фільтр підбором коефіцієнтів для зменшення найменшої середньо квадратичної похибки вихідного сигналу (різниця між вихідним та бажаним сигналом).

Алгоритм оснований на градієнтному спуску, тому його можна визначити наступною формулою (2).

$$W_{n+1} = W_n - \mu\nabla\varepsilon[n], \tag{2}$$

де W – ваги, μ – крок, ε – середньоквадратична похибка.

2.3. Normalized LMS

Основним недоліком класичного LMS є чутливість до масштабування вхідних даних. Через це може бути важко або майже неможливо підібрати learning rate [2]. Normalized least mean square (NLMS) фільтр – варіант алгоритму, що вирішує цю проблему нормалізацією вхідного сигналу.

В цьому варіанті алгоритму крок μ замінюється формулою (3).

$$\varphi(n) = \frac{\mu}{\epsilon + \|x(n)\|^2}, \tag{3}$$

де ϵ – мале додатне число, що забезпечує стабільність, $\|x(n)\|^2 -$ норма вхідного сигналу.

2.4. Адаптивний фільтр RLS

Іншим варіантом адаптивного фільтру є алгоритм RLS, який використовує зважену лінійну регресію для зміни коефіцієнтів фільтру. Головною перевагою RLS є вища швидкість збіжності (рис. 3) [3], проте має недолік у необхідності більшої обчислювальної потужності, що в контексті енергоефективних систем має вирішальне значення.

3. ОПТИМІЗАЦІЯ АДАПТИВНИХ ФІЛЬТРІВ ДЛЯ ВИКОРИСТАННЯ У ВІДКРИТИХ ПЛАТФОРМАХ СЛУХОПРОТЕЗУВАННЯ

3.1. Використання адаптивних фільтрів на відкритій платформі openMHA

OpenMHA є однією з найпопулярніших платформ для розробки програмного забезпечення, спеціалізованого на обробці звукових сигналів в режимі реального часу. В цій роботі openMHA використовується як приклад реалізації та застосування адаптивних фільтрів. Платформа пристосована для роботи на архітектурі ARM. Одним з можливих апаратних середовищ є Raspberry PI. В цій роботі за допомогою мови Python проводиться математичне дослідження фільтру та його можливих оптимізацій. Фінальне тестування проводиться на більш бюджетній платформі ESP32 на відміну від ARM, що є важливим фактором в контексті України.

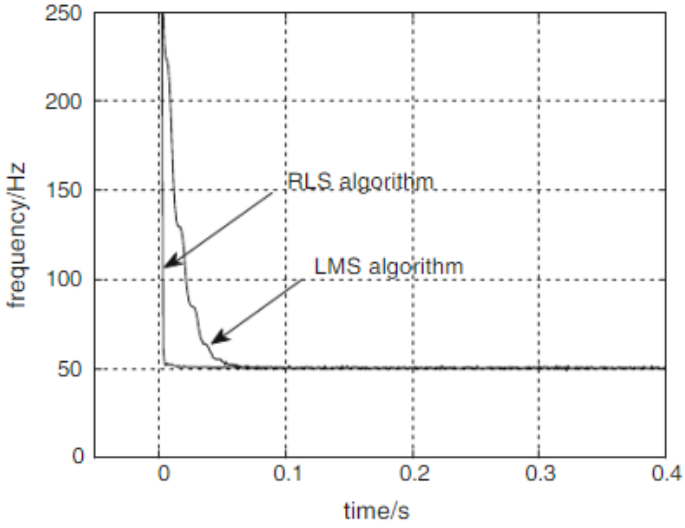


Рисунок 3. Графік збіжності RLS та LMS алгоритму

При виконанні роботи було досліджену одну з частин архітектури openMHA – плагіни. Було розглянуто плагіни, що використовують адаптивні фільтри та їх практичне застосування.

gsc_adaptive_stage – плагін, що реалізовує стадію адаптивного фільтра на основі NLMS для адаптивного променеформувач. Променеформувач – технологія, що дозволяє адаптувати чутливість сенсорів у певному напрямку.

adaptive_feedback_canceller – плагін, для гасіння зворотнього зв'язку (небажаний сигнал, що виникає при потраплянні вихідного сигналу з динаміка в мікрофон). Оснований на алгоритмі NLMS.

nlms_wave – реалізує NLMS алгоритм для перерахування коефіцієнтів фільтру та наближення вихідного сигналу до цільового.

prediction_error – плагін для затримки зворотнього зв'язку, базується на NLMS.

При дослідженні не було виявлено плагінів, що використовують інші адаптивні фільтри, на кшталт RLS [4].

З огляду на існуючі плагіни, оптимізація алгоритму NLMS для підвищення ефективності обчислень та швидкості збіжності може суттєво вплинути на розробку апаратів для слухопротезування. Розглянемо можливі варіанти:

Підбір розміру кроку сильно впливає на збіжність алгоритму. Використання більшого кроку збільшує збіжність, проте занадто великий крок може призвести до нестабільності та розбіжності алгоритму.

Використання leak(витік) параметру. Цей параметр дозволяє алгоритму швидше збігатися при високій дисперсії власних значень вхідних даних. Параметр регулюється за допомогою порівняння похибку після застосування Leaky LMS та звичайного LMS. Якщо Leaky LMS показує себе краще, leak збільшується. Це дозволяє алгоритму автоматично підлаштовуватися під різні умови [5].

Іншим підходом є використання норми градієнта вектору для визначення розміру кроку. На початку, коли похибка велика, градієнт також великий, що призводить до великого розміру кроку і більшої швидкості збіжності.

4. ВИСНОВКИ

У статті розглянуто основні типи фільтрів, що застосовуються для обробки звуку, зокрема фільтри зі скінченною імпульсною характеристикою (FIR) та адаптивні фільтри

(LMS, NLMS, RLS). Особливу увагу приділено адаптивним фільтрам, які є важливими для систем слухопротезування, адже вони дозволяють підлаштовувати обробку сигналів у режимі реального часу під змінні акустичні умови. Це забезпечує підвищення якості обробки аудіосигналу, зокрема покращення розбірливості мови та зменшення шумів. Розглянуті фільтри мають різні властивості, зокрема LMS та NLMS адаптуються до вхідного сигналу, а RLS забезпечує високу швидкість збіжності за рахунок підвищеної обчислювальної складності.

У межах відкритої платформи openMHA було досліджено наявні плагіни, які реалізують алгоритми адаптивного фільтра NLMS, а також їхні прикладні сфери застосування. Було визначено потенційні методи для покращення збіжності та ефективності алгоритму LMS, зокрема підходи для оптимального вибору розміру кроку.

Розроблено план тестування математичної моделі на мові програмування Python. Остаточне тестування проводиться на платформі ESP32, з огляду на доступність та сумісність цієї платформи у порівнянні з ARM-архітектурою першоджерела.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Kumar N. Optimal Design of FIR and IIR Filters using some Evolutionary Algorithms. 2013.
2. Haykin S. Adaptive filter theory. Upper Saddle River, NJ : Prentice Hall, 2002.
3. Frequency Estimation on Power System Using Recursive-Least-Squares Approach / L. Li et al. *Proceedings of the 2012 International Conference on Information Technology and Software Engineering* / ed. by W. Lu et al. Berlin, Heidelberg, 2013. P. 11–18.
4. *Open community platform for hearing aid algorithm research. Documentation of openMHA plugins.* URL: https://www.openmha.org/docs/openMHA_plugins.pdf (дата звернення: 14.11.2024).
5. Kamenetsky M., Widrow B. A variable leaky LMS adaptive algorithm. *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers, 2004.* 2004. P. 125–128 Vol.1. URL: https://doi.org/10.1109/ACSSC.2004.1399103.

ПЕРЕХІД ВІД МОДЕЛІ GAME-AS-A-PRODUCT ДО GAME-AS-A-SERVICE

Шпадківський О.В.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

vaivipir@gmail.com

Дослідження аналізує перехід від ігрової моделі Game-as-a-Product (GaaP) до ігрової моделі Game-as-a-Service (GaaS). Метою дослідження є аналіз ключових аспектів даного переходу, використовуючи методи порівняння, аналізу та узагальнення. На основі отриманих даних, було визначено, що модель GaaS стимулюється розвитком хмарних сервісів, інтернет-технологій, мобільних платформ та штучного інтелекту. Останнє, у свою чергу, дозволяє якісно персоналізувати ігровий досвід. Важливою складовою є безперервне залучення гравців, яке забезпечується за допомогою зворотнього зв'язку, мікротранзакцій, підписок та внутрішньоігрових покупок.

Ключові слова: Game-as-a-Product (GaaP), Game-as-a-Service (GaaS), персоналізація ігрового досвіду, ігрова індустрія.

1. ВСТУП

Протягом останніх десятиліть ігрова індустрія зазнала суттєвих трансформацій, обумовлених як технологічними інноваціями, так і еволюцією бізнес-моделей. Однією з найбільш значущих змін став перехід від концепції *Game-as-a-Product* (далі – GaaP, "гра як продукт") до *Game-as-a-Service* (далі – GaaS, "гра як сервіс").

Перехід від GaaP до GaaS відображає новітній підхід до розробки ігор, заснований не лише на створенні завершеного продукту, але й на формуванні гнучкої та динамічної екосистеми. Ця екосистема постійно оновлюється і адаптується до потреб користувачів, що дозволяє залучати нову аудиторію та підтримувати інтерес існуючих гравців протягом тривалого часу.

Актуальність дослідження обумовлюється наявною потребою у розгляді та детальному аналізі особливостей переходу від ігрової моделі *Game-as-a-Product* (далі – GaaP, "гра як продукт") до ігрової моделі *Game-as-a-Service* (далі – GaaS, "гра як сервіс"), котрий є однією з найбільш значущих для ігрової індустріх за останнє десятиліття. **Метою дослідження** є розкрити та проаналізувати основні аспекти переходу від ігрової моделі GaaP до ігрової моделі GaaS.

2. ПЕРЕХІД ВІД ГРИ ЯК ПРОДУКТУ ДО ГРИ ЯК СЕРВІСУ: ТЕОРЕТИЧНИЙ АНАЛІЗ

Еволюція ігрової індустрії від ігрової моделі GaaP до GaaS відображає глибокі трансформації, зумовлені технологічними інноваціями, змінами в бізнес-моделях та зростаючими вимогами споживачів. Спочатку ігри поширювалися через фізичні носії, такі як картриджі та диски, і були завершеними продуктами без можливості оновлення після релізу. Розвиток персональних комп'ютерів та Інтернету в 1990-х роках започаткував перехід до моделей з регулярними оновленнями та підписками, як у випадку з Ultima Online та EverQuest

[1]. Запуск цифрових платформ, зокрема Steam у 2003 році, радикально змінив спосіб дистрибуції ігор, дозволивши розробникам безпосередньо поширювати оновлення та додатковий контент. Мобільна революція початку 2010-х років та поява платформ App Store і Google Play впровадили моделі монетизації на кшталт free-to-play з мікротранзакціями, що сприяло популярності ігор на зразок Angry Birds та Clash of Clans. Високошвидкісний Інтернет та технології серверної інфраструктури дозволили створювати масштабні багатокористувацькі проєкти, такі як World of Warcraft, що успішно використовували модель підписки та регулярних оновлень контенту [2]. Хмарні технології та стрімінгові сервіси, такі як Playstation Now та Xbox Cloud розширили можливості GaaS, надаючи доступ до високоякісних ігор без потреби в потужному апаратному забезпеченні. Еволюція бізнес-моделей, включаючи free-to-play, підписки, мікротранзакції та сезонні пропуски, відіграла ключову роль у переході до GaaS, забезпечуючи стабільніший дохід для розробників [3]. Соціальні медіа та стрімінгові платформи створили нові канали для взаємодії з аудиторією, підсилюючи важливість активних спільнот у успіху сучасних ігор. Інтеграція технологій штучного інтелекту дозволила створювати адаптивні та персоналізовані ігрові середовища, підвищуючи рівень залученості гравців [5]. Перехід до GaaS обумовлений розвитком цифрової дистрибуції та високошвидкісного Інтернету, зміною споживчих уподобань, еволюцією бізнес-моделей, технологічними інноваціями, соціальними факторами та глобалізацією ринку [4]. Порівняння моделей GaaP та GaaS показує фундаментальні зміни у підходах до розробки, монетизації та взаємодії з гравцями: GaaS пропонує регулярні оновлення, різноманітні моделі монетизації, довготривалу взаємодію та акцент на соціальних аспектах, тоді як GaaP базується на одноразових продажах завершених продуктів. Хоча GaaP залишається актуальною для певних жанрів та аудиторій, GaaS стає домінуючою моделлю, відповідаючи сучасним вимогам гравців до динамічного та інтерактивного контенту. Цей перехід не лише змінив спосіб створення та розповсюдження ігор, але й вплинув на саму природу ігрового досвіду. Зокрема, поглибив зв’язок між гравцям, стаючи більш соціально орієнтованим. Окрім того, досвід гравця став більш інтерактивним та персоналізованим.

3. ТЕХНОЛОГІЧНІ ТЕНДЕНЦІЇ У СФЕРІ GAAS

Модель GaaS спричинила значні інновації у розробці та видавництві ігор, впровадивши нові підходи, що підвищують якість продукту та покращують досвід гравців. Однією з ключових інновацій є динамічний контент та адаптивність ігрового середовища, де регулярні оновлення та розширення надають гравцям нові можливості, підтримуючи їхній інтерес до гри. Адаптивність досягається за допомогою штучного інтелекту та алгоритмів машинного навчання, які аналізують поведінку гравців і відповідно налаштовують складність або пропонують індивідуальні завдання, що підвищує персоналізацію ігрового процесу. Впровадження методологій Agile (базується на ітеративній розробці) та DevOps(опирається на збирання, налаштування та розгортання програмного забезпечення) у розробці ігор дозволяє здійснювати поетапну розробку з можливістю регулярного вдосконалення, швидко реагуючи на зміни та побажання гравців. DevOps сприяє безперервному вдосконаленню та автоматизації процесів розробки та розгортання, що знижує час між ітераціями та спрощує внесення змін без перерв у роботі гри. Прямий зворотній зв'язок з гравцями через соціальні мережі, форуми та внутрішньоігрові опитування забезпечує швидке адаптування гри до потреб аудиторії та підвищує рівень залученості користувачів, створюючи довірчі відносини та спільне відчуття причетності до процесу розробки. Інтеграція аналітики та великих даних дозволяє розробникам відстежувати поведінкові патерни гравців, ухвалювати обґрунтовані рішення щодо модифікації контенту та покращувати користувацький досвід, а також краще сегментувати аудиторію для пропозиції індивідуалізованих можливостей. Технологічна

підтримка для регулярних оновлень, з використанням хмарних технологій та сервісів безперервної інтеграції та доставки, забезпечує оперативне оновлення ігрового контенту без значних інфраструктурних витрат та підтримує стабільність гри.

Також у контексті GaaS запроваджуються нові бізнес-моделі та методи монетизації, які забезпечують стабільний потік доходів та подовжують життєвий цикл гри. Мікротранзакції дозволяють гравцям здійснювати невеликі покупки всередині гри, надаючи можливості для персоналізації та покращення ігрового досвіду. Підписні сервіси забезпечують стабільний потік доходів через регулярну плату за доступ до ексклюзивного контенту, спеціальних функцій або бонусів, мотивуючи гравців регулярно повертатися до гри. Внутрішньоігрові покупки та платний контент розширюють функціональні можливості та забезпечують постійне надходження доходів навіть після початкового завантаження гри. Модель free-to-play з платним контентом має широкий потік нових гравців, забезпечуючи високий рівень залучення та підтримуючи баланс між безкоштовним та платним контентом.

Вплив таких технологічних трендів як штучний інтелект, доповнена та віртуальна реальність, хмарні технології та Інтернет речей, значно розширив можливості GaaS. Штучний інтелект прияє у персоналізації ігрового досвіду та підвищує інтерактивність. Індивідуалізія та “гнучкість” контенту досягається через ретельний аналіз широкого набору маркерів, які відзначають внутрішньоігрову поведінку гравців. Доповнена та віртуальна реальність створюють нові формати взаємодії, надаючи користувачам відчуття присутності у віртуальних світах та збагачуючи реальне середовище інтерактивними елементами. Хмарні технології забезпечують зберігання великого обсягу даних, знижують витрати на інфраструктуру та дозволяють запускати ігри на віддалених серверах, розширюючи можливості GaaS. Інтеграція з Інтернетом речей відкриває нові перспективи для інтеграції ігрового процесу з фізичними пристроями, підвищуючи взаємодію між віртуальним та фізичним світом.

4. ВПЛИВ GAAS НА ІГРОВУ ІНДУСТРІЮ ТА КОРИСТУВАЧІВ

Впровадження моделі «гра як сервіс» (GaaS) суттєво змінило поведінку гравців, їхні очікування та взаємодію з ігровим контентом, переходячи від одноразового проходження до тривалої та регулярної участі. Гравці тепер очікують постійних оновлень, нових функцій та унікальних подій, що підтримують їхній інтерес. Підвищена роль соціальної взаємодії стала невід'ємною складовою GaaS, де гравці прагнуть співпраці, конкуренції та взаємодії зі спільнотою, що підсилює їхню залученість та значущість ігрових досягнень. Бажання персоналізації та індивідуалізації контенту стимулює гравців до кастомізації персонажів і придбання ексклюзивних предметів, що підвищує рівень залученості та створює додаткові джерела доходу для розробників. Зростання вимог до якості обслуговування та підтримки формує новий стандарт обслуговування, який вимагає від розробників оперативної реакції на запити аудиторії та забезпечення надійності сервісу.

Етичні та соціальні аспекти GaaS викликають дискусії щодо впливу на психічне та фізичне здоров'я користувачів, особливо в контексті можливої залежності від ігор [6]. Використання механік, що стимулюють тривалу взаємодію, підкреслює необхідність відповідального підходу до дизайну ігор, включаючи надання гравцям інструментів для контролю часу, проведеного у грі. Монетизація через мікротранзакції та внутрішньоігрові покупки породжує питання щодо етичності, особливо якщо вона створює нерівність між платними та безкоштовними користувачами [7]. На наш погялад, важливим є дотримання балансу, котрий гарантував би чесність ігрового процесу та зберігав довіру гравців. Використання даних гравців для персоналізації контенту піднімає питання конфіденційності, вимагаючи від розробників дотримання етичних стандартів при зборі та зберіганні інформації. Майбутні перспективи GaaS пов'язані з розширенням можливостей завдяки новим

технологіям, таким як штучний інтелект, хмарні сервіси, доповнена реальність та Інтернет речей, які сприятимуть створенню більш інтерактивних та персоналізованих ігор. Однак розробники зіткнуться з викликами перенасичення ринку та посилення конкуренції, що вимагатиме інноваційного підходу та унікального контенту для привернення уваги аудиторії. Забезпечення сталого доходу та лояльності гравців стане критичним, оскільки підтримка інтересу аудиторії потребуватиме регулярних оновлень, високого рівня обслуговування та уважного відстеження потреб користувачів. Етичні виклики, пов'язані з монетизацією та можливістю залежності від ігор, вимагають від розробників відповідального підходу до дизайну ігрових механік та впровадження опцій для контролю часу у грі. Таким чином, модель GaaS продовжує еволюціонувати, впливаючи на поведінку гравців та встановлюючи нові стандарти в індустрії ігор. Успіх майбутніх проєктів залежатиме від здатності розробників збалансувати інновації, етичні аспекти та задоволення потреб гравців, підтримуючи довіру та лояльність аудиторії в умовах зростаючої конкуренції.

5. ВИСНОВКИ

Перехід від моделі Game-as-a-Product (GaaP) до Game-as-a-Service (GaaS) став однією з найбільших змін у сучасній ігровій індустрії, що кардинально змінила підходи до розробки, монетизації та взаємодії з користувачами. Впровадження GaaS дозволило розробникам створити динамічні та адаптивні ігри, які постійно оновлюються і підтримують інтерес гравців протягом тривалого періоду. Такий підхід також забезпечує стабільний потік доходів через різноманітні форми монетизації, такі як мікротранзакції, підписки та внутрішньоігрові покупки.

У результаті дослідження вдалось визначити, що, що успішність моделі GaaS значною мірою залежить від впровадження новітніх технологій, таких як штучний інтелект, доповнена та віртуальна реальність, хмарні сервіси та Internet of Things. Ці інновації сприяють створенню персоналізованого, інтерактивного досвіду, що відповідає сучасним очікуванням гравців. Технологічний прогрес також відкриває нові можливості для розробників, зокрема, у створенні інтеграцій з повсякденними девайсами та забезпеченні доступу до високоякісного контенту навіть на менш потужних пристроях.

Однак разом із перевагами модель GaaS створює й певні виклики. Серед основних проблем можна виділити необхідність підтримки високих стандартів обслуговування, вирішення питань конфіденційності та забезпечення балансу між платним і безкоштовним контентом. Гравці стали більш обізнаними і критично налаштованими до агресивної монетизації, що підвищує важливість відповідального підходу до дизайну ігор, включаючи можливості для контролю часу в грі та мінімізацію ризиків залежності.

Таким чином, майбутній успіх проєктів GaaS залежатиме від здатності розробників підтримувати баланс між економічними інтересами та потребами аудиторії. Важливим аспектом стане врахування етичних питань, пов'язаних із тривалою взаємодією гравців із контентом та монетизаційними стратегіями. Розробники, які зможуть зберігати довіру аудиторії та постійно вдосконалювати ігровий процес, мають високі шанси на досягнення лояльності користувачів і здобуття довготривалого успіху.

Тож модель GaaS не лише змінює спосіб створення ігор, але й формує нові стандарти взаємодії між розробниками та гравцями, які визначатимуть майбутнє ігрової індустрії.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Kobayashi, Nobushige, Koyama, Yuhsuke, 2020, The Early History of the Hobbyist Production Field of Video Games and its Impacts on the Establishment of Japan’s Video Game Industries, 73–82 p.

2. Kerr A. Global games: production circulation and policy in the networked era. Routledge, 2016, p. 65.

3. Davidovici-Nora M. Innovation in business models in the video game industry: free-to-play or the gaming experience as a service. Computer Games Journal, 2013, vol. 2, no. 3, pp. 22–51.

4. Wilhelmsson U., Wang W., Zhang R., et al. Shift from game-as-a-product to game-as-a-service research trends. Service Oriented Computing and Applications, 2022, vol. 16, pp. 79–81.

5. Morton D. What Netflix's entrance in gaming means for big and small gamers alike. Forbes, 2021. Available at: https://www.forbes.com/sites/forbestechcouncil/2021/11/08/what-netflixs-entrance-in-gaming-means-for-big-and-small-gamers-alike/ (дата звернення: 14.11.2024)

6. Mohammad, S., Jan, R. A., & Alsaedi, S. L. (2023). Symptoms, Mechanisms, and Treatments of Video Game Addiction. Cureus, 15(3).

7. Lelonek-Kulet, B., Bartczuk, R. P., & Wiechetek, M. (2020). Pay for play – behavioural patterns of pay-to-win gaming. Computers in Human Behavior, 115(1).

ПОШУК ВІДПОВІДНОСТІ МІЖ ТОЧКАМИ ТА ЇХ ВІДСТЕЖЕННЯ

Бірук С.В.[1], Шаповал Н.В.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] biruk.serhii@lll.kpi.ua

Метою дослідження є підвищення точності та надійності відстеження точок у складних відеосередовищах. Запропоновано модифікацію для метода, який використовує згорткові нейронні мережі для обчислення багатомасштабних карт ознак та модернізовану 4D кореляцію для аналізу схожості областей. Отримані результати демонструють високу точність відстеження навіть при оклюзіях та швидких рухах, що свідчить про наукову новизну та практичну значущість методу для автономних систем та безпілотних літальних апаратів.

Ключові слова: відстеження точок, глибоке навчання, 4D кореляція, траєкторія, оптичний потік.

1. ВСТУП

Розробка удосконалених алгоритмів, здатних точно та надійно відстежувати точки в складних умовах, є критично важливою для підвищення ефективності автономних систем. У сучасних конфліктах безпілотні літальні апарати відіграють ключову роль у розвідці, спостереженні та забезпеченні ситуаційної обізнаності на полі бою. Вони потребують високої точності відстеження об'єктів для безпечної навігації, уникнення перешкод і виконання складних маневрів у реальному часі. Таким чином, дослідження в області пошуку відповідностей між точками та їх відстеження у складних відеосередовищах має високу практичну значущість і відповідає нагальним потребам сучасних технологій оборони та безпеки.

Розроблений метод дозволяє ефективно відстежувати рух кількох точок у відео, використовуючи поєднання згорткових нейронних мереж для обчислення багатомасштабних карт ознак та ітеративного уточнення траєкторій за допомогою Transformer. Це забезпечує високу точність та надійність відстеження навіть у складних умовах, таких як оклюзії, швидкі рухи та зміни освітлення. Отримані результати можуть бути застосовані в автономних системах, безпілотних літальних апаратах та інших областях, де необхідне точне та стійке відстеження об'єктів у реальному часі.

2. ПОВ’ЯЗАНІ РОБОТИ

Ранні підходи до відстеження покладалися на оптичний потік, який оцінює рух на рівні пікселів, припускаючи мінімальний рух між послідовними кадрами, викликаний рухом об'єкта або камери. Метою є обчислення наближення до 2D-поля руху – проєкції 3D-швидкостей поверхневих точок на зображувальну площину.

Наразі існує велика кількість варіацій реалізації оптичного потоку. Найбільш широко цитовані методи обчислення оптичного потоку включають методи Хорна та Шунка, Лукаса та Канаде, Нагеля, Анандама. Ще одним варіантом для відстеження руху у відео є відстеження ознак (feature tracking), такі як метод Канаде-Лукаса-Томасі (KLT Tracker), SIFT, SURF та інші.

Відстеження ознак слідкує за розрідженим набором важливих точок зображення протягом багатьох кадрів, тоді як оптичний потік оцінює щільне поле руху між кадрами.

Задачу відповідності точок можна описати як щільне відстеження ознак або далекодійний оптичний потік. Тут слово "точка" позначає об'єкти, які відстежуються у відео. Слово "піксель" застосовується в більш загальному сенсі, маючи на увазі будь-яку окрему клітинку на сітці зображення. Потрібно відстежувати траєкторію точок з більшою щільністю протягом усього відео, оскільки представлення оптичного потоку краще підходить для послідовних пар кадрів, а не для довгих послідовностей. Важливо розуміти, що хоча задаються цілі для відстеження як координати одного пікселя, успішне відстеження вимагає врахування навколишнього контексту навколо цієї точки. Тобто модель повинна аналізувати не лише сам піксель, але й сусідні області, щоб точно визначати положення цілі на кожному кадрі.

У 2022 році було вперше представлено реалізацію PIPs [1], яка спрямувала розвиток відстеження точок в область глибокого навчання та нейронних мереж. Серед останніх досліджень – TAPIR [2], LocoTrack [3] та CoTracker [4].

3. ОПИС МЕТОДУ

Для відстеження руху кількох точок у відео, яке складається з T кадрів. Модель працює з N незалежними точками одночасно, які задаються координатами (x_1, y_1) пікселя на першому кадрі – це точка, яку потрібно відстежувати надалі. І мета – знайти координати x_t, y_t цієї точки на кожному наступному кадрі t, тобто простежити її рух у часі, тобто отримати траєкторію $U = \{U\}_{t=1}^{T}$, де $U_t \in R^2$, та відповідні ймовірності оклюзії $O = \{O_t\}_{t=1}^{T}$, де $O_t \in [0,1]$. Відповідно до попередніх робіт [2–4], метод передбачає траєкторію в два етапи: етап ініціалізації, після якого слідує етап уточнення.

Спочатку для кожного кадру відео (позначені як $I_1, \dots, I_{T-1}$) обчислюються карти ознак за допомогою згорткової нейронної мережі. Кожне зображення пропускається через мережу, яка витягує важливі характеристики (ознаки) з зображення. Отримані карти ознак позначаються як $F_t = F(I_t)$, де ($t = 1, \dots, T$) (кожен кадр від 1 до T). Ці карти ознак містять стислу інформацію про кожен кадр і мають розміри $C \times H \times W$, де C – кількість каналів ознак, а H і W – висота і ширина кадру. Обчислюємо карти ознак на S = 3 різних масштабах. Це означає, що для кожного кадру отримується кілька версій карти ознак з різними рівнями деталізації. Кожна масштабована карта ознак F_t^s має розміри:

$$F_t^s \in R^{d \times \frac{H}{k \cdot 2^{s-1}} \times \frac{W}{k \cdot 2^{s-1}}}, \quad s = 1,2,3.$$

Це дозволяє моделі аналізувати зображення на різних рівнях масштабу, що покращує її здатність розпізнавати об'єкти різних розмірів і враховувати як дрібні деталі, так і загальну структуру сцени.

Кожна f_t^s містить сітку розміром $(2D + 1) \times (2D + 1)$ точок, де D визначає радіус області навколо точки. Ця сітка охоплює сусідні пікселі навколо P_t, дозволяючи нам враховувати інформацію з навколишнього контексту. Ознаки f_t^s отримуються шляхом вибірки карти ознак F_t^s у дробових координатах, використовуючи білінійну інтерполяцію.

На наступному етапі обчислюється 4D кореляція. Для цього обчислюється кореляція між ознаками навколо цієї точки запиту на кадрі t та ознаками навколо поточних оцінок траєкторії $P_t = (x_t, y_t)$ на інших кадрах. Кореляція показує, наскільки схожі між собою ці області, і допомагає визначити, де саме знаходиться точка на кожному кадрі.

На відміну від 4D кореляції, що використовується в LocoTrack [3], в цьому дослідженні запропонована модифікація, де обчислюється 4D кореляція між ознаками навколо точки запиту $f_{t_q}^s$ і ознаками навколо поточної позиції f_t^s для кожного масштабу s. Тобто

порівнюємо кожну ознаку з області навколо заданої точки з кожною ознакою з області навколо поточної позиції на іншому кадрі. Результатом є масив даних розміром $(2D+1)^4$, який містить всі можливі комбінації порівнянь ознак. Це допомагає моделі точніше визначити, де знаходиться наша ціль на кожному кадрі, навіть якщо вона зміщується, обертається або змінюється її зовнішній вигляд

Оскільки обсяг даних після обчислення 4D кореляції дуже великий, його розмірність зменшується за допомогою 4D кореляційного енкодера. Енкодер приймає кореляційні дані та перетворює їх у компактніше представлення фіксованого розміру E_t^k.

Таким чином, отримуємо компактні та інформативні кореляційні ознаки, які можуть бути ефективно використані в наступному етапі ітеративного уточнення точок.

Процес ітеративного оновлення використовується для покращення відстеження об'єкта у відео, покращуючи траєкторію руху об'єкта на кожному кроці, де використовується трансформер, який отримує вхідні дані у вигляді кореляційних вбудовувань E_t^k, поточної оцінки траєкторії об'єкта U^k, і статусу оклюзії O^k.

Замість того, щоб розглядати абсолютні позиції об'єкта, використовується різниця позицій між суміжними кадрами. Різниця позицій між суміжними кадрами $(U_t^k - U_{t-1}^k)$ і $(U_{t+1}^k - U_t^k)$ використовується як вхідні дані до трансформера. Після обробки всіх вхідних даних передбачається оновлення траєкторії об'єкта ΔT^k та оновлення статусу оклюзії ΔO^k. Далі передбачені оновлення застосовуються до поточної траєкторії та статусу оклюзії:

$$T^{k+1} \coloneqq T^k + \Delta T^k, \quad O^{k+1} \coloneqq O^k + \Delta O^k$$

Цей процес повторюється протягом K ітерацій, поступово уточнюючи траєкторію U^K і статус оклюзії O^K. Кожна ітерація дозволяє моделі краще зрозуміти, де саме знаходиться об'єкт і чи він видимий, що допомагає більш точно відстежувати об'єкт у відео, навіть коли траєкторія є складною або об'єкт частково закривається іншими предметами.

Для порівняння результатів використовуються стандартні TAP-Vid[5] метрики: точність позиції $(< {\delta_{avg}}^x)$, точність оклюзії (OA) та середній показник Жаккара (AJ). $< {\delta_{avg}}^x$ – обчислює точність позиції для точок, видимих в еталоні. Він обчислює відсоток правильних точок, усереднений за значеннями порогової помилки 1, 2, 4, 8 і 16 пікселів. OA — представляє середню точність бінарних результатів класифікації для оклюзії. AJ – це метрика, яка оцінює як точність позиції, так і точність оклюзії. Після 8000 кроків тренування були отримані результати на датасеті Kinetics: 48.7 AJ, 82.3 OA та 62.4 $< {\delta_{avg}}^x$.

4. ВИСНОВОК

У цій роботі представлено модифікацію до методу LocoTrack для відстеження руху кількох точок у відео, який поєднує згорткові нейронні мережі та ітеративне уточнення. Запропонований підхід дозволяє ефективно відстежувати траєкторії точок протягом усього відео, враховуючи різні масштаби та можливі оклюзії. Використання багатомасштабних карт ознак та 4D кореляції покращує точність відстеження в порівнянні з традиційними методами оптичного потоку та відстеження ознак. Отримані результати свідчать про потенціал нашого методу в галузі комп'ютерного зору та відеоаналізу.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Adam W. H., & Zhaoyuan F. Particle Video Revisited: Tracking Through Occlusions Using Point Trajectories, 2022

2. Carl Doersch, Yi Yang, Mel Vecerik, Dilara Gokay, Ankush Gupta, Yusuf Aytar, Joao Carreira, and Andrew Zisserman. TAPIR: Tracking any point with per-frame initialization and

temporal refinement. arXiv, 2306.08637, 2023

3. Seokju Cho, Jiahui Huang, Jisu Nam, Honggyu An, Seungryong Kim, and Joon-Young Lee. Local all-pair correspondence for point tracking. Proc. ECCV, 2024.

4. Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. Proc. ECCV, 2024.

5. Doersch, C., Gupta, A., Markeeva, L., Recasens, A., Smaira, L., Aytar, Y., Carreira, J., Zisserman, A., Yang, Y.: Tap-vid: A benchmark for tracking any point in a video. Advances in Neural Information Processing Systems 35, 13610–13626, 2022

НАПІВКЕРОВАНЕ МОДЕЛЮВАННЯ ВПЛИВУ ДЛЯ ЕЛЕКТРОННОЇ КОМЕРЦІЇ

Заіка Б.Ю.[1], Шаповал Н.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] zaikabohdan5@gmail.com [0009-0001-9567-8361],
[2] shovgun@gmail.com [0000-0002-8509-6886]

У сучасній електронній комерції компанії покладаються на аналіз даних для розуміння поведінки споживачів та ефективного впливу на них. Цифрова реклама дозволяє інформувати клієнтів про новинки та спеціальні пропозиції, але неправильне використання маркетингових ресурсів може призвести до зайвих втрат. Класичні методи класифікації не враховують інформацію про контрольну групу, обмежуючи цим точність їх прогнозів. Моделювання впливу застосовує дані про контрольну групу, що робить його більш ефективним, але воно має основною проблемою обмеженість ресурсів та клієнтів. Дана робота пропонує застосування напівкерованого навчання для вирішення цієї проблеми, розкриває переваги такого підходу та демонструє на прикладі результати його використання.

Ключові слова: моделювання впливу, електронна комерція, напівкероване навчання, контрольні та тестові групи.

1. ВСТУП

У сучасному середовищі електронної комерції, де конкуренція постійно зростає і бізнесові умови швидко змінюються, компанії все більше покладаються на аналіз даних для розуміння поведінки споживачів та ефективного впливу на них. Залучення уваги клієнтів, збільшення онлайн-продажів, підтримка конкурентоспроможності та збереження постійного зв'язку з аудиторією роблять маркетингові кампанії ключовим інструментом для інтернет-бізнесу. За допомогою цифрової реклами компанії можуть інформувати споживачів про нові продукти, спеціальні пропозиції, зміни в асортименті та відповідати на їхні запити. Однак нераціональне використання маркетингових ресурсів може призвести до втрати клієнтів і фінансових втрат через надмірну комунікацію або невдалу персоналізацію.

Звичайні методи класифікації зосереджені на визначенні ймовірності того, що користувач здійснить цільову дію після взаємодії з рекламою. На основі таких прогнозів приймаються рішення щодо необхідності контакту з певними користувачами. Проте справжня мета маркетингових кампаній в електронній комерції полягає в оцінці різниці в поведінці користувача за наявності взаємодії з рекламою та без неї. Традиційні підходи не враховують дані про контрольні групи, що обмежує їх ефективність у цьому контексті.

Моделювання впливу, навпаки, використовує контрольні групи для точного оцінювання різниці в ймовірності цільового результату між групами. Це робить його більш придатним для аналізу потенційних клієнтів в електронній комерції. Крім того, такі моделі дозволяють безпосередньо визначити користувачів, взаємодія з якими буде найефективнішою. Цей підхід дає змогу раціональніше використовувати маркетингові ресурси та уникати контактів із клієнтами, яких реклама може відштовхнути від здійснення

покупки [1]. Це і є основним аргументом на користь застосування моделювання впливу для прийняття рішень щодо маркетингових кампаній: можливість прогнозувати вплив взаємодії на користувача, уникати небажаних контактів і визначати, з ким найдоцільніше взаємодіяти для досягнення бізнес-цілей.

У моделюванні впливу часто вагомим викликом є обмежені ресурси та кількість доступних клієнтів, що ускладнює проведення масштабних експериментів та побудову точної моделі. Для вирішення даної проблеми в цій роботі пропонується застосування напівкерованого навчання та висвітлюються переваги такого підходу.

2. ПОСТАНОВКА ЗАДАЧІ МОДЕЛЮВАННЯ ВПЛИВУ В ЕЛЕКТРОННІЙ КОМЕРЦІЇ

Розглянемо бінарну задачу моделювання впливу дій. Існує n клієнтів, на яких досліджується вплив певної дії (наприклад, реклами) на отримання бажаного результату. Вони поділяються на дві групи:

- тестова група (англ. treatment group) – група об'єктів, до яких було застосовано дію;
- контрольна група (англ. control group) – група об'єктів, до яких не було застосовано дію.

Нехай $X = \{x_1, \dots, x_n\} \in \mathbb{R}^{n \times k}$ – значення k ознак, які описують властивості n клієнтів експерименту; $Y = \{y_1, \dots, y_n\} \in \{0,1\}^n$ – цільова змінна, де $y_i = 1$ означає отримання бажаного результату на i-ому клієнті, а $y_i = 0$ – відсутність бажаного результату; $W = \{w_1, \dots, w_n\} \in \{0,1\}^n$ – змінна, яка відображає належність клієнтів до тестової ($w_i = 1$) або контрольної ($w_i = 0$) групи; $P(\dots)$ – ймовірність. Тоді вплив дії на i-ого клієнта визначається як різниця ймовірностей отримання бажаного результату з та без застосування дії на нього:

$$U(x_i) := P(y_i = 1 | x_i; w_i = 1) - P(y_i = 1 | x_i; w_i = 0).$$

Необхідно навчити модель, яка визначатиме значення $U(x_i)$ для наданих об'єктів [2]. Таким чином, основна мета полягає у моделюванні різниці в поведінці щодо виконання цільової дії між тестовими і контрольними групами.

На відміну від звичайної бінарної класифікації, де клієнти діляться на два класи за цільовою змінною, у моделюванні впливу взаємодій також враховується ознака взаємодії з клієнтом. Тому в цьому контексті клієнтів ділять на чотири групи, які зображені на рисунку 1:

1. Схильні до переконання (англ. Persuadable) – клієнти, взаємодія з якими забезпечить бажаний результат, тоді як без взаємодії цього результату не буде. Це єдина група, взаємодія з якою приносить позитивний ефект.
2. Лояльні (англ. Sure Things) – клієнти, які дадуть бажаний результат незалежно від взаємодії. Взаємодія з цією групою не має сенсу.
3. Втрачені (англ. Lost Causes) – клієнти, для яких бажаний результат не буде отриманий, незалежно від взаємодії з ними. Взаємодія з цією групою є невигідною.
4. Сплячі собаки (англ. Sleeping Dogs або Do-Not-Disturbs) – клієнти, взаємодія з якими призведе до відсутності бажаного результату, тоді як без взаємодії результат буде позитивним. Це найгірша група клієнтів для взаємодії.

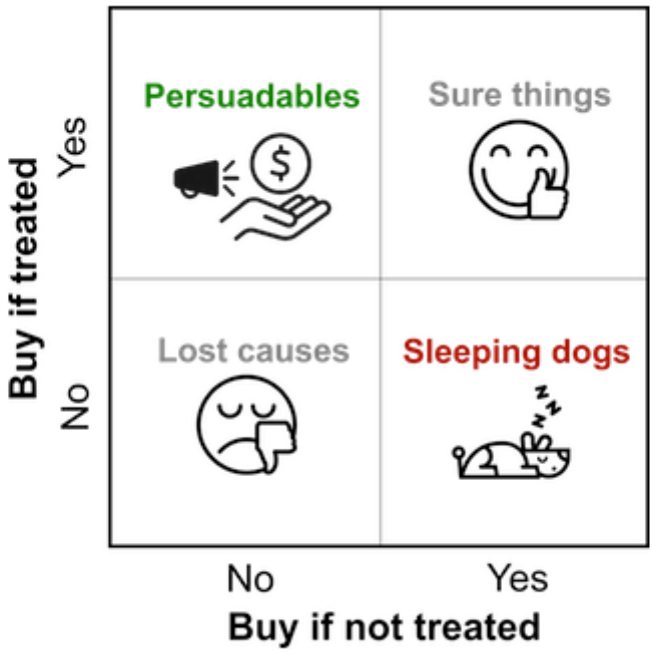


Рисунок 1. Чотири групи клієнтів в моделюванні впливу

Ця класифікація підкреслює переваги моделювання впливу взаємодій для вирішення поставлених завдань порівняно зі звичайними методами моделювання. Звичайні моделі не здатні відрізнити схильних до переконання клієнтів від лояльних, і часто надають пріоритет останнім через їхню більшу ймовірність отримання бажаного результату. Натомість моделі впливу взаємодій дозволяють визначити, з якими клієнтами взаємодія найбільше підвищує ймовірність отримання бажаного результату порівняно зі сценарієм без впливу, що є більш правильним підходом до пріоритизації [3].

3. ОСНОВНІ ВИКЛИКИ ЗАСТОСУВАННЯ МОДЕЛЮВАННЯ ВПЛИВУ В ЕЛЕКТРОННІЙ КОМЕРЦІЇ

Одним із викликів застосування моделювання впливу в електронній комерції є дотримання етичних та правових норм. Проведення експериментів із залученням контрольних груп вимагає відповідального підходу, що враховує баланс між бізнесовими цілями та правами споживачів на конфіденційність. Зокрема, під час розподілу аудиторії на контрольні та експериментальні групи необхідно дотримуватися вимог законодавства щодо збору та використання персональних даних, що може обмежити можливості моделювання [4].

Ще одним важливим аспектом є правильний вибір метрик для оцінки успішності кампаній. У моделюванні впливу метрики повинні точно відображати вплив маркетингових заходів на цільову поведінку користувачів. Якщо вибір метрики є недосконалим або неточним, це може призвести до хибних результатів, що ускладнить процес прийняття рішень та може спричинити нераціональне використання ресурсів [5].

Проблема оцінки та інтерпретації результатів також є вагомою у використанні моделювання впливу. Для успішного застосування моделей важливо не лише отримувати прогнози, але й розуміти, чому саме модель пропонує ті чи інші рішення. Це вимагає

високого рівня експертизи та здатності до інтерпретації складних алгоритмів, що може бути суттєвим бар'єром для аналітиків бізнесу [6].

Складність збору та обробки даних є ще однією серйозною перешкодою для застосування моделювання впливу. Побудова таких моделей вимагає наявності великих обсягів якісних даних, що охоплюють як групи під впливом реклами, так і контрольні групи. Організація такого процесу потребує великих ресурсів та комплексного підходу до управління даними, що може стати значним навантаженням для компанії та ускладнити реалізацію моделювання впливу в комерційних задачах [7].

Остання проблема є ключовою, оскільки у реальних умовах ресурси для здійснення взаємодії та загальна кількість доступних клієнтів є обмеженими. Побудова моделі впливу вимагає проведення випадкового експерименту з вибіркою, яка буде поділена на тестову та контрольну групи для збору необхідних даних. Важливо, щоб розмір цієї вибірки був не великим. По-перше, випадковий експеримент на групі користувачів може непередбачувано змінити їх поведінку відносно компанії. По-друге, менша кількість користувачів в експерименті – це менші витрати на нього, які замість цього можна використати на взаємодію з більшою кількістю користувачів поза експериментом або на інші потреби компанії. По-третє, взаємодія з користувачами поза експериментом відбуватиметься вже на основі розробленої моделі впливу, що зробить взаємодію більш вигідною та ефективною. У маркетинговій сфері це дозволяє мінімізувати витрати на взаємодію та не псувати відносини з більшістю клієнтів, тоді як у медицині критично важливо зменшити ризики та зберегти ресурси, оскільки об'єктами є пацієнти, а взаємодією можуть бути експериментальні ліки.

4. ВИКОРИСТАННЯ НАПІВКЕРОВАНОГО НАВЧАННЯ В МОДЕЛЮВАННІ ВПЛИВУ

Для вирішення проблеми обмеженості ресурсів та клієнтів для взаємодії можна застосувати напівкероване навчання в побудові моделей впливу. Напівкероване навчання є підходом у машинному навчанні, який поєднує використання маркованих і немаркованих даних для виконання певних навчальних завдань. Воно знаходиться між керованим і некерованим навчанням за своєю природою. Основна мета напівкерованих алгоритмів – покращити результативність конкретного завдання, залучаючи додаткову інформацію, яка зазвичай стосується іншого завдання. Наприклад, під час класифікації можна скористатися немаркованими даними, щоб спростити процес класифікації. Аналогічно, методи кластеризації можуть виграти від інформації про те, що певні дані належать одному класу. Як і в машинному навчанні загалом, більшість досліджень у сфері напівкерованого навчання зосереджується на задачах класифікації [8].

Напівкероване навчання вже неодноразово демонструвало свою ефективність у покращенні результатів. Наприклад, метод MixMatch об'єднує різні підходи, зокрема регуляризацію консистентності та створення псевдоміток, щоб максимально використовувати немарковані дані, що дозволяє значно підвищити точність моделей класифікації порівняно з традиційними підходами, які використовують лише марковані дані. Особливо відчутний ефект досягається завдяки застосуванню операцій над даними, таких як обертання, обтинання або зміна кольору, що сприяє навчанню на основі подібності та відмінностей між даними, які мають часткові мітки [9]. Інший підхід полягає у використанні псевдо-маркованих даних для поліпшення класифікації, як у методах кластеризації, де немарковані зразки з високою впевненістю отримують тимчасові мітки. Завдяки цьому навчальні моделі можуть ефективніше використовувати інформацію з немаркованих даних і підвищувати точність передбачень [10].

Застосування напівкерованого навчання в моделюванні впливу дозволяє ефективніше використовувати доступні дані, особливо коли марковані дані обмежені. У реальних умовах часто наявна велика кількість немаркованих даних про користувачів і такий підхід дозволяє залучити ці дані до процесу побудови моделі, підвищуючи її точність без необхідності створення обширних маркованих вибірок. Також напівкерований підхід дозволяє спеціально виділяти лише частину користувачів для участі в експерименті, створюючи марковану вибірку для початкового тренування моделі, а надалі довчати модель на основі немаркованих даних від інших користувачів. Таким чином користувачі з немаркованих даних не піддаються взаємодії, яка може непередбачувано на них вплинути, але водночас допомагають покращити моделі впливу.

Цей підхід також допомагає значно знизити витрати на розмітку даних, яка може бути трудомісткою та коштовною, особливо у великих маркетингових кампаніях. Використовуючи лише обмежену кількість маркованих даних разом із великим обсягом немаркованих, компанії можуть мінімізувати витрати на збір та обробку маркованих даних, зберігаючи при цьому високий рівень точності та ефективності моделі.

Крім того, напівкероване навчання сприяє підвищенню точності моделі завдяки здатності виявляти додаткові закономірності в немаркованих даних, що допомагає у створенні більш точних прогнозів. Це особливо важливо для моделювання впливу, де приховані закономірності в поведінці користувачів можуть залишатися непомітними за обмежених маркованих вибірок.

Також напівкероване навчання сприяє покращенню генералізації та стабільності моделі, роблячи її більш стійкою до шуму в маркованих даних. Немарковані дані слугують додатковим джерелом інформації, допомагаючи моделі уникнути перенавчання та підвищуючи її здатність до адаптації у різних умовах. Це особливо корисно для моделей, які використовуються в динамічних ринкових умовах, де стабільність та здатність до генералізації є критичними для досягнення точних прогнозів.

5. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для демонстрації запропонованого підходу було використано метод Z-Learner. Він базується на спеціальному перетворенні цільової змінної, що дозволяє трансформувати будь-яку ймовірнісну класифікаційну модель у таку, яка здатна прогнозувати вплив. На відміну від більшості підходів, які передбачають використання двох окремих моделей, Z-Learner застосовує єдину модель, що прямо оцінює вплив, замість розрахунку ймовірностей для тестової та контрольної груп окремо [1]. Це робить його досить зручним у поєднанні з напівкерованим навчанням, оскільки не потребує додаткових модифікацій для врахування ознаки наявності взаємодії. Для застосування алгоритму напівкерованого навчання до мета-учня Z-Learner достатньо у якості вхідних даних подати ознаки користувачів з маркованих та немаркованих даних, а у якості вихідних – результат z-перетворення, який розраховується за наступною формулою. Нехай, $X \in \mathbb{R}^{n \times k}$ – значення k ознак, які описують n клієнтів; $W \in \mathbb{R}^{n \times 1}$ – ознака взаємодії, де $w_i = 1$ означає що проводилась взаємодія з i-им користувачем, інакше 0; $Y \in \mathbb{R}^{n \times 1}$ – ознака виконання цільової дії, де $y_i = 1$ означає що i-ий користувач виконав цільову дію, інакше 0. Тоді Y трансформується в Z за формулою:

$$Z = \begin{cases} 1 & \text{якщо } (W = 1 \text{ та } Y = 1) \text{ або } (W = 0 \text{ та } Y = 0); \\ 0 & \text{інакше.} \end{cases}$$

У якості напівкерованого методу навчання було обрано метод RESSEL. Цей метод забезпечує різноманітність, яка необхідна для ансамблю, за допомогою напівкерованого навчання, тоді як ансамбль надає напівкерованій частині стійкість і використовує механізм

ранньої зупинки для кращих результатів. Запропонований підхід застосовано на даних про клієнтів, де для тренування використано 60 тисяч клієнтів, 20% з яких мали марковані дані. Додатково для порівняння побудовано модель на основі 100% маркованих тренувальних даних. Для тестування моделей було виділено окремо 15 тисяч клієнтів. Якість результатів методу Z-Learner з керованим та напівкерованим навчанням наведено на рисунках 2, 3 та в таблиці 1.

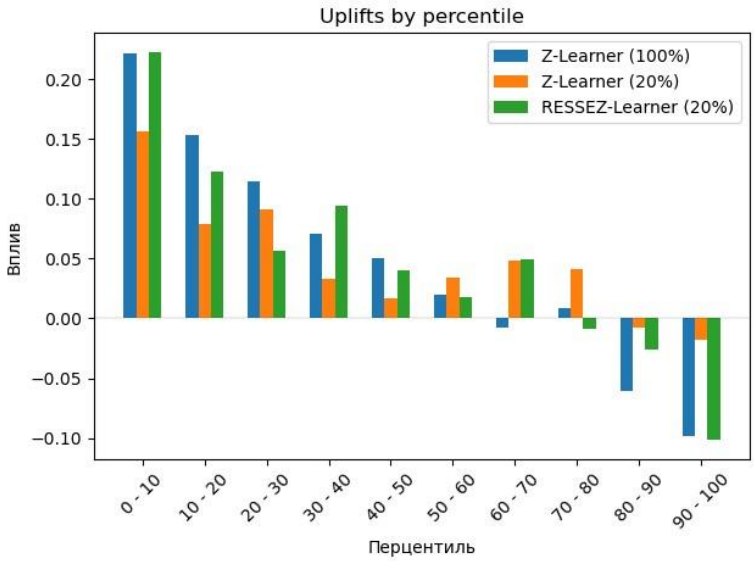


Рисунок 2. Стовпчикова діаграма впливу за перцентильними рангами

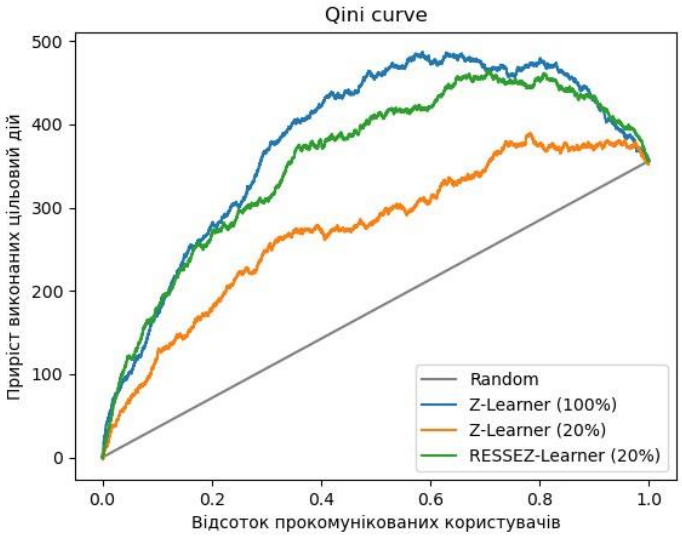


Рисунок 3. Графік кривої Квіні

Таблиця 1. Значення числових критеріїв якості натренованих моделей Z-Learner

Тип навчання та % маркованих даних	Вплив на топ 15%	Вплив на топ 30%	Коефіцієнт Квіні
Кероване навчання, 20% маркованих даних	12,82%	10,82%	6,79%
Напівкероване навчання, 20% маркованих даних	19,35%	13,57%	12,63%
Кероване навчання, 100% маркованих даних	21,46%	16,31%	14,26%

Використання напівкерованого підходу дозволило суттєво покращити значення метрик вплив на топ 15%, вплив на топ 30% та Коефіцієнт Квіні на 6.53%, 3.55% та 5.84% відповідно. Судячи з рисунків 2 та 3, застосування напівкерованого підходу на 20% макрованих даних дозволило отримати якість результатів близьку до моделі, натренованої на 100% маркованих даних керованим навчанням. Проте на обох рисунках помітно, що модель натренована на 100% маркованих даних більщ стабільна в пріоритизації користувачів, що очікувано. Це свідчить про значний потенціал напівкерованого навчання у застосуванні для моделювання впливу.

6. ВИСНОВКИ

У роботі розглянуто основні проблеми та виклики, пов'язані з моделюванням впливу в електронній комерції, зокрема обмеженість у ресурсах для взаємодії з клієнтами та загальній кількості доступних споживачів. Окремо зазначено недоліки класичних методів, які не враховують даних контрольних груп, що може призводити до менш точних прогнозів та нераціонального використання маркетингових ресурсів. Було розглянуто постановку задачі моделювання впливу в контексті електронної комерції та групи, на які поділяються користувачі, що підкреслило перевагу даного підходу над звичайними класифікаційними моделями. Також у роботі сформульовано основні виклики моделювання впливу в електронній комерції, зосередивши увагу на головному – обмеженості ресурсів і клієнтів.

У результаті було запропоновано застосування напівкерованого навчання в моделюванні впливу для вирішення проблеми обмеженості та наведено аргументи у вигляді переваг такого підходу. Напівкероване навчання у моделюванні впливу дозволяє ефективно використовувати немарковані дані, знижуючи витрати на їх розмітку та підвищуючи точність моделі за умов обмежених ресурсів. Такий підхід покращує генералізацію і стабільність моделі, роблячи її більш адаптивною до змін у поведінці користувачів, що важливо для динамічної електронної комерції. Крім того, цей підхід зменшує кількість клієнтів, на яких випадковий експеримент для збору даних може непередбачувано вплинути, що підвищує ефективність моделі впливу.

Перевірка запропонованого підходу на мета-учні Z-Learner з напівкерованим методом навчання RESSEL показала, що дана ідея має потенціал для практичного застосування. В майбутніх роботах планується детальніше розглянути запропонований підхід і дослідити його на інших видах моделей впливу.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Jaroszewicz, Szymon. Uplift modeling for clinical trial data. In: ICML Workshop on Clinical Data Analysis. 2012. p. 79-95.

2. Devriendt, Floris; Moldovan, Darie; Verbeke, Wouter. A literature survey and experimental evaluation of the state-of-the-art in uplift modeling: A stepping stone toward the development of prescriptive analytics. Big data, 2018, 6.1: 13-41.

3. Gubela R. et al. Conversion uplift in e-commerce: A systematic benchmark of modeling strategies. International Journal of Information Technology & Decision Making, 2019, 18.03, P. 747-791.

4. Zhang, Xin, et al. "Temporal Uplift Modeling for Online Marketing." Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2024. p. 6247-6256.

5. Wang, D., Xu, Q., Feng, Y., Ignatius, J., Yin, Y., & Xiao, D. (2024). Uplift modeling and its implications for appointment date prediction in attend home delivery. Decision Support Systems, 114303.

6. Betlei, A., Diemert, E., & Amini, M. R. (2018). Uplift prediction with dependent feature representation in imbalanced treatment and control conditions. In Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13–16, 2018, Proceedings, Part V 25 (pp. 47-57). Springer International Publishing.

7. Chen, X., Liu, Z., Yu, L., Yao, L., Zhang, W., Dong, Y., ... & Gu, J. (2022). Imbalance-aware uplift modeling for observational data. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 36, No. 6, pp. 6313-6321).

8. Методи та технології напівкерованого навчання: курс лекцій [Електронний ресурс]: навч. посіб. для здобувачів ступеня магістра за освітньою програмою «Системи і методи штучного інтелекту» спеціальності 122 «Комп'ютерні науки» / КПІ ім. Ігоря Сікорського; уклад.: Віктор Синєглазов, Олена Чумаченко. – Київ: КПІ ім. Ігоря Сікорського, 2022. – 90с.

9. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., & Raffel, C. A. (2019). Mixmatch: A holistic approach to semi-supervised learning. Advances in neural information processing systems, 32.

10. Lerner, B., Shiran, G., & Weinshall, D. (2020). Boosting the performance of semi-supervised learning with unsupervised clustering. arXiv preprint arXiv:2012.00504.

ВИЯВЛЕННЯ ШАХРАЙСЬКОЇ ПЛАТІЖНОЇ ДІЯЛЬНОСТІ НА ОСНОВІ МЕТОДІВ ШТУЧНОГО ІНТЕЛЕКТУ

Пишнюк А.О.[1], Кузнєцова Н.В.

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] a.pyshniuk@gmail.com

Метою роботи є дослідження ефективних методів боротьби та запобігання платіжному шахрайству в мережі інтернет. На практиці перевірити і спробувати покращити сучасні підходи зі застосуванням моделей машинного навчання. Важливим напрямком дослідження є робота зі вхідними даними, пошук нестандартних взаємозв'язків між ознакам поведінки користувачів і ймовірністю шахрайства для побудови ефективної системи. Досліджені у роботі моделі було порівняно між собою і обрано найкращу.

Ключові слова: онлайн платежі, шахрайство в інтернеті, штучний інтелект, методі і моделі машинного навчання, класифікація

1. ВСТУП

З кожним днем шахрайська діяльність у сфері платежів становить дедалі більше загрози для економіки та фінансової безпеки не тільки окремих компаній, приватних установ, але й цілих країн. Зловмисники постійно шукають нові методи обману та незаконного збагачення через платіжні системи та персональні дані користувачів інтернету. Тому питання розробки ефективних методів виявлення шахрайських дій стає вкрай актуальним для банків, фінансових установ, провайдерів платіжних послуг та приватних компаній у всьому світі [1].

Методи аналізу даних, машинного навчання та штучного інтелекту стали ключовими інструментами в боротьбі з платіжним шахрайством. Саме використання штучного інтелекту дозволяє автоматизувати процес виявлення підозрілих транзакцій, визначаючи відхилення від звичайних шаблонів поведінки користувачів при здійсненні оплат в інтернеті. Застосування таких методів та моделей допомагає створити системи, які здатні надійно виявляти шахрайські схеми та швидко реагувати на нові загрози.

Основною метою даної роботи є створення ефективної моделі машинного навчання, яка дозволяє передбачити ймовірність того, що певна платіжна транзакція є шахрайською. Для цього модель аналізує дані користувача, його історію платежів, дії на інтернет-ресурсах та інформацію щодо поточної транзакції, щоб визначити ймовірність шахрайства.

Актуальність дослідження полягає в тому, що побудова і вдосконалення подібних моделей допоможе фінансовим установам, державним та приватним організаціям ефективніше запобігати платіжному шахрайству, тим самим зменшуючи свої операційні та репутаційні збитки. Навіть невеликі покращення точності, швидкодії, інтерпретованості подібних систем можуть нести за собою великий монетарний вплив на економіку різних підприємств.

2. ПІДХОДИ ДО ВИРІШЕННЯ ПОСТАВЛЕНОЇ ПРОБЛЕМИ

Сьогодні розроблено чимало різних методів для виявлення шахрайських платіжних операцій. Вони варіюються від простих підходів, які не вимагають складних математичних розрахунків і мають низьку точність та погану масштабованість, до складних рішень, які мають як технічні, так і математичні труднощі в реалізації. Основними підходами є такі:

1. Системи на основі правил. Дані системи працюють шляхом встановлення заздалегідь визначених критеріїв або порогових значень для позначення підозрілих транзакцій. Вони базуються на параметрах, таких як суми, швидкість виконання операцій, географічне положення тощо. Хоча ці системи прості у впровадженні, їхня здатність виявляти складні шахрайські схеми досить обмежена.
2. Машинне навчання з учителем. Даний підхід використовує розмічені набори даних, де кожна транзакція (або користувач, який здійснює транзакцію) класифікована як законна або шахрайська. Модель навчається на цих даних і може прогнозувати ймовірність шахрайства для нових транзакцій. Серед популярних алгоритмів цього типу є логістична регресія, дерева рішень, ансамблеві рішення.
3. Машинне навчання без учителя. Подібний підхід використовується для ситуацій, коли немає розмічених даних. Такі моделі аналізують транзакції на наявність аномалій, не використовуючи попередньо визначені категорії. Алгоритми кластеризації, наприклад, такі як K-Means і DBSCAN, групують подібні транзакції та виявляють незвичні, підозрілі шаблони поведінки, які за певними ознаками відрізняються від норми.
4. Гібридні методи. Такі методи можуть комбінувати переваги декількох різних підходів і систем. Наприклад, на основі правил і алгоритмів машинного навчання. Вони використовують заздалегідь визначені правила для виявлення відомих шахрайських схем, а також машинне навчання для виявлення нових та більш витончених шахрайських патернів.
5. Методи графічного аналізу. Даний підхід дозволяє виявляти шахрайство, аналізуючи взаємозв'язки між учасниками платіжних операцій. Графічне представлення даних допомагає виявити підозрілі зв'язки, такі як мережі шахрайських рахунків, або шахрайських акаунтів у мережі інтернет. Це особливо корисно для виявлення організованих шахрайських схем.
6. Аналіз тексту із застосуванням моделей обробки природної мови. Використовується для аналізу неструктурованих даних, таких як описи транзакцій або клієнтські відгуки. Ці методи скоріше виступають допоміжними до основних і добре працюють в комбінації з іншими, допомагаючи виявляти шахрайські схеми через аналіз текстового вмісту.

Вибір підходу визначається багатьма чинниками, наприклад, такими як толерантність до ризиків, доступні дані, обчислювальні ресурси та здатність систем адаптуватися до нових шахрайських схем [2]. Ефективність кожного з цих методів в першу чергу залежить від доступності і якості даних.

Зазвичай зіштовхуючись з подібною проблемою компанії та установи в першу чергу пробують вирішити її більш простими методами, часто навіть без застосування машинного навчання, оскільки це може бути для них операційно неефективно. Лише після отримання певної базової системи для визначення платіжного шахрайства та оцінки її якості, компанії починають досліджувати і впроваджувати складніші підходи, моделі.

3. ОЗНАКИ ШАХРАЙСЬКОЇ ПЛАТІЖНОЇ ДІЯЛЬНОСТІ

Кожна система і модель штучного інтелекту завжди починається з даних. Якщо порівнювати систему, яка використовує штучний інтелект, з людським організмом, то, можна сказати, що дані є кровоносною системою цього організму. Від якості, кількості даних, їх точності розмітки (якщо дані розмічені) залежить дуже багато. Який би складний алгоритм штучного інтелекту ви не використовували для своєї задачі, якщо дані некоректні, містять багато пропущених значень, не консистентні, на жаль, модель ніколи не вийде якісною. І навпаки, якщо дані повні, містять максимально якісну розмітку, гарно оброблені і підготовлені, навіть на найпростіших алгоритмах можна отримати доволі непогані результати.

В поточному досліджені будуть використовуватися дані зібрані з реальних джерел – декількох інтернет-сайтів соціальних мереж. Основний вид шахрайства на подібних сайтах складають карткові шахраї, які використовують їх для перевірки наявності коштів на викрадених картках і їх зняття. Виходячи з розуміння «що таке платіжне шахрайство» було сформовано перелік гіпотез щодо того, які ознаки можна використовувати у моделі. Вони будуть розділені на такі категорії:

1. Інформація щодо поточної транзакції користувача. Сюди можна віднести абсолютно всі ознаки, які формуються з даних про транзакцію з конкретним індикатором. Наприклад, такі як, час транзакції, ім’я користувача на картці, країна випуску картки, банк, платіжна система, коди помилок, якщо вони є, даної транзакції на стороні банку.
2. Інформація щодо попередніх транзакцій користувача. Наприклад, кількість успішних/неуспішних транзакцій, час між транзакціями, кількість різних карток, які використовував користувач раніше тощо.
3. Аудиторні показники користувача. Вік, регіон, країна, джерело трафіку, з якого прийшов користувач, девайс, браузер користувача тощо.
4. Взаємодія користувача з сайтом. Час використання сервісів на сайті, взаємодія з іншими користувачами, використання пошукової системи тощо.
5. Мультиакаунтність. Враховуючи, що одна людина може мати багато різних акаунтів на одному чи на різних сайтах, важливо також агрегувати інформацію і на рівні метакористувача, оскільки інформація з попередніх реєстрацій може бути дуже корисною при виявленні шахрайства на поточній.
6. Інформація щодо картки. Викрадені картки можуть потрапити в руки різних шахраїв, і вони можуть спробувати використовувати їх на однакових сайтах. Саме тому агрегування на рівні платіжної картки також може нести багато користі.

Всі ці ознаки можуть бути дуже корисними при виявленні різного роду шахрайства, але лише за умови їх правильного збору.

4. МОДЕЛІ КЛАСИФІКАЦІЇ І КРИТЕРІЇ ЯКОСТІ

Задачу виявлення шахрайських транзакцій будемо вирішувати за допомогою моделей класифікації. Алгоритми класифікації є важливим інструментом для аналізу статистичних проблем. У машинному навчанні або статистиці класифікація називається проблемою визначення того, чи належить об’єкт до певної категорії на основі попередньо навченої моделі. Такі моделі навчаються статистично на основі набору навчальних даних, категорії яких визначені заздалегідь. Опишемо основні методи класифікації, які були застосовані у даному дослідженні, а також методи для порівняння точності таких моделей.

4.1. Моделі машинного навчання для класифікації даних

Логістичну регресію іноді називають логістичною моделлю або logit-моделлю. Дана модель аналізує зв'язок між кількома незалежними змінними та категоріальною залежною змінною та оцінює ймовірність настання події шляхом підгонки даних до логістичної кривої. Існує дві моделі логістичної регресії, бінарна та мультиноміальна логістичної регресії. Бінарна логістична регресія зазвичай використовується, коли залежна змінна є дихотомічною, а незалежні змінні можуть бути як неперервними, так і категоріальними. Якщо залежна змінна не є дихотомічною і складається з більш ніж двох категорій регресія є мультиноміальною [3]. Сигмоїдна або логістична функція визначає дану модель. Дану функцію часто використовують для моделювання популяції населення. Графічне зображення сигмоїдної функції представлено на рисунку 1.

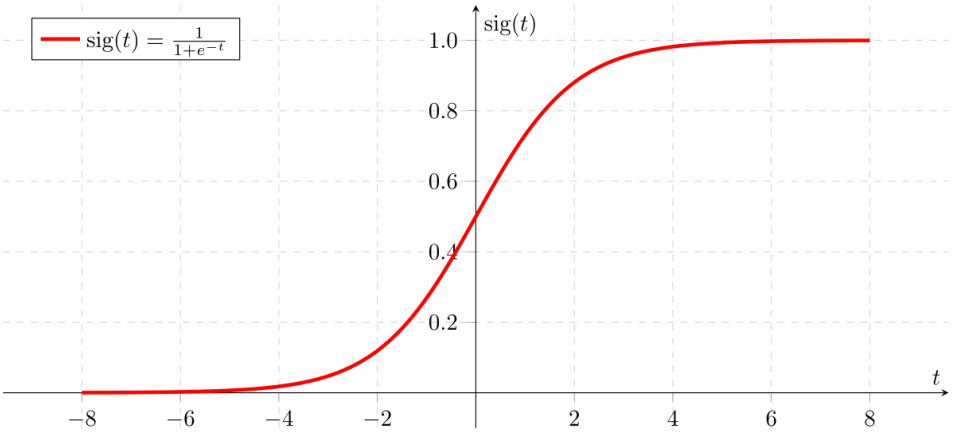


Рисунок 1. Сигмоїдна функція

Формально дану модель можна подати у вигляді наступного математичного рівняння:

$$y = g(W^T X + b)$$

де g – сигмоїдна функція, W – матриця ваг, X – матриця значень вхідних ознак, b – вектор значень вільних членів.

Дерева рішень – це популярний алгоритм машинного навчання, який використовується як для завдань класифікації, так і для регресії. Даний алгоритм фактично є рекурсивним розділенням простору елементів. Дерево рішень складається з вузлів, які утворюють кореневе дерево, тобто це спрямоване дерево з вузлом під назвою «корінь», який не має вхідних ребер. Усі інші вузли мають рівно одне вхідне ребро. Вузол із вихідними краями називається внутрішнім або тестовим вузлом. Усі інші вузли називаються листами (також відомі як кінцеві вузли або вузли рішень).

XGBoost (Extreme Gradient Boosting, екстремальний градієнтний бустинг) – це оптимізований та високоефективний алгоритм машинного навчання, в основу якого покладено дерева рішень та фреймворк градієнтного бустингу. Даний алгоритм має видатну продуктивність та гнучкість, в порівнянні з іншими класичними алгоритмами машинного навчання, в обробці різноманітних задач машинного навчання. XGBoost є ітеративним алгоритмом із кількома деревами. Кожне дерево вчиться на залишках усіх попередніх дерев. На відміну від алгоритму випадкового лісу, прогнозований результат XGBoost – сума прогнозів всіх дерев [4].

$$\hat{y}_i = \sum_{k=1}^{n} f_k(x_i)$$

де $f_k(x_i)$ – результат k-того дерева для прикладу i.

Архітектуру алгоритму XGBoost ілюстративно представлено на рисунку 2.

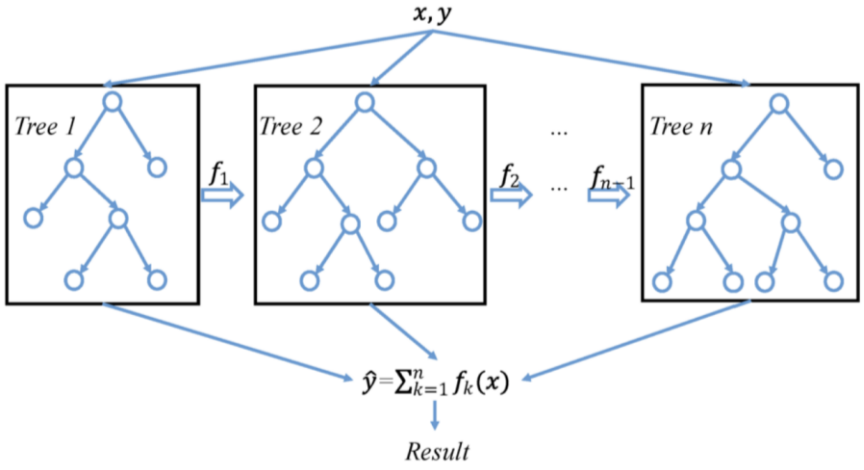


Рисунок 2. Архітектура XGBoost

4.2. Методи оцінки якості побудованих моделей класифікації

Жодна модель у світі не може буде практично застосована, якщо немає чіткого розуміння якості її роботи як на тренувальних, так і на тестових даних, яких модель ще не бачила. Тому після розгляду різних найпопулярніших моделей класичного машинного навчання для вирішення проблем класифікації перейдемо до методів оцінки їх якості.

Враховуючи, що у дослідженні було використано незбалансований набір даних (оскільки шахрайський транзакцій значно менше ніж звичайних), для оцінки якості розроблених моделей і їх порівняння між собою оберемо наступні метрики:

1. Precision – частка правильно спрогнозованих об’єктів позитивного класу серед усіх спрогнозованих об’єктів позитивного класу.

$$precision = \frac{TP}{TP + FP}.$$

2. Recall – частка знайдених моделлю об’єктів позитивного класу серед усіх об’єктів позитивного класу. Іншими словами, покриття моделлю позитивного класу.

$$recall = \frac{TP}{TP + FN}.$$

3. F1-score, яка є середнім гармонічне precision та recall.

$$f1 - score = (1 + \beta^2)\frac{precision * recall}{\beta^2 * precision + recall}.$$

де β – вага precision. Зазвичай β покладають рівним 1.

4. AUC ROC – площа під кривою ROC – метрика оцінки якості моделі класифікації у випадку бінарної класифікації. Вона оцінює здатність моделі розрізняти позитивний та негативний класи шляхом вимірювання компромісу між TPR – True Positive Rate і FPR – False Positive Rate.

$$TPR = \frac{TP}{TP + FN}, FPR = \frac{FP}{TN + FP}.$$

5. РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ

Використовуючи описані вище, і ще деякі додаткові, алгоритми, було побудовано декілька моделей виявлення шахрайських транзакцій та порівняні за метриками якості, описаними вище. Порівняння якості моделей наведено у таблиці 1.

Таблиця 1. Порівняння реалізованих алгоритмів класифікації

Алгоритм	Precision	Recall	F1-score	AUC
Logistic Regression	0.734	0.594	0.656	0.881
Decision Tree	0.745	0.691	0.717	0.912
Random Forest	0.778	0.671	0.72	0.917
XGBoost	0.804	0.682	0.738	0.93
SVM	0.751	0.647	0.695	0.889

Як бачимо з отриманих значень метрик, алгоритм XGBoost є найкращим серед усіх інших алгоритмів як по значенню метрики f1-score, так і по значенню метрики AUC (площі під ROC-кривою). Це і не дивно, оскільки на даний момент XGBoost вважається state-of-the-art серед класичних алгоритмів машинного навчання. Також, наведемо графіки отриманої ROC-кривої для моделі XGBoost на рисунку 3.

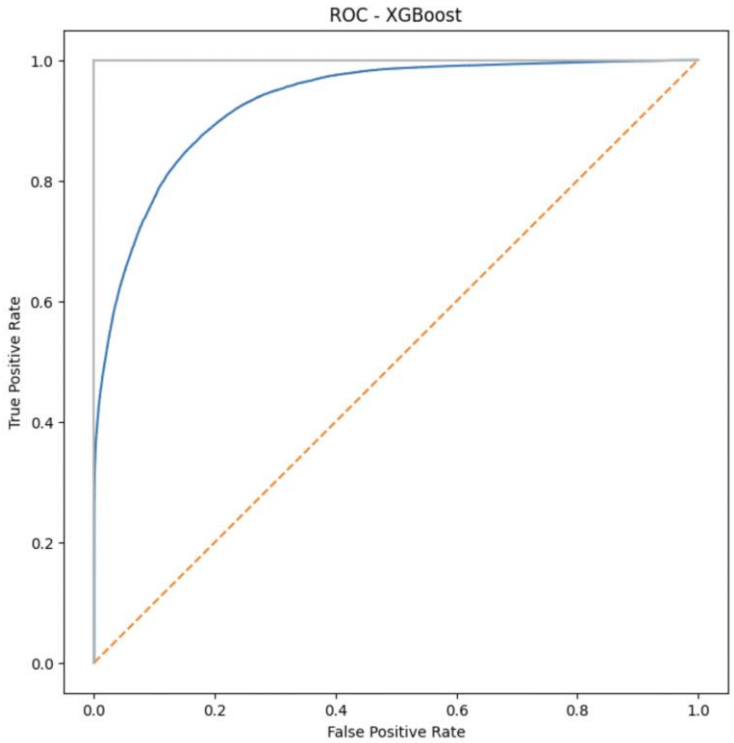


Рисунок 3. ROC-крива для XGBoost

6. ВИСНОВКИ

У проведеному дослідженні було розглянуто задачу виявлення платіжного шахрайства в інтернеті, проведено аналіз ознак платіжного шахрайства, побудовано моделі класифікації, виконано їх порівняння між собою за допомогою різних метрик. У якості вхідних ознак в моделі було використано датасет з реальних онлайн-сервісів з великою кількістю даних про транзакції. Було зібрано багато нестандартних характеристик, які допомогли описати різні

патерни шахрайства і значно покращити фінальну модель. Серед побудованих моделей було обрано найкращу, якою виявилася модель, основана на алгоритмі XGBoost. Дана модель має показники f1-score на рівні 0,74 та AUC на рівні 0,93, що є високими показниками якості. Модель була побудована за допомогою мови Python та сучасних бібліотек машинного навчання, таких як scikit-learn, xgboost, pandas, тому її вже можна легко інтегрувати у системи різних підприємств та установ для своєчасного запобігання онлайн-шахрайству. У подальших дослідженнях для покращення моделі варто в першу чергу заглибитися в якість розмітки, на якій вона вчиться. Оскільки розмітку робили реальні люди, вона може містити неточності, зміщення. За допомогою різних методів можна додавати в модель показники якості роботи кожної людини, яка ці дані розмічає, і таким чином зменшувати вплив людського фактору на розмітку.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Fraud Detection with Machine Learning: Identifying Suspicious Patterns in Financial Transactions. URL: https://medium.com/@zhonghong9998/fraud-detection-with-machine-learning-identifying-suspicious-patterns-in-financial-transactions-8558f3f1e22a (дата звернення: 06.11.2024)

2. Fraud Detection and Security: An Overview of Methods and Technologies. URL: https://medium.com/@derya.cortuk/fraud-detection-and-security-an-overview-of-methods-and-technologies-4fcf94fa1e2b (дата звернення: 08.11.2024)

3. Hyeoun-Ae P. An Introduction to Logistic Regression: From Basic Concepts to Interpretation with Particular Attention to Nursing Domain. J Korean Acad Nurs. 2013. Vol.43 №.2. P. 154-164. DOI: 10.4040/jkan.2013.43.2.154.

4. Cutler A., Cutler R.D., Stevens R.J. Ensemble Machine Learning: Methods and Applications. 2011. P.157-176. DOI:10.1007/978-1-4419-9326-7_5.

ВИКОРИСТАННЯ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ У ТРАНСПОРТНІЙ ГАЛУЗІ НА ОСНОВІ МЕТОДІВ ШТУЧНОГО ІНТЕЛЕКТУ

Поліщук М.С.[1], Гуськова В.Г.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] maksym.polischuk.th@gmail.com, [2] guskovavera2009@gmail.com

У дослідженні розглянуто різні методи рекомендацій для транспортних засобів із застосуванням штучного інтелекту, з внесенням модифікацій для їх вдосконалення. Оцінено актуальність завдання, сучасний стан галузі, існуючі підходи та алгоритми. Визначено переваги і недоліки методів, що дозволило обрати оптимальний підхід для застосування в рекомендаціях транспорту з акцентом на інноваційні методи для підвищення ефективності. Метою дослідження є розробка рекомендаційної системи, яка використовує методи машинного навчання та аналізу даних для надання персоналізованих рекомендацій на платформах електронної комерції. У роботі буде розглянуто різні підходи, такі як колаборативна фільтрація, контент-орієнтовані методи та гібридні моделі, а також вивчено вплив алгоритмів штучного інтелекту на покращення якості рекомендацій.

Ключові слова: рекомендаційні системи, транспортні засоби, штучний інтелект, алгоритми

1. ВСТУП

У сьогоднішньому цифровому середовищі, де обсяги інформації постійно зростають, уміння ефективно обробляти та аналізувати дані стає надзвичайно важливим для успіху бізнесу, особливо в електронній комерції. Рекомендації та поради завжди відігравали значну роль у прийнятті рішень людьми. Якщо раніше це було зосереджено на досвіді інших, то сьогодні на перший план виходять автоматизовані системи, здатні виконувати ті ж функції, іноді навіть краще за людей. Завдяки розвитку технологій штучного інтелекту та машинного навчання з'явилася можливість створення інтелектуальних рекомендаційних систем, які аналізують поведінку користувачів, їхні уподобання та потреби, пропонуючи відповідні товари та послуги.

Рекомендаційні системи допомагають орієнтуватися у складному інформаційному просторі, виділяючи важливі дані та пропонуючи користувачам те, що може відповідати їхнім інтересам. Це підвищує ефективність пошуку та вибору продуктів або контенту, а також сприяє формуванню більш індивідуального та змістовного досвіду взаємодії з технологіями. Рекомендації можуть стимулювати відкриття нових інтересів, допомагаючи користувачам вийти за межі звичного й знайти щось нове та надихаюче. Вони відіграють важливу роль у формуванні адаптивного інформаційного середовища, що враховує індивідуальні особливості кожного користувача та забезпечує більш персоналізоване спілкування з цифровим світом.

Такі системи не лише впливають на економічні показники та бізнес-процеси, але й глибоко змінюють культуру споживання та взаємодії людей з технологіями. Вони стають каталізаторами змін у способах вибору, сприйняття та взаємодії з інформаційним

середовищем, формуючи нові моделі поведінки та споживання. Тому розуміння та оптимізація рекомендаційних систем набувають особливого значення, адже вони впливають не лише на економічну ефективність, але й на якість життя та добробут людей.

Ця дисертація присвячена розробці та впровадженню рекомендаційної системи для проекту в сфері електронної комерції, з використанням сучасних підходів у штучному інтелекті. Незважаючи на значний прогрес у цій області, створення та оптимізація рекомендаційних систем залишаються складними завданнями через необхідність врахування багатьох факторів, таких як точність, релевантність і персоналізація рекомендацій. Додатковою складністю є потреба в забезпеченні масштабованості та високої продуктивності системи, щоб вона могла ефективно працювати з великими обсягами даних та великою кількістю користувачів.

З урахуванням вищевикладеного, це дослідження має на меті не лише розробити та впровадити ефективну рекомендаційну систему, але й дослідити вплив різних факторів на її продуктивність, а також надати рекомендації щодо оптимізації системи під конкретні потреби проекту електронної комерції. Отже, результати цієї роботи можуть стати важливим внеском у розвиток технологій рекомендаційних систем та запропонувати практичні рішення для їх реалізації в реальних бізнес-процесах.

2. АНАЛІЗ ОСОБЛИВОСТЕЙ ПОБУДОВИ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

2.1. Опис існуючих методів, що використовуються у персоналізації

Неперсоналізовані рекомендації: Ці системи пропонують однакові товари для всіх користувачів, спираючись на середні оцінки чи загальні відгуки. Це не враховує індивідуальних уподобань і схоже на показ товарів у фізичних магазинах або на платформах.

Колаборативна фільтрація: Цей метод прогнозує вподобання користувачів на основі оцінок інших. Він створює матрицю оцінок, де аналізуються схожості між користувачами або товарами. Популярним є метод латентних факторів (SVD), що дозволяє прогнозувати оцінки навіть для нових пар "користувач-об’єкт".

Контентна фільтрація: Орієнтується на аналіз характеристик продуктів і профілю користувача, зокрема на текстових описах або відгуках. Метод TF-IDF оцінює важливість слів у тексті для надання рекомендацій, які найбільше відповідають вподобанням користувача. Цей метод корисний для нових користувачів чи товарів.

Гібридні рекомендаційні системи: Поєднують кілька методів для точніших результатів. Наприклад, комбінуються результати колаборативної та контентної фільтрації за допомогою вагових коефіцієнтів або вибору найбільш підходящої моделі залежно від ситуації. Гібридні моделі дозволяють комбінувати дані з різних підходів для покращення персоналізації та точності рекомендацій.

2.2. Рекомендаційні системи з використанням нейронних мереж

Рекомендаційні системи з нейронними мережами: Нейронні мережі використовуються для розробки рекомендаційних систем завдяки здатності виявляти складні залежності в великих обсягах даних. Глибоке навчання дозволяє ефективно аналізувати характеристики користувачів і продуктів, формуючи точні рекомендації. Зокрема, конволюційні нейронні мережі (CNN) обробляють візуальні дані, а рекурентні нейронні мережі (RNN) аналізують текст, що допомагає зрозуміти переваги користувачів. Автоенкодери зменшують розмірність даних, виявляючи приховані фактори.

Навчання нейронної мережі: Навчання базується на алгоритмах оптимізації, зокрема стохастичному градієнтному спуску (SGD), що використовується для корекції ваг мережі. Для

пришвидшення процесу використовуються розширення SGD, такі як імпульс та методи Adagrad, RMSProp, і Adam, останній з яких поєднує переваги попередніх та ефективно адаптує швидкість навчання. Функція активації ReLU і її модифікації, зокрема Leaky ReLU, забезпечують ефективність в великих мережах. Для ініціалізації ваг використовують методи Xavier та He.

Обчислювальні ресурси: Використання графічних процесорів (GPU) значно прискорює навчання великих моделей завдяки паралельній обробці даних.

Методи оптимізації: Метод Adam поєднує адаптивну зміну швидкості навчання і ефективну обробку градієнтів, що робить його ідеальним для великих моделей. Його застосування вимагає менше ітерацій для стабільного навчання порівняно з методом SGD. Вибір оптимізаційного алгоритму залежить від задачі, складності моделі та доступних ресурсів.

2.3. Моделі, що базуються на векторизації

У сучасному цифровому середовищі, де щоденно з'являються десятки мільйонів товарів, важливо ефективно виявляти та порівнювати подібні пропозиції на сайті. Це дозволяє об'єднувати товари від різних продавців в одну картку товару, полегшуючи користувачам пошук та вибір. Кожен продукт містить різноманітні дані, такі як зображення, назва, опис та додаткові атрибути. Основна мета полягає у зборі та обробці цієї інформації для виконання завдань пошуку відповідностей між товарами.

Для вилучення характеристик із продуктів використовуються векторні представлення (ембеддинги). Це досягається за допомогою різних текстових моделей, таких як fastText і transformers, для обробки описів та назв товарів. Для аналізу зображень застосовуються конволюційні нейронні мережі (CNN), наприклад, ResNet, EffNet та NFNet. Отримані вектори використовуються для генерації характеристик та пошуку відповідностей між товарами, створюючи векторний простір, в якому подібні товари мають схожі векторні представлення.

Prod2Vec: Концепція та реалізація

Prod2Vec не є універсальним алгоритмом, а радше концепцією, яка адаптується для конкретних завдань. Основна ідея полягає у навчанні векторних представлень продуктів на основі послідовності переглядів або покупок користувачів. Цей підхід запозичує терміни з області обробки природної мови, трактуючи послідовність покупок як "речення", а продукти в цій послідовності — як "слова". Модель використовує метод skip-gram для максимізації ймовірності появи сусідніх продуктів у контексті поточного продукту. Prod2Vec моделює контекст послідовності продуктів, що дозволяє продуктам з подібними контекстами отримувати схожі векторні представлення. Проте оригінальна модель не враховує одночасні покупки кількох продуктів, що було виправлено в модифікаціях, таких як bagged-prod2vec.

Навчання моделей та оптимізація

Навчання моделей Prod2Vec здійснюється за допомогою стохастичного градієнтного підйому (SGD), який є ефективним для завдань великого масштабу. Однак обчислювальна складність градієнтів зростає пропорційно до розміру словника, що робить процес обчислювально витратним при великих наборах даних. Для вирішення цієї проблеми застосовується метод негативного семплінгу, який значно знижує обчислювальну складність, роблячи процес навчання більш ефективним.

2.4. Запропонований алгоритм

Запропонований алгоритм орієнтований на знаходження схожих товарів для користувачів, виходячи з їхньої історії переглядів або покупок. Множина продуктів включає зображення, назви, описи та атрибути, які слугують вхідними даними для алгоритму.

Використовуючи підхід transfer learning, обрана модель поєднує конволюційні нейронні мережі для обробки зображень з моделями обробки природної мови для аналізу текстових даних. Результатом роботи моделі є векторне представлення кожного товару, яке зберігається в векторному просторі.

Для пошуку схожих товарів використовується алгоритм пошуку найближчих сусідів у цьому векторному просторі. Це дозволяє знаходити N найближчих товарів до заданого, що слугують рекомендаціями користувачам. Збереження векторних даних може здійснюватися в різних типах сховищ, таких як реляційні або нереляційні бази даних, або в хмарних сервісах, що забезпечує гнучкість та масштабованість системи.

3. ОГЛЯД ПРОГРАМНОГО ПРОДУКТУ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

3.1. Опис датасету

Ефективність моделі залежить від якості та обсягу даних, на яких вона навчається. Великі датасети покращують різноманітність даних і знижують ризик перенавчання, але потребують значних обчислювальних ресурсів. У цій роботі використано реальний датасет, зібраний через GraphQL API за допомогою Python-скрипта, що забезпечує ефективне завантаження даних з використанням asyncio та aiohttp. Дані зберігаються у форматі HDF5 для економії місця. Текстові дані обробляються за допомогою стандартних методів (стеммінг, видалення стоп-слів) та додавання категорій. Завантаження зображень також оптимізовано за допомогою асинхронних бібліотек. В результаті отримано датасет з понад 200 000 автомобілів, кожен з яких має зображення, назву, атрибути тощо.

3.2. Опис моделі

Проаналізувавши існуючі методи обробки зображень і тексту, було розроблено архітектуру нейронної мережі, яка відображена на схемі (рис. 1).

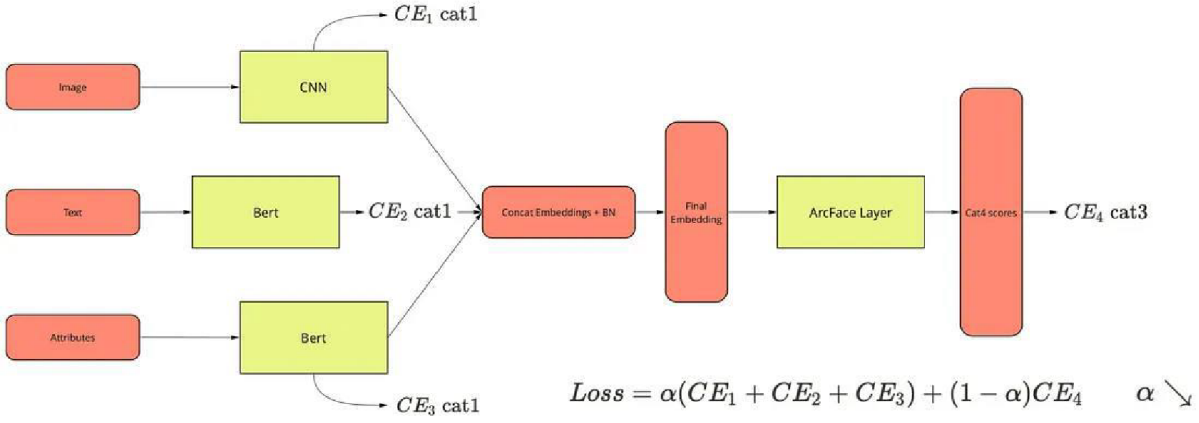


Рисунок 1. Запропонована архітектура нейронної мережі

Перший модуль запропонованої архітектури відповідає за обробку зображень і виконує роль енкодера. Він перетворює вхідне зображення товару на ембединг розміром 312 елементів. Для виконання цієї задачі можуть використовуватись різні нейронні мережі, такі як GoogleNet, MobileNet, EfficientNet, VGGNet, ResNet тощо. У даній роботі обрано ResNet34 як основну архітектуру.

Наступні модулі системи включають дві нейронні мережі для обробки текстових даних. Серед різних алгоритмів NLP було обрано BERT. Ключовою перевагою цього підходу є наявність у відкритому доступі попередньо навчених моделей для різних мов. У нашому

випадку потрібна модель, адаптована для роботи з російською мовою, оскільки всі назви та атрибути товарів спочатку створюються саме цією мовою, а вже потім перекладаються на українську. Щоб уникнути можливих помилок, пов’язаних із перекладом, було вирішено використовувати мову оригіналу.

Наступний компонент моделі – це повнозв’язний шар, який об'єднує вхідні вектори, створюючи єдиний вектор, що слугує представленням продукту. На цьому етапі застосовується метод batch normalization для покращення стабільності та швидкості навчання.

Далі йде функція активації. У рамках експериментів було протестовано кілька варіантів, зокрема Softmax та ArcFace. ArcFace, яка зазвичай використовується у задачах розпізнавання облич, була обрана через схожість концепцій між задачами векторизації продуктів і розпізнавання облич, що дозволяє очікувати суттєвого покращення результатів моделі.

Після цього додається ще один повнозв’язний шар, який використовується для передбачення категорій нижнього рівня (cat3). Цей шар використовується лише під час навчання моделі для обчислення перехресної ентропії.

Після завершення навчання і переходу до тестування модель більше не використовує передбачення cat3. Замість цього нас цікавить лише векторне представлення продукту, тому вихід моделі переміщується до шару з функцією ArcFace.

3.3. Результати тренування

У машинному навчанні тренувальні та тестові вибірки відіграють ключову роль у забезпеченні ефективного навчання та оцінки моделей. Тренувальна вибірка використовується для навчання моделі на відомих даних, тоді як тестова — для оцінки її роботи на нових, невідомих даних. Це дозволяє виявити перенавчання та правильно налаштувати гіперпараметри, забезпечуючи здатність моделі узагальнювати результати для реальних завдань.

Навчання нейронної мережі виконується з використанням батчінгу, де розмір батча становить 32. Для організації процесу застосовуються вбудовані інструменти фреймворку PyTorch: `Dataset` та `Dataloader`. Вони дозволяють ефективно працювати з великими обсягами даних завдяки мультипроцесорності, що забезпечує виконання кількох задач паралельно. Кількість епох навчання була визначена як 10. За цей період модель має змогу достатньо добре вивчити патерни у даних і продемонструвати задовільні результати. Упродовж кожної епохи значення коефіцієнта a у функції втрат поступово зменшується. Це зміщує акцент у навчанні з перших трьох енкодерів на фінальні шари моделі, що дозволяє оптимізувати процес навчання на пізніх етапах.

Початкове значення параметра α встановлюємо як 0,5. Нижче на рис. 2 наведено графік залежності α від епохи.

Результати експериментів підтверджують доволі високу точність розробленої рекомендаційної системи. Модель демонструє здатність ефективно враховувати контекстні взаємодії між транспортом, а саме через категорії, що призводить до покращення якості рекомендацій.

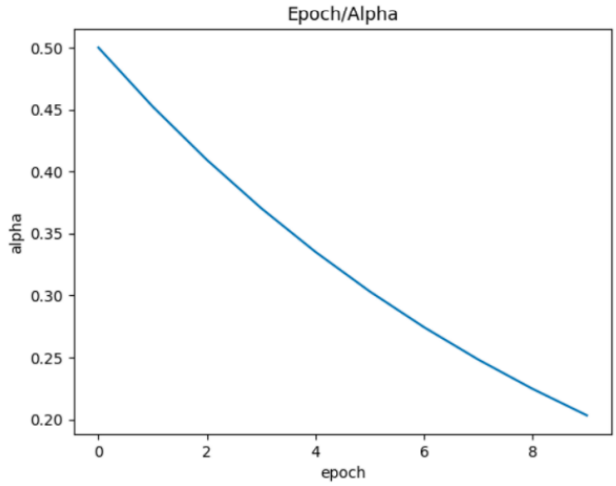


Рисунок 2. Графік залежності альфа від епохи

4. ВИСНОВКИ

У дослідженні проаналізовано методи формування рекомендацій для транспортних засобів із використанням штучного інтелекту та вдосконалено їх шляхом модифікацій. Вивчено сучасні тенденції, алгоритми рекомендацій і їх переваги та недоліки, що дозволило обрати оптимальний підхід для вирішення завдань у сфері рекомендацій транспорту. Для навчання моделі використано дані з автомаркетплейсу, застосовано методи веб-скрапінгу, нейронні мережі та алгоритми пошуку векторів у просторі. Алгоритм продемонстрував високу точність на тестових даних, хоча подальше вдосконалення потребує більше ресурсів і розширення вибірки.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Aggarwal, C. C. (2016). Recommender Systems: The Textbook. Springer.

2. Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. Communications of the ACM, 35(12), 61-70.

3. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web (pp. 173-182).

4. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

МОНІТОРИНГ РОСТУ РОСЛИН НА ОСНОВІ ЗОБРАЖЕНЬ МЕТОДАМИ ГЛИБОКОГО НАВЧАННЯ

Сандрацький В.В.[1], Новогрудська Р.Л.[2], Шаповал Н.В.[3]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] sandrvetal@gmail.com, [2] rinan@ukr.net,
[3] shovgun@gmail.com [0000-0002-8509-6886]

Метою даного дослідження було створити модель здатну класифікувати фази розвитку кукурудзи. Така модель може бути складовою більшої системи моніторингу в сільському господарстві і може слугувати інструментом для прийняття рішення щодо, наприклад, внесення добрив та засобів захисту рослин. Було запропоновано використовувати для класифікації не одне зображення, а кілька, по аналогії як це робить людина. Проведені експерименти показали доцільність такого підходу.

Ключові слова: моніторинг, IoT, нейронні мережі, передавальне навчання.

1. ВСТУП

Моніторинг фаз розвитку рослин є важливим для ефективного управління сільськогосподарським виробництвом, оскільки дозволяє оптимізувати агротехнічні заходи та підвищити врожайність. Традиційні методи моніторингу, такі як візуальні спостереження та ручні вимірювання, часто займають багато часу, потребують високої кваліфікації персоналу та значних зусиль і можуть бути неточними. Сучасні технології дозволяють здійснювати моніторинг більш швидко та точно. Одним з основних інструментів є супутникові знімки, які забезпечують детальне і широкомасштабне спостереження за станом посівів. Крім того, IoT датчики дають змогу збирати точні дані про умови навколишнього середовища, такі як температура, вологість і рівень освітленості. Ці методи дозволяють не лише підвищити ефективність аграрних практик, а й оперативно виявляти проблеми, що виникають під час росту рослин. Інтеграція даних з різних джерел також дає змогу створювати більш точні моделі прогнозування розвитку рослин і приймати обґрунтовані рішення для внесення добрив.

2. ФОРМУЛЮВАННЯ ПРОБЛЕМИ

Одним із поширених методів моніторингу за сільськогосподарськими угіддями є супутникові знімки. Вони є потужним інструментом для моніторингу стану рослин на полях, оскільки вони дозволяють отримувати детальну інформацію про великі території без необхідності фізичної присутності на місці. Супутники можуть також вимірювати індекси здоров'я рослин, такі як NDVI (Normalized Difference Vegetation Index), що дає змогу оцінити рівень вегетації і спрогнозувати потенційну врожайність [1]. Це дає можливість фермерам та агрономам планувати агротехнічні заходи, оптимізувати використання ресурсів і приймати більш точні рішення для підвищення ефективності вирощування.

Не у всіх агрокомпаніях є доступ до супутникових знімків, тому варто розглядати рішення “на місцях”. Моніторинг полів за допомогою датчиків та системи IoT дозволяє фермерам ефективно контролювати умови вирощування рослин і оперативно реагувати на зміни в навколишньому середовищі. Система зазвичай складається з різних датчиків, таких як

сенсори вологості ґрунту, температури і вологості повітря, а також камери для виявлення хвороб рослин. Дані з цих датчиків передаються на платформу IoT, наприклад ThingSpeak, де їх можна переглядати в реальному часі [2]. Використання засобів штучного інтелекту (ШІ) дозволяє точніше виявляти проблеми, наприклад, хвороби рослин, через аналіз зображень, що надходять з камер. Це дає змогу фермерам автоматично регулювати полив, чи допомагає оптимізувати застосування пестицидів, оскільки хвороби можуть бути виявлені на ранніх стадіях, зменшуючи площу та частоту обробки. Все це дозволяє оптимізувати виробничі витрати.

Ще одним прикладом, де необхідно моніторити стан рослин – це розумні (автономні) теплиці. Розумні теплиці використовують датчики для вимірювання температури, вологості, рівня CO2 та освітленості, що дозволяє автоматично регулювати ці параметри для створення ідеальних умов [3]. ШІ використовується для аналізу даних з датчиків та прогнозування потреб рослин, а також для автоматизації таких процесів, як полив та освітлення. ШІ також допомагає також у виявленні хвороб або стресу рослин, на основі даних з камер, що дозволяє оперативно реагувати та застосовувати необхідні заходи.

Дане дослідження зосереджено на створенні системи моніторингу сільськогосподарських полів на основі датчиків, камер та зображень рослин з БПЛА. Дані з системи моніторингу передаються моделі ШІ, яка їх аналізує та передає висновок користувачу, який вже зможе прийняти рішення щодо внесення добрив, поливу, тощо. Запропонована система наведена на рис. 1.

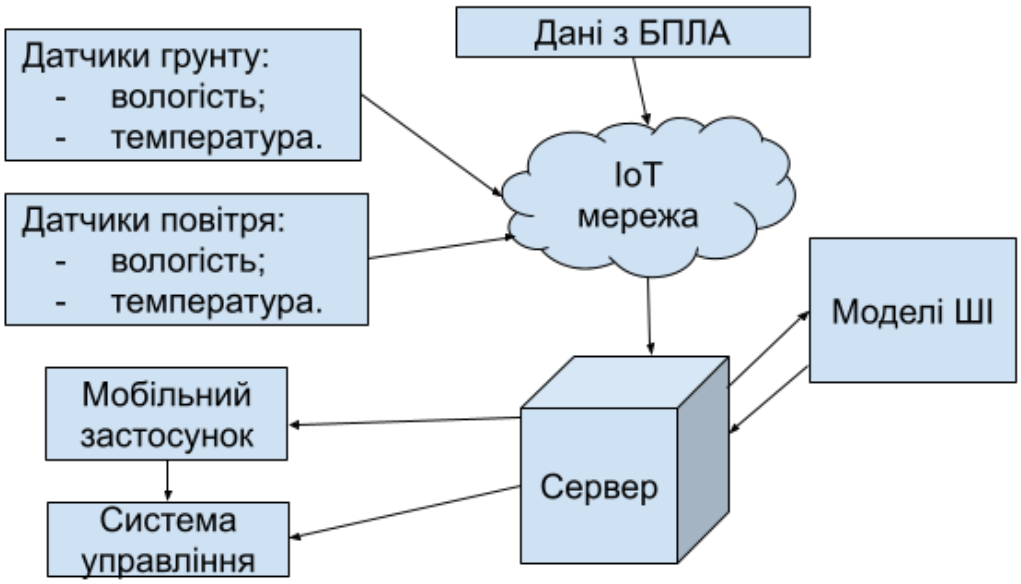


Рисунок 1. Система моніторингу стану сільськогосподарських полів

3. КЛАСИФІКАЦІЯ ФАЗ РОЗВИТКУ РОСЛИН

Для побудови моделі класифікації фази розвитку рослин було використано реальні зображення рослин з полів кукурудзи зібрані в 2023 році. Дані збиралися агрономами, які безпосередньо приїжджали на поле і фотографували рослини. Зображення помічалися класом, який позначав стадію розвитку кукурудзи. Загалом є 12 фаз розвитку. Для розробки моделі було обрано лише 7 фаз розвитку, для яких потрібно або застосовувати добрива або приймати рішення щодо врожаю. Зібрані зображення суттєво відрізнялися в рамках однієї фази розвитку, оскільки не було усталеної методики щодо збору зображень. На одну фазу могло

бути зроблено від трьох до семи зображень, освітленість та частини рослини, які фотографувалися також відрізняються від екземпляра до екземпляра, і все це в рамках однієї фази розвитку. Приклади зображень для однієї фази наведені на рис. 2.

Рисунок 2. Одна й та сама фаза розвитку рослин. Зібрані зображення з двох полів

В рамках дослідження розв'язувалися 2 задачі для яких було обрано два датасети. Задача 1 – класифікація 7 фаз розвитку кукурудзи, кількість екземплярів на клас 40–50. Загальна кількість зображень 1560. Задача 2 – класифікація двох сусідніх фаз (одна переходить в іншу, зображення візуально схожі), по 90 екземплярів на клас. Загальна кількість зображень 1150. Для створення моделі використовувалося передавальне навчання. В якості переднавченої мережі використовувалась мережа MobileNetV2. Спочатку ваги мережі заморожувалися, навчалися лише шари класифікації. Далі виконувалося доналаштування моделі – розморожувалися ваги мережі та проводилося донавчання з малою швидкістю навчання. Всі зображення бралися разом без розділення на екземпляри, тобто всі зображення екземпляру мали класифікуватися у відповідності до фази розвитку окремо один від одного. Для обох задач результати такого підходу наведені в таблиці 1.

Таблиця 1. Результати класифікації з використанням передавального навчання

Задача	Точність на навчальній вибірці		Точність на перевірочній вибірці	
	передавальне навчання	донавчання	передавальне навчання	донавчання
Задача 1	0,69	0,68	0,6	**0,62**
Задача 2	0,82	0,85	0,79	**0,81**

Як видно з таблиці, точність для задачі 1 незадовільна. Агроном приймаючи рішення щодо фази розвитку оцінює сукупність характеристик рослини, тому і робиться декілька фото, які відображають різні нюанси. Пропонується аналізувати зображення не окремо, а одразу всі наявні в екземплярі. Таким чином зображення з екземпляру спочатку об'єднуються у єдиний тензор і подаються на модель, а потім для кожного екземпляра (який складається з кількох зображень) результати з усіх зображень усереднюються перед подачею на класифікатор. Результати такої моделі для обох задач подані в таблиці 2.

Таблиця 2. Результати класифікації з використанням об'єднання зображень

Задача	Точність на навчальній вибірці	Точність на перевірочній вибірці
Задача 1	79,23	77,46
Задача 2	94,51	86,96

Результати показали підвищення точності на 15% для задачі 1 та на 5% для задачі 2. Це дає можливість зробити висновок в перспективності застосування підходу щодо об’єднання зображень екземпляру для підвищення точності класифікації.

4. ВИСНОВКИ

В дослідженні було розглянуто можливості застосування підходів штучного інтелекту до аналізу фаз розвитку рослин. Було запропоновано використання об’єднання кількох зображень з одного і того ж поля для підвищення точності класифікації фаз розвитку кукурудзи. Проведені експерименти показали точність класифікації 86%. В подальших дослідженнях варто розглянути більших набір даних та інші мережі для виділення ознак з зображення. Створені моделі можуть входити в модель штучного інтелекту, який використовується в системі моніторингу сільськогосподарських полів чи в розумних теплицях.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. AMIRI, Mahdis; POURGHASEMI, Hamid Reza. Mapping the NDVI and monitoring of its changes using Google Earth Engine and Sentinel-2 images. In: Computers in Earth and Environmental Sciences. Elsevier, 2022. p. 127-136.

2. SUNEJA, Bhavesh, et al. Cloud-based tomato plant growth and health monitoring system using IOT. In: 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM). IEEE, 2022. p. 237-243.

3. CHEN, Tony; YIN, Huiming. Camera-based plant growth monitoring for automated plant cultivation with controlled environment agriculture. Smart Agricultural Technology, 2024, 8: 100449.

НАПІВКЕРОВАНЕ НАВЧАННЯ ЗОРОВОГО ТРАНСФОРМЕРА ДЛЯ ЗАДАЧІ СЕГМЕНТАЦІЇ ДОРОЖНЬОГО ТРАФІКУ В НЕСТРУКТУРОВАНОМУ СЕРЕДОВИЩІ

Шабо О.[1], Шаповал Н.[2]

Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”, Київ, Україна

[1] andriyshabo@gmail.com [0009-0008-2661-4752], [2] shovgun@gmail.com [0000-0002-8509-6886]

Напівкероване навчання (SSL) є перспективним напрямком для забезпечення зорового трансформера (ViT) необхідною кількістю даних, адже ручна анотація в семантичній сегментації (SS) займає багато часу. Пропонується стабільний напівкерований метод навчання ViT з використанням невеликої кількості маркованих даних. Поєднання набору сильних аугментацій та парадигми подвійного вчителя забезпечує хорошу продуктивність для SS дорожнього трафіку в неструктурованому середовищі.

Ключові слова: напівкероване навчання, зоровий трансформер, семантична сегментація, неструктуроване середовище.

1. ВСТУП

При використанні ViT для комп’ютерного зору (CV) зазвичай виникають дві основні проблеми. Першою є потреба ViT у великій кількості навчальних даних, що пов’язане з відсутністю таких індуктивних упереджень CNN як інваріантність до перенесень об’єктів та хороша локальність. Цей недолік був частково усунений навчанням ViT на масивних наборах даних як ImageNet1k з наміром подальшого донавчання на конкретному датасеті. Проте залишається актуальним питання як використати немарковані дані з датасету для донавчання, щоб додатково покращити ефективність. Іншою вадою є потреба ViT у великій обчислювальній потужності, особливо у випадку розв’язання задач високороздільного щільного передбачення, внаслідок квадратичної складності softmax.

В даній роботі показано, що використання набору сильних аугментацій дозволяє виконати стабільне напівкероване навчання ViT на основі парадигми двох вчителів для SS дорожнього трафіку в неструктурованому середовищі не вимагаючи сильного налаштування гіперпараметрів. Зазвичай в системах автономної навігації відсутні потужні GPU, а доступ до хмарних обчислень обмежений, адже рішення стосовно дорожньої ситуації потрібно приймати негайно. Тому в роботі у якості моделі використовується трансформер із сімейства EfficientViT [1] завдяки його невисоким потребам в обчислювальних потужностях та високій швидкості обробки зображень.

2. ФОРМУЛЮВАННЯ ПРОБЛЕМИ

SS належить до сфери CV і є задачею високороздільного щільного передбачення. Маски для SS є комплекснішими, ніж мітки для класифікації зображень (IC), а отже їх створення є набагато більш витратним за часом процесом. Велика кількість маркованих даних може бути також відсутня внаслідок етичних, легальних чи ліцензійних перешкод. SS відіграє важливу роль в автономній навігації, адже забезпечує систему важливою інформацією про навколишнє середовище. Більшість експериментів із SS проводяться в контрольованих умовах на даних зі стабільним середовищем. Проте на дорозі часто виникають нестандартні ситуації, що погано

представлені в цих датасетах. На відміну від популярного Cityscapes [2] для SS дорожнього трафіку з даними зібраними в європейських містах, Indian Driving Dataset (IDD) [3] містить зображення отримані в неструктурованому середовищі індійських міст. Навколишнє оточення автомобіля в цьому датасеті містить погано визначені межі проїжджої частини і пішохідних доріг, щільний трафік і велику різноманітність його учасників, включно з різними тваринами. Таким чином даний набір є кращим кандидатом для навчання моделі нестандартним умовам на дорозі.

3. МАТЕМАТИЧНІ ОСНОВИ ВИКОРИСТАНИХ МЕТОДІВ

3.1. Загальна схема напівкерованого навчання

В процесі експериментів було виявлено, що концепція учень-вчитель не забезпечує позитивних результатів для навчання ViT в неструктурованому середовищі внаслідок проблеми колапсу моделі. Задля розв'язання цієї проблеми пропонується створення алгоритму навчання на основі Dual Teacher [4], що використовує CutMix [5] та ClassMix [6]. Ці дві сильні аугментації змінюють одна-одну в кінці кожної епохи разом з вчителем, що дозволяє запобігти наближенню параметрів учня до вчителів. У випадку датасету з неструктурованим середовищем кількість можливих положень об'єктів, їх підвидів та способів перекривання сильно зростає. Навчання в цих умовах також ускладнюється при малій кількості маркованих даних. Тому пропонуються адаптації CutMix та ClassMix для SSL, щоби знизити вплив цих недоліків. Застосування аугментацій в SS також ускладнене тим, що афінні і деякі інші трансформації мають вплив і на маски. Щоби розв'язати згадані проблеми пропонується наступний алгоритм SSL ViT, що зображений на рис. 1.

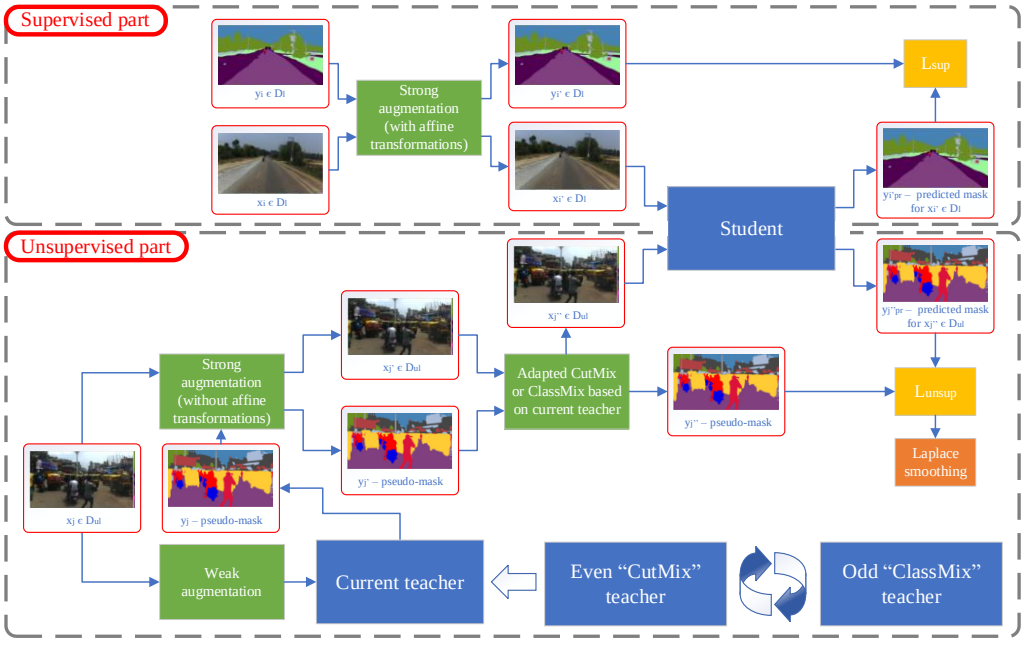


Рисунок 1. Схема запропонованого алгоритму напівкерованого навчання

3.2. Опис використаних аугментацій та їх модифікацій

Для забезпечення потреби ViT у великій кількості навчальних даних, що є особливо актуальним у випадку керованого навчання, використовується набір сильних аугментацій на (рис. 1). В якості блоку "Strong augmentations" використана послідовність RandomResizeCrop з малим діапазоном можливого масштабу та співвідношення сторін для запобігання перенавчання, RandomHorizontalFlip та RandAug [7]. RandAug була адаптована до SS так, щоб

аугментація застосовувалась і до маски зображення у випадку, якщо трасформація афінна. В якості блоку “Weak augmentations” використано RandomHorizontalFlip та ColorJitter.

CutMix, представлена як покращена версія Mixup та Cutout, та визначається формулами (1) – (8).

$$I = \{i | i \in \mathbb{N}, 1 \le i \le \mathcal{N}\}, \quad (1)$$

$$\lambda_i \sim \mathfrak{B}(4,4), \quad (2)$$

$$r_i^x \sim \mathcal{U}(0, W), \quad (3)$$

$$r_i^y \sim \mathcal{U}(0, H), \quad (4)$$

$$r_i^w = W\sqrt{1-\lambda_i}, \quad (5)$$

$$r_i^h = H\sqrt{1-\lambda_i}, \quad (6)$$

$$X_i = (1-\mathcal{B}) \odot X_i + \mathcal{B} \odot X_{\sigma(I)_i}, \quad (7)$$

$$Y_i = (1-\mathcal{B}) \odot Y_i + \mathcal{B} \odot Y_{\sigma(I)_i}, \quad (8)$$

де I – індекси зображень в пакеті розміром $\mathcal{N}$, $\mathfrak{B}$ – бета розподіл, $\mathcal{U}$ – рівномірний розподіл, W та H – ширина та висота зображення, r_i^x та r_i^y – координата лівого верхнього кута прямокутника $\mathcal{B}$, що накладається, по осі x та осі y, r_i^w та r_i^h – ширина та висота $\mathcal{B}$, ‘$\odot$’ – оператор по-піксельного множення, $\sigma(I)$ – випадкова перестановка набору I.

При аугментації немаркованих зображень на певну область часто можуть перенестись пікселі з меншою впевненістю. Для зниження негативного впливу цієї проблеми пропонується адаптація ProbCutMix, що задається використанням формул (9) – (11) замість формули (2).

$$c_j^i = \max_{c=\overrightarrow{1,N_c}} p_c^{i,j}, \quad (9)$$

$$\bar{c}_i = \frac{1}{HW}\sum_{j=1}^{HW} c_j^i, \quad (10)$$

$$\lambda_i = \frac{\bar{c}_i}{\bar{c}_i + \bar{c}_{\sigma(I)_i}}, \quad (11)$$

де $p_c^{i,j}$ – передбачена моделлю ймовірність належності j-го пікселя i-го зображення до c-го класу, N_c – кількість класів, c_j^i – впевненість моделі у прогнозі класу j-го пікселя i-го зображення, $\bar{c}_i$ – середня впевненість моделі у прогнозованій масці для i-го зображення. Завдяки цим змінами розмір $\mathcal{B}$ буде тим більшим, чим менша середня впевненість вчителя в модифікованому зображенні і чим більша в зображенні звідки вирізається $\mathcal{B}$.

Аугментація ClassMix для SS переносить половину випадкових класів для кожного зображення з перемішаного випадковим чином пакету на зображення на тій же позиції у вихідному пакеті. Вона добре підходить для сегментації дорожнього трафіку завдяки тому, що накладені на зображення об’єкти часто потрапляють в правдоподібне положення. Пропонується пара модифікацій для посилення ефективності ClassMix для немаркованих даних. У ObjectClassMix випадковий вибір класів виконується з усіх присутніх на зображенні, окрім ‘drivable’ та ‘sky’. Метою є зниження частоти ситуацій, коли дорога чи небо перекриватиме більш складні для сегментації об’єкти, адже дані два класи займають велику площу. У MaxClassMix замість випадкового набору обирається половина класів з найвищою середньою впевненістю, визначеною формулами (12)-(13), а у ProbClassMix класи вибираються випадковим чином за розподілом ймовірностей заданим формулою (14).

$$\mathcal{C}_k^i = \left\{ j \middle| \underset{c=\overrightarrow{1,N_c}}{argmax}\, p_c^{i,j} = k \right\}, \quad (12)$$

$$\bar{c}_k^i = \frac{1}{|\mathcal{C}_k^i|}\sum_{j \in \mathcal{C}_k^i} c_j^i, \quad (13)$$

$$p_k^i = \frac{\bar{c}_k^i}{\sum_{k=1}^{N_c} \bar{c}_k^i}, \quad (14)$$

де C_k^i – всі пікселі i-го зображення класифіковані моделлю як k-тий клас, $\bar{c}_k^i$ - середня впевненість моделі в k-тому класі на i-тому зображенні, p_k^i– ймовірність вибору k-го класу, для переносу пікселів з даним класом з другого зображення у перше. Мета цих модифікацій – це збільшення середньої впевненості моделі в масці зображення, що отримується в результаті.

3.3. Використання попереднього керованого навчання

Внаслідок того, що на початку навчання модель робить неякісні передбачення, маски згенеровані вчителем вповільнюють процес навчання. Негативний вплив даного явища можна частково зменшити послідовно збільшуючи коефіцієнт біля некерованої складової функції втрат. Проте замість цього пропонується провести попереднє керване навчання моделі використовуючи лише марковані дані, а після цього виконати напівкероване донавчання. Для керованого навчання використовується лише керована частина з (Рис. 1) а також втрати, що задаються формулою (15); для напівкерованого донавчання ж використовується вся схема навчання з (Рис. 1) та комбіновані втрати, що задаються формулами (15) – (17).

$$\mathcal{L}_{sup} = -\frac{1}{B_l}\sum_{i=1}^{B_l}\frac{\sum_{j=1}^{HW}\mathbb{I}\{y_j^i \neq ignore\_class\} logp_{y_j^i}^{i,j}}{\sum_{j=1}^{HW}\mathbb{I}\{y_j^i \neq ignore\_class\}}, \quad (15)$$

$$\mathcal{L}_{unsup} = -\frac{1}{B_{ul}}\sum_{i=1}^{B_{ul}}\frac{1}{HW}\sum_{j=1}^{HW} logp_{y_j^i}^{i,j}, \quad (16)$$

$$\mathcal{L}_{comb} = \mathcal{L}_{sup} + \mu\mathcal{L}_{unsup}, \quad (17)$$

де $p_{y_j^i}^{i,j}$ – передбачена моделлю ймовірність належності j-го пікселя i-го зображення до правильного класу y_j^i, B_l та B_{ul} – розміри пакету маркованих та немаркованих зображень, $ignore\_class$ – клас, що відмічає пікселі чиї втрати не враховуються, $\mathcal{L}_{sup}$ та $\mathcal{L}_{unsup}$– керована та некерована втрати, $\mu = 5$ – вага некерованих втрат, $\mathcal{L}_{comb}$ – комбіновані втрати для SSL.

3.4. Зважування пікселів некерованої складової функції втрат

В процесі навчання маски для немаркованих даних генерує вчитель. Таким чином, не всі пікселі відповідають маркуванню, якби воно було виконане вручну. Поширення похибки стандартним чином для всіх пікселів призводить до вивчення моделлю неправильного розподілу об’єктів на зображенні, що не дозволяє підвищити ефективність навчання використовуючи немарковані дані. Стандартне розв’язання цієї проблеми в IC, тобто фільтрація за допомогою порогу, в SS не є перспективним, адже модель по різному впевнена в різних пікселях на зображенні, а поріг може суттєво змінюватись віл пакету до пакету. Тому пропонується використати Laplace Smoothing [8], що відсіює пікселі з впевненістю меншою за q-тий квантиль, а решту зважує, де вища впевненість забезпечує пікселю більшу вагу.

4. ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

4.1. Перевірка ефективності модифікацій CutMix та ClassMix

Для проведення дослідження ефективності представленого алгоритму було використано EfficientViT B1 з попередньо навченими на Cityscapes вагами для основи мережі і випадковими вагами для сегментаційної голови. В якості датасету виступає набір з Indian Driving Dataset з 373 маркованими і 2603 немаркованими тренувальними, 700 валідаційними і 1048 тестовими зображеннями з найвищою ієрархією з 7-ми класів. Для всіх експериментів використано PyTorch, оптимізатор AdamW, присвоювач швидкості навчання cosine decay, навчання впродовж 80 епох з розігрівом 10% та обрано значення mIoU як основну метрику. Для попереднього керованого навчання використано розмір пакету 16, швидкість навчання для основи мережі 10^{-4} та 10^{-3} для голови. Після цього проводиться напівкероване донавчання з

розміром маркованого пакету 2 і немаркованого 8, швидкістю навчання для основи мережі $5 \cdot 10^{-4}$ і $2 \cdot 10^{-3}$ для голови та коефіцієнтом $\alpha = 0.99$ для EMA. Тестування ефективності комбінацій аугментацій представлено в табл. 1.

Таблиця 1. Результати навчання моделі залежно від використаних аугментацій

	Supervised pretrain	Model 1	Model 2	Model 3	Model 4	Model 5
CutMix type	-	Ordinary	Prob	Prob	Prob	Prob
ClassMix type	-	Ordinary	Ordinary	Object	Prob	Max
mIoU	65.94	68.16	**68.84**	68.58	68.49	68.18
Dice score	77.05	78.8	**79.39**	79.22	79.14	78.88
Accuracy	76.71	77.9	**78.85**	78.57	78.46	**78.85**
Precision	77.47	79.98	**80.2**	80.12	80.07	79.08
mIoU gain	-	2.22	**2.9**	2.64	2.55	2.24

Як бачимо, напівкероване донавчання підвищує значення mIoU порівняно з попереднім керованим навчанням. Судячи з результатів використання модифікованої версії ProbCutMix покращує значення mIoU, проте подальша заміна звичайного ClassMix на одну з модифікацій, що підвищувала б середню впевненість масок, його погіршує. Таким чином при навчанні ViT в даних умовах врахування впевненості вчителя в згенерованих масках є важливішим для CutMix, ніж для ClassMix. Отже надалі використовуватиметься саме конфігурація “Model 2”.

4.2. Перевірка ефективності попереднього керованого навчання

Наступним етапом є перевірка ефективності попереднього керованого навчання. Тут розмір пакету при керованому навчанні знижується до 10 і проводиться зменшення зображень до більшого розміру перед подачею на вхід. При напівкерованому навчанні без попереднього керованого навчання коефіцієнт перед некерованою компонентною втрат лінійно збільшується в процесі навчання. Результати цих експериментів зображені в табл. 2.

Таблиця 2. Результати перевірки ефективності попереднього керованого навчання

	Supervised pretrain	Semi-supervised only	Semi-supervised with supervised pretrain
mIoU	69.37	70.59	**71.88**
Dice	79.72	80.76	**81.94**
Accuracy	79.01	80.13	**81.61**
Precision	80.63	81.78	**82.44**
mIoU gain	-	1.22	**2.51**

Бачимо, що немарковані дані покращують результат. Проте, при використанні попереднього керованого навчання перед напівкерованим донавчанням, отримуємо більше значення mIoU, що доводить ефективність даного прийому. Приклади сегментації зображень найкращою з отриманих моделей зображено на рис. 2.

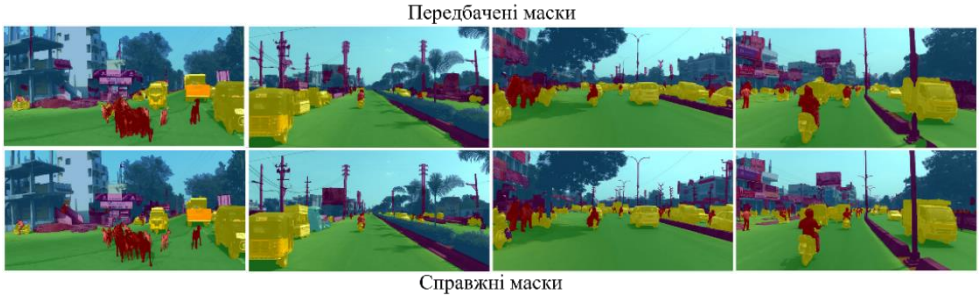


Рисунок 2. Приклади сегментації зображень

Як бачимо, враховуючи складність датасету, модель добре справляється із виділенням основних учасників дорожнього руху а також навколишніх об’єктів, що співпадає із висновками, які можна зробити зі значень метрик.

5. ВИСНОВКИ

В даному дослідженні було представлено алгоритм SSL ViT для SS дорожнього трафіку в неструктурованому середовищі та експериментально доведено його ефективність в реалістичних умовах, для чого використано ресурсоефективний ViT. Ґрунтуючись на отриманих результатах можна зробити висновок, що CutMix є більш чутливим до впевненості вчителя в прогнозованій масці, ніж ClassMix, отже для кращих результатів варто враховувати впевненість саме у CutMix за допомогою використання ProbCutMix. Бачимо, що застосування набору сильних аугментацій є ефективним для забезпечення ViT необхідними даними при навчанні в неструктурованому середовищі. Такий підхід, оснований на парадигмі Dual Teacher з використанням попереднього керованого навчання і Laplace Smoothing дозволяє отримати стабільний процес навчання без необхідності сильного налаштування гіперпараметрів. Хоча набір сильних аугментацій і створює зображення, що не є правдивим для людського ока, такий підхід виявився дієвим завдяки неструктурованості середовища сегментації.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Cai, H., Li, J., Hu, M., Gan, C., & Han, S. (2023). EfficientViT: Lightweight Multi-Scale Attention for High-Resolution Dense Prediction. У *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE. https://doi.org/10.1109/iccv51070.2023.01587
2. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., & Schiele, B. (2016). The Cityscapes Dataset for Semantic Urban Scene Understanding. У *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/cvpr.2016.350
3. Varma, G., Subramanian, A., Namboodiri, A., Chandraker, M., & Jawahar, C. V. (2019). IDD: A Dataset for Exploring Problems of Autonomous Navigation in Unconstrained Environments. У *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE. https://doi.org/10.1109/wacv.2019.00190
4. Na, J., Ha, J.-W., & Chang, H. J. (2023). Switching Temporary Teachers for Semi-Supervised Semantic Segmentation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt & S. Levine (Ред.), *Advances in Neural Information Processing Systems* (D. Han & W. Hwang, Автор кореспонденції; Т. 36, с. 40367–40380). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2023/file/7eeb42802d3750ca59e8a0523068e9e6-Paper-Conference.pdf
5. Yun, S., Han, D., Chun, S., Oh, S. J., Yoo, Y., & Choe, J. (2019). CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features. У *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE. https://doi.org/10.1109/iccv.2019.00612
6. Olsson, V., Tranheden, W., Pinto, J., & Svensson, L. (2021). ClassMix: Segmentation-Based Data Augmentation for Semi-Supervised Learning. У *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE. https://doi.org/10.1109/wacv48630.2021.00141
7. Cubuk, E. D., Zoph, B., Shlens, J., & Le, Q. V. (2020). Randaugment: Practical automated data augmentation with a reduced search space. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE. https://doi.org/10.1109/cvprw50498.2020.00359
8. Jin, Y., & Lin, D. (2022). Semi-Supervised Semantic Segmentation via Gentle Teaching Assistant. У *Advances in Neural Information Processing Systems* (J. Wang, Автор кореспонденції; Т. 35, с. 2803–2816). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2022/file/12d286282e1be5431ea05262a21f415c-Paper-Conference.pdf

КЛАСТЕРИЗАЦІЯ ТЕКСТОВИХ ДАНИХ НА ОСНОВІ ВКЛАДЕННЯ СЛІВ

Шутяк Д.О.[1], Подколзін Г.Б.[2]

Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна

[1] dima.shutyak@gmail.com [0009-0008-6480-3706],
[2] podkolzin.gleb@lll.kpi.ua [0000-0002-7120-2772]

Кластеризація текстових даних є ключовою частиною аналізу неструктурованих текстових повідомлень. Для використання цих методів текстові дані необхідно перетворити в векторні, тобто провести вкладання слів (word embedding) [1]. В цій статті ми наводимо модифікацію метода кластеризації HDBSCAN* [2] з використанням власної метрики та інших параметрів саме для класифікації неструктурованих текстових даних.

Ключові слова: текстова кластеризація, вкладення слів, великі мовні моделі, машинне навчання, python.

Keywords: Text clustering, word embedding, large language models, machine learning, python.

1. ВСТУП

В аналізі текстових даних кластеризація має дуже високий потенціал саме дякуючи алгоритмам векторизації (embedding) цих текстів. Векторні простори, які створюють ці мовні моделі мають дуже якісні властивості, тому використання різних метрик дистанцій та загалом моделей класифікації (кластеризації) на них дають дуже якісні просторові зв’язки між текстами зі схожим/пов’язаним змістом. Побудовані таким чином кластери можуть в подальшому бути використані або разом з моделями автоматичного реферування тексту для створення міток або категорій для вхідних текстових даних, на основі їх приналежності до певного кластеру текстів.

З розвитком великих мовних моделей було також створено дуже велику кількість моделей векторизації тексту, оскільки це є основним ключовим функціоналом нейронних мереж з аналізу тексту. Моделі які показали найкращі результати за MTEB (Massive Text Embedding Benchmark) [3] – це зазвичай моделі дуже великої розмірності, а тобто високого використання пам’яті та відносно низької швидкості, тому в цій роботі будуть порівняні саме моделі з набагато більшою швидкістю, але ті які мають якісні результати по оцінці отриманих вкладень. Це моделі такі як: nomic-embedding-v1 [4], gte-v1.5 [5], stella_en_400M_v5 [6].

За основу беремо алгоритм кластеризації – HDBSCAN. Це ієрархічний алгоритм кластеризації, а потім використовуючи техніку для видалення плоскої кластеризації, заснованої на стабільності кластерів.

2. ВИКОРИСТОВУВАНИЙ МЕТОД КЛАСТЕРИЗАЦІЇ

Метод полягає в наступному: множина точок об'єднується в зв’язний граф за геометричними властивостями точок. На наступному кроці ребра, що відповідають дуже віддаленим точкам відкидаються. Цей процес є ітеративним. В результаті отримуємо Набір зв’язних частко графу, які і описують очікувану кластеризацію.

Далі цей метод реалізуємо з використанням узагальнення алгоритму HDBSCAN*.

2.1. Загальний опис алгоритму HDBSCAN* та його узагальнення

Спочатку трансформуємо простір початкових даних відповідно до щільності/розрідженості. Ущільнюємо ієрархію кластера на основ мінімального розміру кластера. Обираємо стабільні кластери з ущільненого дерева. Будуємо мінімальне кістякове дерево графа, зваженого за відстанню (дистанцією).

Виділяємо кластери на основі щільності для відокремлення шуму.

Для оцінки щільності використовуємо метрики *(distance)*.

Розглядаємо узагальнену метрику:

$$d_k(a,b) = max\{core_k(a), core_k(b), d(a,b)\},$$

де $core_k(a) -$ це відстань до k-го найближчого сусіда. (впорядковуємо відстані від точки a за зростанням); $d(a,b) -$ відстань між точками $a,\ b$. Саме для цієї відстані, метрики будемо розглядати різні варіанти, що і буде одним із методів модифікації алгоритму HDBSCAN*.

Структура алгоритму полягає в знаходженні кластерів з більшою щільністю. Реальні дані мають викиди та пошкодження, які будуть сприйматись як шум.

Основою алгоритму є однозв'язна кластеризація, вона може бути досить чутливою до шуму: одна точка даних шуму може діяти як "міст" між кластерами високої щільності, з'єднуючі їх разом. Алгоритм має бути стійким до шуму.

Саме тому ми використовуємо узагальнену відстань $d_k(a,b)$, яка розріджує шум, не впливаючи на кластери з високою щільністю.

Застереження тут полягає в тому, що, збільшення k збільшує кількість точок, які інтерпретуються як шум.

Перший крок алгоритму полягає в побудові графу даних, вага ребер якого є відстанями між точками за обраною метрикою. Будуємо граф наступним чином: для кожної точки x знаходимо радіус ($r_k(x)$) такий, що коло з цим радіусом містить фіксовану кількість – k точок. В статті [7] було проаналізовано, що найбільш ефективним значенням цього параметру k було рівне 100. Далі за зростаючим додатнім параметром r обираємо всі точки, для яких $r_k(x) \le r$. І в цій множині точок з'єднуємо ребрами точки, для яких відстань $\left|\left|x_i - x_j\right|\right| \le \alpha r$. Ітеративно продовжуємо цей процес за зростанням r. Таким чином отримуємо зв'язний граф з усіх наших точок.

Наша модифікація алгоритму HDBSCAN* полягає в варіюванні параметру α та розгляду різних метрик $d(a,b)$.

На наступному кроці розглянемо порогове значення, починаючи з високого та поступово знижуючи. Відкидаємо будь-як ребра, вага яких перевищує цей поріг. Відкидаючи таким чином ребра ми розділяємо граф на зв'язні компоненти.

Таким чином ми будуємо ієрархію зв'язних компонентів для різних порогових рівнів. Де порогом є вага відкинутих ребер.

3. ДЕТАЛІЗАЦІЯ РОБОТИ АЛГОРИТМУ

Компонента зв'язності графа має n^2 ребер. Для побудови ефективного алгоритму треба оптимізувати, зменшувати кількість кроків відкидання ребер. Для цього треба знайти мінімальний набір ребер, такий що відкидання будь-якого ребра з набору спричинить роз'єднання компонентів. Також потрібно, щоб в цьому наборі не було ребра меншої ваги, яке може з'єднати компоненти.

Таким чином, на першому кроці алгоритму будуємо мінімальне кістякове дерево графа, зваженого за відстанню (дистанцією), будуємо за допомогою алгоритма Прима [8].

Ми будуємо дерево по одному ребру, завжди додаючи ребро найменшої ваги, яке з'єднує поточне дерево з вершиною, якої ще немає в дерев.

Наприкінці ми отримуємо єдине дерево – ієрархія всіх даних, а не відокремлені кластери, тому отримане дерево необхідно перетворити і з нього вилучити всі побудовані кластери. Після цього починається другий крок алгоритму.

На початку нам потрібно поняття мінімального розміру кластера, який ми беремо як параметр для HDBSCAN. Маючи значення для мінімального розміру кластера, ми тепер можемо пройти по ієрархії та під час кожного розбиття та перевіряти кількість точок в нових кластерах. Та частина при розділенні, що має менше точок, ніж мінімальний розмір кластера фіксується, як частина попереднього по ієрархії, до розділення, кластеру і не є самостійним кластером.

В подальшому ми відрізняємо кластери та відокремлені точки за стійкістю до розбиття. Кластери, які зберігаються при великій кількості кроків розбиття є очікуваними кластерами. Короткочасні кластери, ймовірно, є лише артефактами підходу. Якщо на певному кроці при розбитті отримуємо обидві частини, що мають менше точок, ніж мінімальний розмір кластера, то цей кластер не підлягає подальшому розбиттю. Інша ситуація полягає в наступному: одна, перша, частина має менше точок, ніж мінімальний розмір кластера, а друга частина при одному з наступних розбиттів поділяється на два кластери (частини, що більші за мінімальний розмір кластера). Тоді перша частина вважається шумом.

Формалізуємо стійкість кластера до розбиття. Для цього нам потрібна інша міра, ніж відстань, щоб розглянути стійкість кластерів. Для цього ми будемо використовувати $\lambda = \frac{1}{distance}$, де *distance* - це вага ребра графа. Для заданого кластера ми можемо визначити два значення:

λ_{birth} – значення лямбда, коли з'явився кластер при розділенні "батьківського" та λ_{death} – значення лямбда, коли кластер поділяється на дві частини, що одночасно або менші за мінімальний розмір кластера, або більші. Для точок p, що належать тій частині кластеру, яка при розділенні містить менше точок, ніж мінімальний розмір, визначаємо λ_p – значення при якому ця точка перестає належати кластеру (момент розділення), це значення знаходиться десь між λ_{birth} та λ_{death} кластеру, до якого належать p.

Маючи ці значення, можна задати стійкість для кожного кластеру:

$$s_{cluster} = \sum_{p \in cluster} (\lambda_p - \lambda_{birth}).$$

Далі проводиться ітеративний процес для визначення стабільних кластерів починаючи з кінцевих вершин (листя): для кожного кластеру знаходиться його стійкість та порівнюється із сумою стійкостей його дочірніх кластерів. Якщо сума стійкостей дочірніх кластерів більша за власну стійкість кластера, то його стійкість відкидається та задається цією сумою для подальших кроків. А якщо ж власна стійкість кластера більша за суму, то цей кластер вважається очікуваним, а всі його "нащадки" перестають вважатись такими, якщо вони раніше були.

Пройшовши ітерацію до кореневого вузла дерева всі кластери, які залишились очікуваними відбираються і складають кластеризацію.

В цей момент відбору кластерів всі точки, які не опинились в жодному кластері відмічаються як шум.

В коді використовуються посилання на стандартні бібліотеки.

Змістовний фрагмент коду проведення кластеризації

```
clf = hdbscan.HDBSCAN(min_cluster_size=100,
                      min_samples=10,
                      cluster_selection_epsilon=0.0,
                      metric='euclidean',  # значення metric ставимо “l1”, “l2”,
                                           “infinity”, код для яких наведено вище
                      cluster_selection_method='eom',
                      alpha=1,  # перебираємо значення для alpha:
        # array([0.5  , 0.625, 0.75 , 0.875, 1.   , 1.125, 1.25 , 1.375, 1.5  , 1.625, 1.75 , 1.875, 2.   ])
                      prediction_data=True)
cluster_labels2 = clf.fit(u2)

dblabels2 = cluster_labels2.labels_

n_clusters2_ = len(set(dblabels2)) - (1 if -1 in dblabels2 else 0)
n_noise2_ = list(dblabels2).count(-1)
```

4. МЕТРИКА ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ РОБОТИ

Для оцінки результатів роботи алгоритму використовуємо наступні метрики.

4.1 Silhouette Score Algorithm

Метрика для оцінки оптимальної кількості кластерів для певного набору даних

$S(x_i) = \frac{b(x_i)-a(x_i)}{max(a(x_i),b(x_i))}$ – силует для певної точки x_i,

де $a(x_i) = \frac{1}{|C_I|-1}\sum_{j\in C_I, i\neq j} \quad d(x_i, x_j) -$ середня відстань від x_i до інших точок даних у тому самому кластері (C_I);

$b(x_i) = min_{J\neq I}\left\{\frac{1}{|C_J|}\sum_{j\in C_J} \quad d(x_i, x_j)\right\}$ – найменша середня відстань від x_i до точок даних в іншому кластері (C_J);

$|C_I|$ –кількість точок в кластері C_I;

$d(x_i, x_j) -$ відстань між точками x_i *та* x_j;

$a(x_i) -$ міра подібності x_i до точок власного кластера;

$b(x_i)$ –міра неузгодженості x_i з сусіднім кластером.

Значення $S(x_i)$ близьке до 1, означає, що дані належним чином згруповані. Зменшення значення $S(x_i)$ вказує на погіршення кластеризації для точки x_i. Значення $S(x_i)$ варіюється від –1 до 1.

Визначаємо середнє значення для $S(x_i)$: $S_J = \frac{1}{|C_J|}\sum_{x_i\in C_J} S(x_i)$

Значення S_J є мірою того, наскільки тісно згруповані всі точки в кластері C_J.

Код цієї метрики є в стандартних бібліотеках.

5. РЕЗУЛЬТАТИ РОБОТИ

Робота алгоритму виявила, що для кожної з відстаней кількість кластерів не залежить від параметру альфа, параметру, який відповідає за побудову мінімального кістякового дерева графа. Але кількість кластерів змінюється для різних відстаней.

5.1. Кількість кластерів та точок шуму

Кількість кластерів та точок шуму наведена в табл. 1.

Таблиця 1. Результати роботи для різних альфа

	ВІДСТАНЬ L1	ВІДСТАНЬ L2	ВІДСТАНЬ L-inf
α = 0.5; α = 1.0625; α = 1.4375; α = 2.0	number of clusters: 13 number of noise points: 154	number of clusters: 12 number of noise points: 84	number of clusters: 12 number of noise points: 74

5.2. Графічні результати роботи

Аналіз графічних результатів (рис. 1–3) вказує, що кластеризація суттєво залежить від вибору відстані.

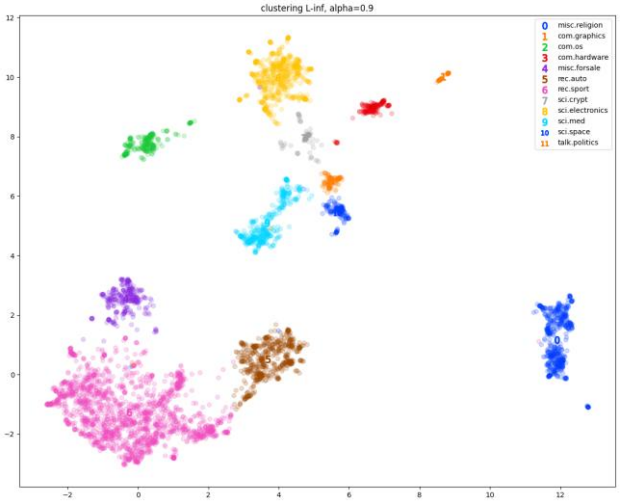


Рисунок 1. Випадок відстані L-infinity

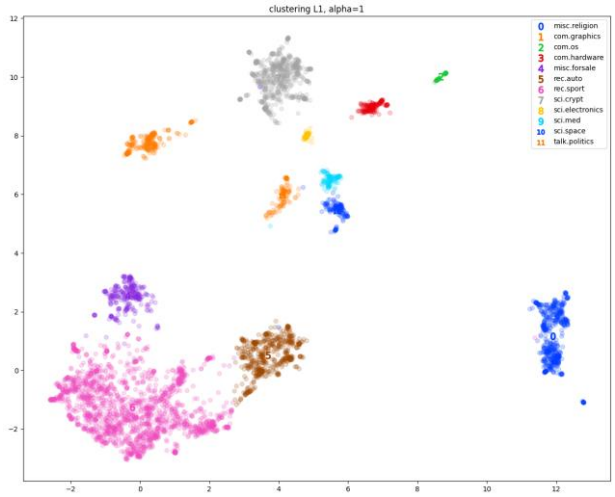


Рисунок 2. Випадок відстані L1

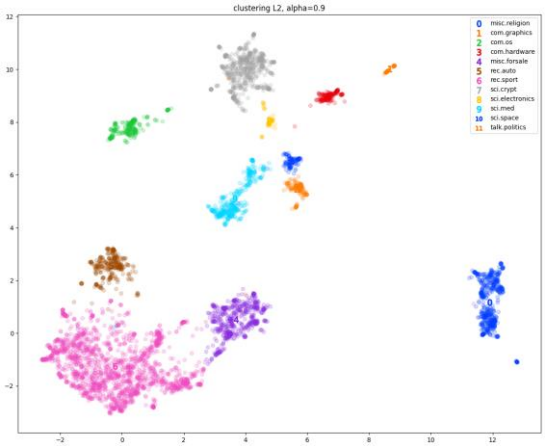


Рисунок 3. Випадок відстані L2.

6. РЕЗУЛЬТАТИ ПОРІВНЯННЯ ЗА ДОПОМОГОЮ МЕТРИК

Значення метрики Silhouette Score (табл. 2) аналізує результати наступним чином: близьке до 1, означає, що дані належним чином згруповані. Зменшення значення вказує на погіршення кластеризації. Значення варіюється від –1 до 1.

Таблиця 2. Значення метрики Silhouette Score для різних відстаней

	L1	L2	L-infinity
α = 0.5; α = 1.0625; α = 1.4375; α = 2.0	0.6313755	0.6347585	0.6407703

Таким чином бачимо, що для кожної відстані зміна альфа не впливає на результат. Кращий результат дає відстань L-infinity, але всі відстані дають достатньо якісний результат.

Тим самим для подальшої роботи з алгоритмом достатньо використовувати альфа, наприклад 1.

Для порівняння наведемо значення метрики Silhouette Score на початково розмічених кластерах: $S(C) = 0{,}41319692$.

7. ВИСНОВКИ

В роботі було модифіковано алгоритм HDBSCAN* та проаналізовано результати його роботи при варіюванні параметрів та відстаней, які закладені в нього. Виявилось, що результати роботи не залежать від зміни значення параметрів, але суттєво залежать від закладеної функції відстані. Найкращий результат надала функція відстані L-infinity.

Для значення метрики Silhouette Score на початково розмічених кластерах отримали результат гірший за результати роботи метода. Пояснити цей факт можливо дуже тісним переплетенням даних, розділення яких на різні кластери дуже складно формалізувати. А саме головна проблема виникла в розділенні на різні кластери:

'com.graphics'; 'com.os'; 'com.hardware'; 'misc.forsale'; 'sci.electronics'.

Всі вони стосуються комп'ютерної тематики, тому використовують дуже схожі терміни.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Petukhova A., Matos-Carvalho J. P., Fachada N., “Text Clustering with LLM Embeddings”. https://arxiv.org/abs/2403.15112;
2. Malzer C., Baum M., “A Hybrid Approach To Hierarchical Density-based Cluster Selection”. https://arxiv.org/abs/1911.02282;
3. Muennighoff N., Tazi N., Magne L., Reimers N., “MTEB: Massive Text Embedding Benchmark”. https://arxiv.org/abs/2210.07316;
4. Nussbaum Z., Morris J. X., Duderstadt B., Mulyar A., “Nomic Embed: Training a Reproducible Long Context Text Embedder”. https://arxiv.org/abs/2402.01613 ;
5. Zehan L, Xin Z., Yanzhao Z., Dingkun L., Pengjun X., Meishan Z., “Towards General Text Embeddings with Multi-stage Contrastive Learning”. https://arxiv.org/abs/2308.03281;
6. Dun Z., ”stella_en_400M_v5”. https://huggingface.co/dunzhang/stella_en_400M_v5;
7. Cormen T. H., Leiserson C. E., Rivest R. L., Stein C., “Introduction to Algorithms, Third Edition”. MIT Press, 2009. ISBN 0-262-03384-4. Section 23.2: The algorithms of Kruskal and Prim, pp. 631—638;
8. Набір даних , який було використано в роботі - Lang K., “20Newsgroup dataset” http://qwone.com/~jason/20Newsgroups/. DOI: 10.24432/C5C323;
9. Saha R.,“Influence of various text embeddings on clustering performance in NLP”. https://arxiv.org/abs/2305.03144

РЕКОМЕНДАЦІЯ ХЕШТЕГІВ МЕТОДАМИ ГЛИБОКОГО НАВЧАННЯ НА ОСНОВІ МУЛЬТИМОДАЛЬНИХ ДАНИХ

Яковлєв С.О.[1], Шаповал Н.В.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] se2001ya@gmail.com, [2] shovgun@gmail.com [0000-0002-8509-6886]

Мета даного дослідження полягає у розробці модифікованого алгоритму злиття ознак для задачі мультимодальної рекомендації хештегів. Для модифікації було взято модель, яка окрім зображення і підпису публікації додатково ще використовує частину попередньої історії користувача. В результаті було розроблено новий модуль злиття ознак, що додатково зважує важливість текстових та візуальних ознак, і може бути корисним для випадку, коли текстовий підпис є доволі коротким або повністю відсутній і відповідно слабко впливає або взагалі не впливає на зміст публікації.

Ключові слова: нейронні мережі, глибинне навчання, мультимодальні дані, хештеги, соціальні мережі.

1. ВСТУП

У сучасну епоху, коли все більшої популярності набувають соціальні мережі, такі як X, Facebook, Instagram тощо виникає неймовірно великий об'єм інформації, за рахунок публікацій, в якому може бути доволі складно розібратись звичайній людині, через що в абсолютній більшості соціальних мереж доступні для використання так звані «хештеги» – це, по суті, ключове слово, або набір слів, інших символів або токенів, зміст яких дуже стисло відображає основну тематику повідомлення, і основне призначення яких полягає у спрощенні пошуку або фільтрації контенту за відповідною тематикою або змістом.

Природним чином постає питання інтеграції методів штучного інтелекту до розв'язання задачі рекомендації релевантних хештегів у соціальних мережах, з метою автоматизації процесу категоризації контенту у них.

Рекомендація також може базуватись на одному типі даних публикації, як то тільки підпис, тільки зображення, тільки відео – тощо, так і одночасно на кількох типах даних, які характеризують публікацію, і називаються мультимодальними даними.

Власне, ця робота присвячена розробці та модифікації моделей глибокого навчання для рекомендації хештегів на основі мультимодальних даних соціальних мереж, конкретно – на прикладі соціальної мережі Instagram.

2. ОГЛЯД ІСНУЮЧИХ РОБІТ

Аналізуючи існуючі підходи до розв'язання поставленої задачі, одразу можна зазначити їх спільну особливість – зазвичай розглядається задача «рекомендації К-хештегів», в якій прогнози отриманні нейронною мережею ранжуються, після цього з них обирається як остаточна рекомендація К найбільш релевантних, замість того, щоб просто обрати усі хештеги, що мають міру достовірності вище певного порогу тощо.

Важливо також виділити основні дві техніки розв'язання поставленої задачі: класифікація з багатьма мітками, яка відрізняється від багатокласової класифікації тим, що

класи не є взаємовиключними, тобто кожен екземпляр даних може належати одночасно кільком класам, а також генерація послідовності – де аналізується не просто набір хештегів, а саме їх послідовність, що дозволяє додатково враховувати взаємозв'язки між ними.

Говорячи про власне основні методи, можно виділити, наприклад, Attention-Based Multimodal Neural Network Model for Hashtag Recommendation (AMNN) [1], де автори розглядають задачу рекомендації хештегів, як задачу генерації послідовності, і використовують архітектуру кодувальник-декодувальник, де спочатку окремо виділяються ознаки із зображення та підпису, після чого вони конкатенуються, що допомагає отримати остаточне представлення публікації, після чого за допомогою рекурентної мережі GRU генерується послідовність хештегів.

Іншим варіантом є модель Co-Attention (CoA) [2], де рекомендація хештегів розглядається як задача класифікації з багатьма мітками, де послідовно спочатку застосовується механізм уваги до зображення на основі тексту, а потім – навпаки, які надалі застосовуються для розрахунку прихованого представлення публікації, яке потім подається на останній шар для класифікації.

Більш цікавим варіантом є модель Memory Augmented Co-attention Model (MACoN) [3] яка окрім того що використовує механізм паралельної спільної уваги для зображення і підпису одночасно, що дозволяє враховувати вплив кожного з них один на одного, ще додатково використовує попередню історію користувача щодо використання хештегів, додаючи за рахунок цього елемент персоналізації до рекомендацій.

Доволі специфічним рішенням даної проблеми є Triplet-Attention Graph Network (TAGNet) [4], де будується так званий «граф візуальної подібності» за припущення, що схожі зображення матимуть однакові хештеги, а в якості ознак кожного із вузлів виступають отримані на основі підпису а також особистого профілю користувача, після чого використовується модуль потрійної-уваги (Triplet-Attention), що розраховує вплив зображення, тексту та даних користувача один на одного.

Модель multimoDal pErSonalIzed hashtaG recommeNdation (DESIGN) [5], об'єднує ключові ознаки майже усіх згаданих вище моделей. Вона додатково до текстового підпису і зображення відповідної публікації ще використовує певну кількість попередніх постів користувача, задля додавання елементу персоналізації, а також – поєднує наведені вище дві техніки розв'язання поставленої задачі задля агрегування результатів і надання більш релевантних результатів.

Свого роду революцію у задачі рекомендації хештегів, як і в більшості інших якщо не всіх задачах обробки природної мови зробили моделі GPT(Generative Pre-trained Transformer), а точніше – розроблений чат-бот GhatGPT для взаємодії з користувачем, який можна використовувати зокрема і для мультимодальної рекомендації хештегів.

Але в цій задачі він має значний недолік, а саме те, що він може робити занадто уніфіковані рекомендації, як, наприклад, генерувати хештеги, які в загальному випадку є дуже популярними, але вже широко використовуються іншими, що може зменшити унікальність контенту користувача та обмежити його видимість серед публікацій конкурентів, що може бути критичним, наприклад, при розповсюдженні реклами у соціальних мережах.

Отже, в якості моделі для подальших досліджень було прийнято рішення взяти модель DESIGN.

3. ГЕНЕРАЦІЯ ХЕШТЕГІВ

Отже, потрібно навчити модель, яка за текстовою і візуальною складовою публікації в соціальній мережі Instagram а також за певною кількістю попередніх публікацій користувача буде здатна порекомендувати набір з K релевантних хештегів до відповідної публікації , який,

з одного боку, є в достатній мірі вичерпним, а з іншого боку – не є надлишковим. Після чого проаналізувати результати та зробити висновки щодо того, яке значення K є загалом оптимальним

Перш за все, коротко опишемо методологію базової моделі DESIGN.

Як вже зазначалось раніше, DESIGN – це модель, яка, по-суті, поєднала особливості усіх інших моделей для розв'язання поставленої задачі.

Загальна її архітектура наведена на рис. 1.

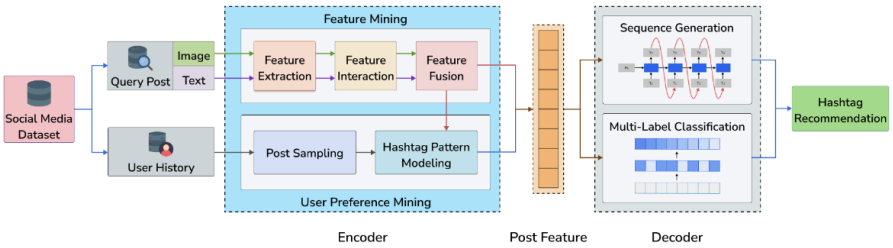


Рисунок 1. Загальна архітектура моделі DESIGN

Важливим компонентом є модуль вилучення ознак з контенту публікації (Feature Mining), який отримує на вхід зображення і підпис, а повертає – суцільний вектор ознак, з урахуванням інформації обох модальностей.

Цей модуль складається із блоку вилучення ознак з кожної модальності окремо (Feature Extraction), блоку аналізу взаємодії (Feature Interaction), що за допомогою механізмів уваги на рівні слова та паралельної спільної уваги враховує вплив текстової та візуальної складової публікації одна на одну, та блоку злиття ознак (Feature Fusion), що на основі векторів текстових та візуальних ознак формує остаточний вектор ознак змісту публікаці

Варто зауважити, що модуль злиття ознак просто сумує вектори візуальних та текстових ознак, отримані на попередньому етапі. Тобто в даному випадку, текстова і візуальна складова здебільшого враховуються однаково.

В рамках же цієї роботи, пропонується використання модифікованого модуля злиття ознак, що буде зважувати важливість кожної з модальностей публікації. А саме:

Припустимо, що на вхід модуля злиття ознак подається вектор текстових ознак t та вектор візуальних ознак v.

Якщо при використанні базового модуля злиття ознак одразу отримуємо, що вектор ознак змісту публікації дорівнює $p = t + v$, то в запропонованому методі отримуємо трохи складніший алгоритм розрахунку:

1. $f = concatenate(t, v)$;
2. $\hat{f} = BatchNorm(f)$;
3. $o = Dense(\hat{f})$;
4. $o = LeakyRelu(o)$;
5. $\hat{o} = BatchNorm(o)$;
6. $w = Dense(\hat{o})$;
7. $w = Sigmoid(w)$;
8. $p = (1 - w) \cdot t + w \cdot v$.

Тобто по суті отримуємо, що в решті-решт вектор ознак змісту публікації обчислюється не просто як сума векторів текстових і візуальних ознак, а як їх покоординатне зважене

середнє, що дозволяє додатково враховувати важливість ознак кожної з модальностей для впливу на зміст публікації.

Також, надалі, для вилучення текстових ознак, як і в оригінальній статті, використаємо мультилінгвальну модель BERT. Щодо вилучення візуальних ознак, в оригінальній роботі досліджувались дві моделі згорткових нейронних мереж – VGG-16 та ResNet50, і було з'ясовано, що модель VGG-16 з них двох є кращим рішенням, тому візьмемо її а також додатково візьмемо базову модель візуального трансформеру (ViT).

4. ПРОВЕДЕННЯ ЕКСПЕРИМЕНТІВ ДЛЯ ДОСЛІДЖЕННЯ

4.1. Метрики оцінювання якості моделей

Оцінка таких моделей є доволі складною задачею, через фактор суб'єктивності, коли кожен користувач призначає хештеги до публікації суб'єктивно, керуючись певними своїми переконаннями та мотивами, через що кількість хештегів до публікації може значно варіюватись, у той час, коли рекомендується зазвичай фіксована кількість.

Не дивлячись на все зазначене, є декілька стандартних рішень щодо метрик оцінювання якості, здебільшого – як для задачі рекомендації.

Для їх визначення, уведемо такі позначення як Gh − множина справжніх призначених хештегів для даної конкретної публікації, Rh − множина рекомендованих хештегів до наведеної публікації, та Ch − набір спільних хештегів, що визначається як перетин двох попередніх множин.

Першою метрикою є стандарте рішення для задачі рекомендації, яке називається Hit rate, і визначається формулою 1:

$$Hit\ rate = \min\{1, |Ch|\} \tag{1}$$

де $|\cdot|$ − кількість елементів у відповідній множині.

По-суті – це індикатор того, що принаймні один хештег для публікації був рекомендований вірно, в той час як усереднення по всій навчальній вибірці даватиме частку тих публікації, для яких хоча б один хештег був підібраний вірно.

Другою метрикою є точність (англ. precision), що є узагальненням відповідної метрики для задачі бінарної класифікації, і визначається формулою 2, з урахуванням попередніх позначень:

$$Precision = |Ch|/|Rh| \tag{2}$$

По-суті – це показник того, наскільки рекомендації складаються з вірних хештегів.

Альтернативою точності є повнота (англ. recall), що визначається формулою 3:

$$Recall = |Ch|/|Gh| \tag{3}$$

На відміну від точності, повнота є показником того, наскільки вірні хештеги охоплюються рекомендаціями.

Для пошуку компромісу між точністю і повнотою використовується міра F1, що по суті є гармонійним середнім між значенням відповідних метрик, і визначається формулою 4:

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

4.2. Підготовка набору даних

Для проведення експериментів було зібрано новий набір даних, що в остаточному варіанті включає 41873 публікації для 5900 користувачів, із середньою кількістю публікацій на користувача – 7,09. Було відкинуто низькочастотні хештеги, після чого кількість унікальних

хештегів становила 1734, а середня кількість хештегів на публікацію – 8.28. Додатково зауважимо, що в даному випадку публікації відображають специфіку соціальної мережі Instagram, де текстовий підпис може бути як дуже довгим, так і доволі коротким або взагалі відсутнім.

4.3. Проведення експериментів

Перейдемо безпосередньо до проведення експериментів.

Для цього розглянемо конфігурації моделі в залежності від того, яка мережа використовується для вилучення візуальних ознак, а також в залежності від модуля злиття ознак, що використовується.

В результаті, графіки значень hit rate та F1 в залежності від кількості рекомендованих хештегів K для всіх моделей можна побачити на рис. 2:

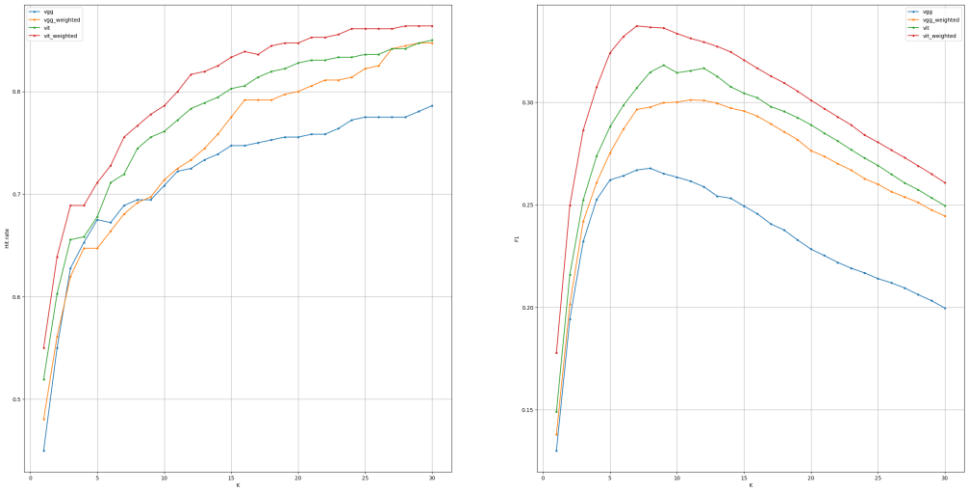


Рисунок 2. Результати для Hit rate та F1

Бачимо, що за значеннями обох метрик найкращими рішеннями для кожного значення K є модель з використанням візуального трансформера для вилучення візуальних ознак та модифікованого модуля злиття ознак.

Значень метрик для випадку K=7, при якому досягається найбільше значення F1, наведенені у табл. 1:

Таблиця 1. Результати при K = 7

Конфігурація моделі	Hit rate	Precision	Recall	F1
VGG	0.69	0.28	0.26	0.27
VGG+Weighted	0.68	0.31	0.28	0.3
ViT	0.72	0.32	0.29	0.31
ViT+Weighted	0.75	0.34	0.33	0.34

5. ВИСНОВКИ

Підсумовуючи, можна сказати, що використання запропонованої модифікації модуля злиття ознак показала покращення у результатах за ключовими метриками з обома моделями вилучення візуальних ознак, але особливо хороші результати було досягнуто при використанні запропонованого модуля злиття ознак в поєднанні з візуальним трансформером, що свідчить про можливість використання наведеної конфігурації моделі у додатку для персоналізованої рекомендації хештегів у соціальній мережі інстаграм.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Q. Yang, G. Wu, Y. Li, R. Li, X. Gu, H. Deng, and J. Wu, “AMNN Attention-based multimodal neural network model for hashtag recommendation,” IEEE Transactions on Computational Social Systems, vol. 7, no. 3, pp. 768–779, 2020. DOI: https://dx.doi.org/10.1109/TCSS.2020.2986778

2. Q. Zhang, J. Wang, H. Huang, X. Huang, and Y. Gong, “Hashtag recommendation for multimodal microblog using co-attention network.” in IJCAI, 2017, pp. 3420–3426. DOI: http://dx.doi.org/10.24963/ijcai.2017/478

3. S. Zhang, Y. Yao, F. Xu, H. Tong, X. Yan, and J. Lu, “Hashtag recommendation for photo sharing services,” in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, 2019, pp. 5805–5812. DOI: http://dx.doi.org/10.1609/aaai.v33i01.33015805

4. Y. -C. Chen, K. -T. Lai, D. Liu, and M. -S. Chen, “Tagnet: Triplet-attention graph networks for hashtag recommendation,” IEEE Transactions on Circuits and Systems for Video Technology, 2021. DOI: https://dx.doi.org/10.1109/TCSVT.2021.3074599

5. Bansal, Shubhi & Gowda, Kushaan & Kumar, Nagendra. (2022). A Hybrid Deep Neural Network for Multimodal Personalized Hashtag Recommendation. IEEE Transactions on Computational Social Systems. DOI: http://dx.doi.org/10.1109/TCSS.2022.3184307.