

# НЕЙРОННА МЕРЕЖА ДЛЯ ДІАГНОСТУВАННЯ ЗАХВОРЮВАНЬ СИСТЕМИ ДИХАННЯ НА БАЗІ СИГНАЛІВ ЕКГ

Скріпченко М.О.<sup>1</sup>, Кислий Р.В.<sup>2</sup>

Інститут прикладного системного аналізу, КПІ ім. Ігоря Сікорського,  
Київ, Україна

<sup>1</sup>kvrware@gmail.com, <sup>2</sup>codegeas65@gmail.com

Незважаючи на досить розвинену індустрію медицини в сучасному світі, задача діагностування захворювань системи дихання є дуже важливою. Величезна кількість людей страждає від респіраторних захворювань кожного року, таких як, наприклад, апное, і, частина людей, нажаль, помирає від несвоєчасного діагностування, навіть якщо воно відбувається досвідченим лікарем. Одним з рішень проблеми неправильного діагностування є аналіз результатів з допомогою машинного навчання. Нейронні мережі є одним з таких рішень, де, через процес навчання на попередньо зібраному наборі результатів, що називають датасетами, система може надавати діагнози на основі певних закономірностей [1, 2]. Власне огляд та покращення точності вже існуючих нейронних мереж даного характеру, і є основною задачею роботи.

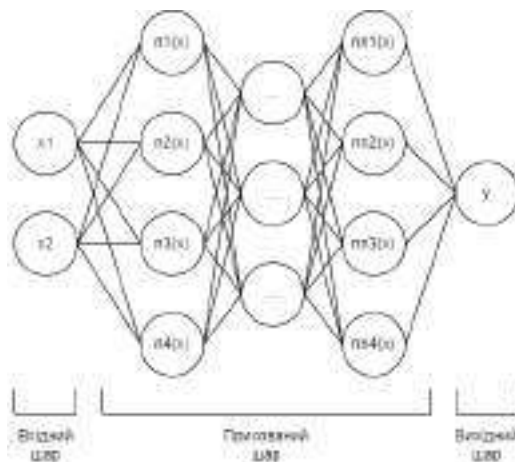


Рисунок 1. Узагальнена структура нейронних мереж

Нейронна мережа - це обчислювальна система, що базується на принципах роботи мозку людини. Тобто нейронні системи мають структуру, показану на рис.1, складаючись з нейронів та взаємозв'язків між ними, за допомогою яких відбувається навчання. З вхідного шару інформація передається на прихований шар, в якому нейрони роблять оцінку за допомогою функцій активації, і на вихідному шарі ми отримуємо результат, який представляє собою ймовірність того чи іншого діагнозу в залежності від задачі. Далі в роботу включається метод зворотного поширення (англ. backpropagation), що відповідає за коригування помилок шляхом поширення сигналів помилок в зворотному шляху, від виходу до входу, під час якого вага нейронів в прихованому шарі, що допомагає в збільшенні точності роботи мережі на наступному проході.

Задача, яка розглядається в роботі, відноситься до категорії аналізу часових рядів (англ. time series analysis), який представляє собою аналіз набору даних, зібраних на певному проміжку часу, що дозволяє виявлення значимих статистичних даних, а також подальшого

прогнозування результатів. Далі буде наведено варіанти нейронних мереж, що можуть бути використані для вирішення задач даного характеру.

Згорткові нейронні мережі (CNN) є класом глибоких мереж, що використовує спеціальні згорткові шари для більш ефективної та швидкої обробки вхідних даних, шляхом “згортки”, тобто зменшення початкових вхідних даних до певного формату меншого розміру, а також спеціальну функцію передобробки ConvNet. ConvNet складається з двох основних шарів, кожен з яких відповідає за згортку даних до більш оптимального розміру, але роблять це по-різному. Перший шар називається шар згортки (англ. convolutional layer), і він проводить згортання шляхом трансформації вхідних даних через поелементне множення на ядро шару, який являє собою сітку/матрицю ваг, які будуть тренуватися, що дозволяє визначати більш важливі елементи на наборі даних. Після чого, для більшої оптимізації розміру даних, використовується шар стягнення(англ. pooling layer), в якого є два варіанти: максимальне стягнення(вибір лише найбільш важливого елементу серед діапазону), або середнє стягнення(середнє значення елементів серед діапазону). Дані шари допомагають зменшити об’єм вхідних даних, при цьому маючи мінімальні втрати в точності прогнозування інформації.

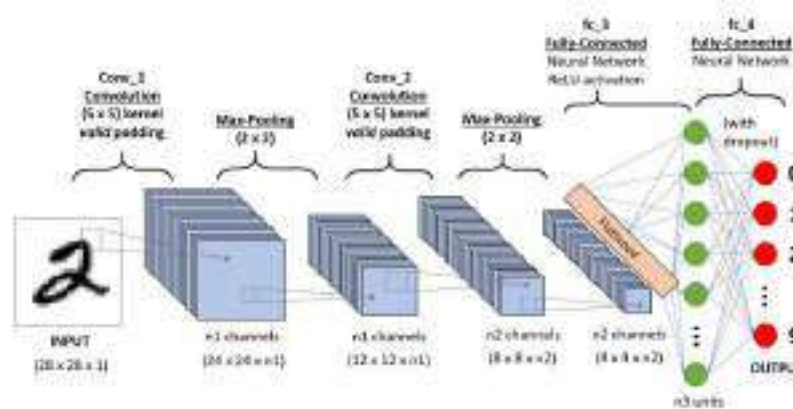


Рисунок 2. Приклад згорткової нейронної мережі

Рекурентні нейронні мережі також є гарною альтернативою при аналізі часових рядів. Основна ідея даної архітектури мереж базується на навчанні шляхом передачі інформації з попереднього стану мережі до наступного, що відкриває можливості більш точного прогнозування кожного наступного стану. Звичайні рекурентні нейронні мережі мають певні проблеми коли розрив між релевантно пов’язаними даними є досить великим, тому одним з найбільших рішень є архітектура довгої короткочасної пам’яті (англ. long short-term memory, LSTM). Основним компонентом даних мереж є стан клітини, яка на рис.3 є лінією, що проходить через верхню частину клітини. Вона зберігає інформацію про те, що мережа вважає важливим результатом попередніх результатів, значення яких можуть знаходитись в діапазоні [0,1], де 0 означає “видалити все”, а 1 - “залишити все”, що означає, що інформація може видалятися, якщо вона перестає бути релевантною, і цей процес контролюється завдяки фільтрам, які дозволяють вирішувати зміни в стані клітини. Спочатку інформація проходить через шар фільтра забуття(англ. forget gate layer), на якому ми змінюємо кожне значення стану клітини відносно результату роботи попередньої клітини та нових вхідних даних, що дозволяє нам відсіяти менш релевантні дані. Наступним кроком є вирішення, яка нова інформація буде зберігатись в стані клітини. За це відповідає шар вхідного фільтра(англ. input gate layer), що визначає які значення потрібно оновити, для яких шар функції активації будує вектор значень, які можна занести в новий стан клітини. Після цього, ми вносимо результат роботи фільтрів на стан клітини шляхом векторного множення значень

шару фільтра забуття на стан попередньої клітини, а потім додавання нових значень станів, які ми отримали з шару вхідного фільтра та шару функції активації. В кінці кінців, ми використовуємо шар вихідного фільтра(англ. output gate layer), в спільній роботі з шаром функції активації, які відсортовують, яку інформацію ми повинні вивести. Дана архітектура мережі дозволяє нам більш контрольований метод зберігання важливих закономірностей, та відкидання випадкових повторень.

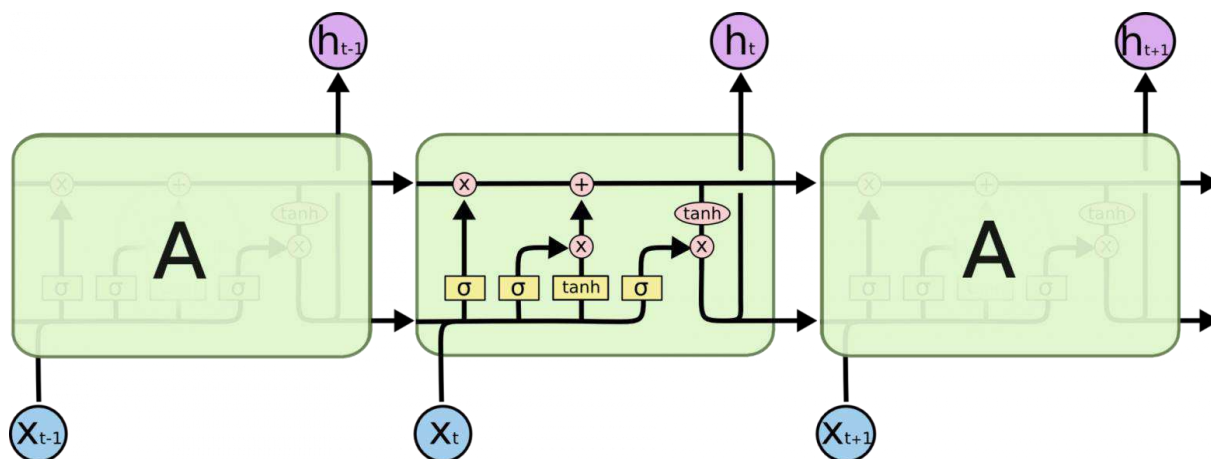


Рисунок 3. Базова структура LSTM-мережі

Також для вирішення таких задач можна використовувати статистичні методи навчання, такі, як, наприклад, алгоритм lightgbm. Даний метод базується на алгоритмі градієнтного бустингу, що відноситься до ансамлевих алгоритмів машинного навчання. Його ідея полягає в побудові ансамблей на основі моделей дерева рішень, які додаються в ансамбль по одному, і навчаються для виправлення помилок прогнозування, що були допущені попередніми моделями. Моделі власне навчаються з використанням будь-якої випадково обраної диференційованої функції витрат та алгоритма оптимізації градієнтного спуску. Lightgbm в свою чергу додає дві ключові ідеї: градієнтна одностороння вибірка(англ. Gradient One Side Sampling, GOSS), що використовується для виключення екземплярів з невеликим градієнтом, щоб віддати пріоритет екземплярам з більшим градієнтом, які матимуть більше значення на кінцеву точність; а також поєднання взаємовиключних ознак(англ. Exclusive Feature Bundling, EFB), яке поєднує розрідженні взаємовиключні ознаки, для зменшення загальної кількості ознак. Введення цих двох концепцій дозволяє отримати приріст в часі навчання алгоритму в 20 разів.

Тепер, маючи узагальнене розуміння нейронних мереж, а також різних можливостей їх реалізації, можна виділити наступні проблеми, поставленні в моїй дипломній роботі:

- дослідження впливу структури нейронної мережі, а тобто кількості проміжних шарів, типу зв'язків між ними, створеної на кафедрі, на точність і використання ЕКГ в діагностуванні;
- покращення функціонування діагностичної системи за рахунок застосування гібридних і нечітких компонентів нейронних мереж (хоч би теоретично);
- використання публічно доступних датасетів ЕКГ хворих і здорових пацієнтів, що містять тисячі записів з десятками атрибутів та призначені для визначення захворювання людини [3, 4].

Тобто запропонованим вирішенням поставленої задачі буде знаходження більш ефективних варіантів імплементації нейронної мережі для отримання точніших результатів.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Automated Prediction of Apnea and Hypopnea, Using a LAMSTAR Artificial Neural Network - Waxman, Jonathan A.; Graupe, Daniel; Carley, David W. Посилання на онлайн ресурс - <https://www.atsjournals.org/doi/10.1164/rccm.200907-1146OC>
2. Blind adaptive filtering for non-invasive extraction of the fetal electrocardiogram and its non-stationarities - D Graupe, M H Graupe. Посилання на онлайн ресурс - <https://journals.sagepub.com/doi/10.1243/09544119JEIM417>
3. MIT-BIH Arrhythmia Database - Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. Посилання на онлайн ресурс - <https://www.physionet.org/content/mitdb/1.0.0/>
4. PTB Diagnostic ECG Database - Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. Посилання на онлайн ресурс - <https://www.physionet.org/content/ptbdb/1.0.0/>

# АЛГОРИТМ РОЗРАХУНКУ ТОЧОК ДЛЯ ЗАХВАТУ ОБ'ЄКТА МАНІПУЛЯТОРОМ НА ОСНОВІ АПРОКСИМАЦІЇ 3D МОДЕЛІ ОБ'ЄКТА ПРОСТОРОВИМИ ПРИМІТИВАМИ

Харченко К.В.<sup>1</sup>, Вовк І.С.<sup>2</sup>

Інститут прикладного системного аналізу, КПІ ім. Ігоря Сікорського,  
Київ, Україна

<sup>1</sup>konst1970@gmail.com, <sup>2</sup>i.s.v.vovk@gmail.com

Задача захвату об'єкта маніпулятором є невід'ємною частиною процесу виконання роботом різноманітних завдань. Наприклад, широко використовується для автоматизації у виробничому середовищі. Зазвичай, це попередньо запрограмовані дії для одного визначеного типу маніпулятора та суворо обмеженого переліку об'єктів або їх типів. Проте, з часом сфера застосування робототехніки стрімко зростає і з'являється потреба у виконанні більш загальних, комплексних задач. Однією з яких є взаємодія зі заздалегідь невизначеною множиною об'єктів. У даній роботі розглядається взаємодія з твердими тілами.

Задача захвату невідомого об'єкта довільної форми, розміру та матеріалу є об'єктом досліджень. Одним з найбільш розповсюджених підходів є використання глибокого навчання. Наприклад у науковій роботі [1] для розрахунку захвату предмета маніпулятором з п'ятьма стрижнями пропонується використовувати глибокі згорткові нейронні мережі. Розрахунки проводяться на датасеті, отриманому за допомогою однієї RGB камери та сенсору глибини. У роботі [2] використовується метод глибокого навчання з підкріпленням. Вхідні дані отримуються з двох камер: закріпленої на основній частині робота та маніпуляторі. Захват виконується маніпулятором з двома паралельними стрижнями. Основним недоліком підходів, які потребують навчання є потреба у зборі великого об'єму даних для навчання та валідації. Такі датасети можуть бути синтезовані або зібрані дослідниками вручну. Наприклад, у роботі [3] дані для навчання моделі були отримані шляхом виконання дослідниками ряду мануальних дій з маніпулятором. Для збору датасету був сконструйований додатковий переносний сетап з аналогічним типом маніпулятора і можливістю виконувати захват предмета по натиску на кнопку. У будь-якому випадку алгоритми, основою яких є глибоке навчання, будуть ефективними лише на класах предметів, які були включені в навчальну вибірку, та є специфічними для типу маніпулятора для якого виконувалось навчання.

Іншим підходом до проблеми захвату об'єкта маніпулятором є використання технік комп'ютерного зору. У роботі [4] використовується часткове просторове зображення об'єкта, створене за допомогою сенсору глибини. Алгоритм захоплення предмета базується на математичній моделі маніпулятора (в роботі використовувався С-подібний маніпулятор). Такий підхід є більш універсальним, оскільки не виділяє якийсь окремий клас предметів. Проте, якщо модель маніпулятора складніша, ніж представлена у роботі, об'єм розрахунків збільшується, що матиме вплив на час виконання захвату.

У даній роботі запропоновано алгоритм розрахунку оптимальних точок для захвату об'єкта маніпулятором, що базується на апроксимації 3D моделі об'єкта просторовими примітивами. Модель об'єкта реконструюється з часткових просторових зображень об'єкта у процесі взаємодії з ним людини. Для отримання яких використовується RGB камера та сенсор глибини, що закріплені стаціонарно.

Основні кроки алгоритму зображені на рис. 1.

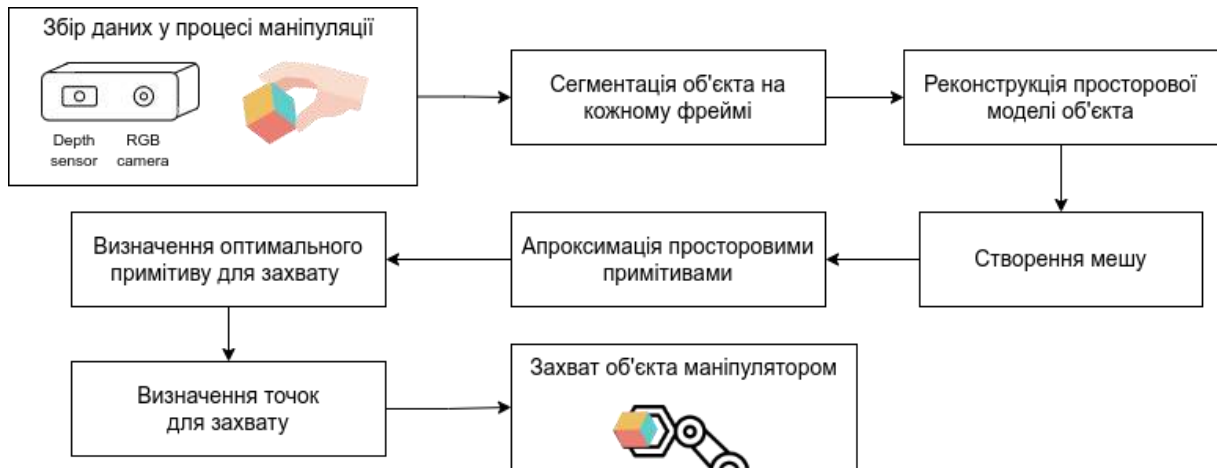


Рисунок 1. Блок-схема алгоритму визначення геометрії об'єкта шляхом взаємодії з ним, для задачі захвату маніпулятором

Алгоритм включає в себе наступні кроки:

- Збір даних у процесі маніпуляції людиною об'єктом у руках.
- Сегментація фону, рук та предмета. На цьому етапі розраховуються часткові просторові зображення предмета у форматі хмари точок.
- Реконструкція просторової моделі об'єкта з даних, отриманих на попередньому етапі. Теселяція хмари точок для отримання полігональної моделі(мешу).
- Апроксимація моделі просторовими примітивами, такими як паралелепіпед, сфера, конус, циліндр. Ці примітиви є простими для захвату будь-яким типом маніпулятора.
- Визначення оптимального для захвату примітиву та визначення точок захвату.

Запропонований у даній роботі підхід є універсальними для роботи з будь-яким класом об'єктів для захвату. Може бути використаним для будь-якого типу маніпуляторів, для яких способи захвату просторових примітивів (паралелепіпед, конус, сфера, циліндр) є визначеними. Запропонований алгоритм є детермінованим, а отже не потребує збору великої кількості даних для навчання.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Philipp Schmidt, Nikolaus Vahrenkamp, Mirko Wachter, Tamim Asfour “Grasping of Unknown Objects using Deep Convolutional Neural Networks based on Depth Images”, 2018
2. Shirin Joshi, Sulabh Kumra, Ferat Sahin “Robotic Grasping using Deep Reinforcement Learning” , 2020
3. Shuran Song, Andy Zeng, Johnny Lee, Thomas Funkhouser “Grasping in the Wild: Learning 6DoF Closed-Loop Grasping from Low-Cost Demonstrations”, 2020
4. Qujiang Leia, Guangming Chen, Jonathan Meijer, Martijn Wisse “A novel algorithm for fast grasping of unknown objects using C-shape configuration”, 2017

# РОЗРОБКА ДОДАТКУ ДЛЯ МОДЕЛЮВАННЯ ТА АНАЛІЗУ ЕЛЕКТРИЧНИХ КІЛ

Харченко К.В.<sup>1</sup>, Тараненко А.В.<sup>2</sup>

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

<sup>2</sup>tatanenko.and@gmail.com

**Метою дослідження є вивчення методів представлення елементів електричного кола та побудови кола на ЕОМ та методів аналізу електричного кола. Порівняння методів побудови моделі електричного кола. У доповіді описана актуальність теми та представлено опис причин для розробки додатку та результати тестової реалізації. Результатом роботи є програма, що приймає на вхід елементи електричного кола та їх параметри і на повертає характеристики утвореної схеми.**

**Ключові слова:** представлення електричного кола на ЕОМ, задача моделювання, аналіз електричного кола, додаток.

## 1. ВСТУП

Метою моделювання електронної схеми є отримання її математичної моделі і подальшого аналізу працездатності електронного пристрою на ЕОМ. Задачі моделювання електричних кіл розв'язуються в 2 етапи. Перший етап полягає у складанні рівнянь рівноваги для електричного кола. Другий етап полягає у вирішенні цих рівнянь за допомогою чисельних методів.

В області математичного моделювання динамічних систем можна виділити два основних напрямки. Одне з них орієнтовано на застосування спеціалізованих професійних прикладних програм з автоматизованого моделювання. Інший напрямок пов'язано з так званим «ручним» моделюванням, освоювати яке необхідно на стадії навчання студентів основам спеціальності і воно ефективне, так як математичне моделювання саме по собі передбачає більш глибоке вивчення фізичних процесів, що лежать в основі функціонування фізичних систем і математичного апарату для їх опису, що в сукупності сприяє ефективному навчанню в технічному університеті.

## 2. МЕТОДИ ПРЕДСТАВЛЕННЯ ЛІНІЙНИХ ЕЛЕМЕНТІВ ЕЛЕКТРИЧНОГО КОЛА ТА МОДЕЛІ СХЕМИ

Математичною моделлю реальної схеми називають сукупність співвідношень (формул, рівнянь, логічних виразів), що визначають характеристики стану системи залежно від її параметрів, початкових умов, вхідних сигналів (включаючи випадкові фактори) і часу.

У моделях розрізняють дані трьох типів:

- дані про елементи об'єкта моделювання;
- дані про властивості об'єкта й елементів;
- дані про співвідношення між елементами й об'єктом.

Абстрагування об'єкта проводять як за глибиною структурування (ієрархічна система, система елементів, нероздільний об'єкт), так і за ступенем абстрагування елементів і властивостями об'єкта (структурний, логічний та кількісний рівні).

На структурному рівні моделюється структура об'єкта на найнижчому рівні його структурування, коли математичну модель подано у вигляді множини, властивості та параметри елементів якої описані через функціональні зв'язки, співвідношення порядку, суміжності, сполучення. При цьому використовують апарат теорії множин і графів, теорії масового обслуговування тощо.

На логічному рівні моделювання кожній множині, булевій матриці бінарних співвідношень або структурному графові відповідають набори логічних співвідношень і змінних, які відображають причинно-наслідкові зв'язки. У проектуванні використовується апарат математичної логіки.

На кількісному рівні кожному елементові множини, булевій матриці або логічній змінній ставиться у відповідність алгебрична чи інша кількісна змінна, а логічні співвідношення переходять у кількісні співвідношення: рівняння, нерівності тощо.

Моделювання на кількісному рівні відображає функціональні, енергетичні, речовинні та просторові зв'язки. Ці зв'язки зазвичай описуються просторово-часовими співвідношеннями і визначаються через звичайні диференціальні рівняння або рівняння у частинних похідних.

Реактивними називають компоненти, до рівнянь яких входять похідні за часом від схемних змінних. Реактивними, зокрема, є звичайні лінійні ємності та індуктивності:

$$i_C = C \frac{du_C}{dt} \quad (1)$$

$$u_L = C \frac{di_L}{dt} \quad (2)$$

Враховуючи специфіку чисельних методів, що застосовуються для аналізу кола, для похідної за часом можна застосувати апроксимацію:

$$\frac{dy_n}{dt} = \frac{\alpha_0}{h} y_n - \frac{1}{h} \sum_{i=1}^K \alpha_i y_{n-i} \quad (3)$$

де  $K$  - порядок методу інтегрування;  $\alpha_0, \dots, \alpha_{K-1}$  - коефіцієнти апроксимаційного поліному;  $h$  - крок інтегрування,  $n$  - індекс кроків за часом.

Використовуючи наведене співвідношення, компонентне рівняння для ємності можна записати у вигляді:

$$i_{Cn} = \frac{\alpha_0}{h} C u_{Cn} - \frac{C}{h} \sum_{i=1}^{K-1} \alpha_i u_{C(n-i)} \quad (4)$$

або

$$i_{Cn} = G_{Cn} u_{Cn} + J_{Cn} \quad (5)$$

де

$$G_{Cn} = \frac{\alpha_0 C}{h}; \quad J_{Cn} = -\frac{C}{h} \sum_{i=1}^{K-1} \alpha_i u_{C(n-i)} \quad (6)$$

Таким чином, в межах даного кроку за часом можна формально представити ємність як паралельне з'єднання лінійної провідності  $G_{Cn}$  і джерела струму  $J_{Cn}$ , величина якого залежить від стану елемента в попередні моменти часу (рис. 1).

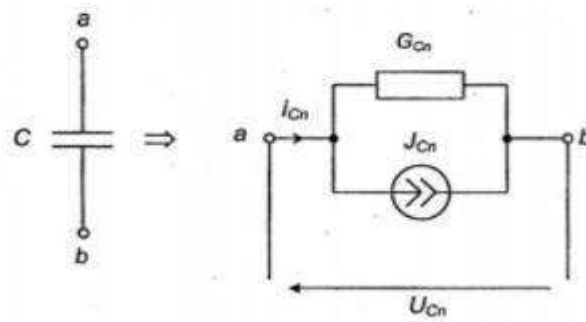


Рисунок 1. Алгебраїзація ємності

Аналогічно, компонентне рівняння індуктивності можна представити у вигляді:

$$u_{Ln} = \frac{\alpha_0}{h} L i_{Ln} - \frac{L}{h} \sum_{i=1}^{K-1} \alpha_i i_{L(n-i)} \quad (7)$$

або

$$u_{Ln} = R_{Ln} i_{Ln} + E_{Ln} \quad (8)$$

де

$$R_{Ln} = \frac{\alpha_0 L}{h}; \quad E_{Ln} = -\frac{L}{h} \sum_{i=1}^{K-1} \alpha_i i_{L(n-i)} \quad (9)$$

Таким чином, в межах даного кроку за часом можна формально представити індуктивність як послідовне з'єднання лінійного опору  $R_{Ln}$  і джерела напруги  $E_{Ln}$ , величина якого залежить від стану елемента в попередні моменти часу (рис. 2).

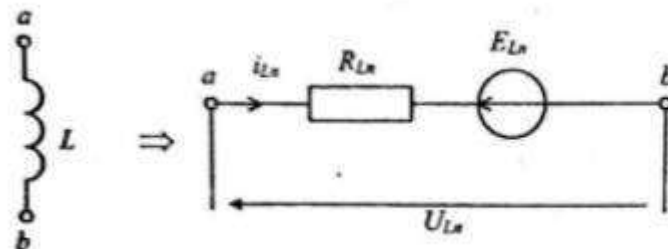


Рисунок 2. Алгебраїзація індуктивності

В практиці моделювання досить часто доводиться розглядати компоненти, властивості яких описуються нелінійними рівняннями виду  $y = f(x)$ , де  $y$  — струм або напруга на компоненті,  $x$  — вектор схемних змінних,  $f(x)$  — деяка нелінійна залежність, що з огляду на застосовувані чисельні методи аналізу вважається неперервною та диференційованою по всіх змінних. Враховуючи, що на кожній ітерації чисельного аналізу схемні змінні зазнають малих прирощень, нелінійну залежність  $f(x)$  можна лінеаризувати в околі поточних значень вектора невідомих:

$$y^{m+1} = f(x^m) + \sum_{i=1}^N \frac{\partial f}{\partial x_i} (x_i^{m+1} - x_i^m) \quad (10)$$

де  $m$  — індекс ітерацій. В загальному випадку ми вважаємо, що величина  $y$  залежить від усіх схемних змінних, однак на практиці зазвичай вона явно залежить лише від невеликої підмножини їх. Похідна  $\partial f / \partial x_i$  визначається в точці  $x^m$  і є константою в межах однієї

ітерації. На наступній ітерації її значення переобчислюється відповідно до поточних значень схемних змінних.

Останнє співвідношення можна переписати у вигляді:

$$y^{m+1} = \sum_{i=1}^N \frac{\partial f}{\partial x_i}(x_i^{m+1}) + (f x^m - \sum_{i=1}^N \frac{\partial f}{\partial x_i}(x_i^m)) \quad (11)$$

Отже, нелінійний компонент в межах даної ітерації можна представити як комбінацію лінійного компонента типу провідності або опору і джерела струму або напруги, величина змінної якого на даній ітерації є відомою.

В результаті застосування операцій алгебраїзації та лінеаризації вдається формально звести початкову схему до резистивного кола із залежними та незалежними джерелами.

### 3. РЕЗУЛЬТАТИ РОЗРОБКИ ДОДАТКУ

Завданням даної роботи є створення додатку для моделювання та аналізу електричних схем. Для цього було обрано мову програмування Python, тому що це скриптова об'єктно-орієнтована мова, що має величезну кількість додаткових модулів (бібліотек) для спрощення процесу написання коду даної програми. Також будуть обрані такі бібліотеки Python: NumPy та Matplotlib.

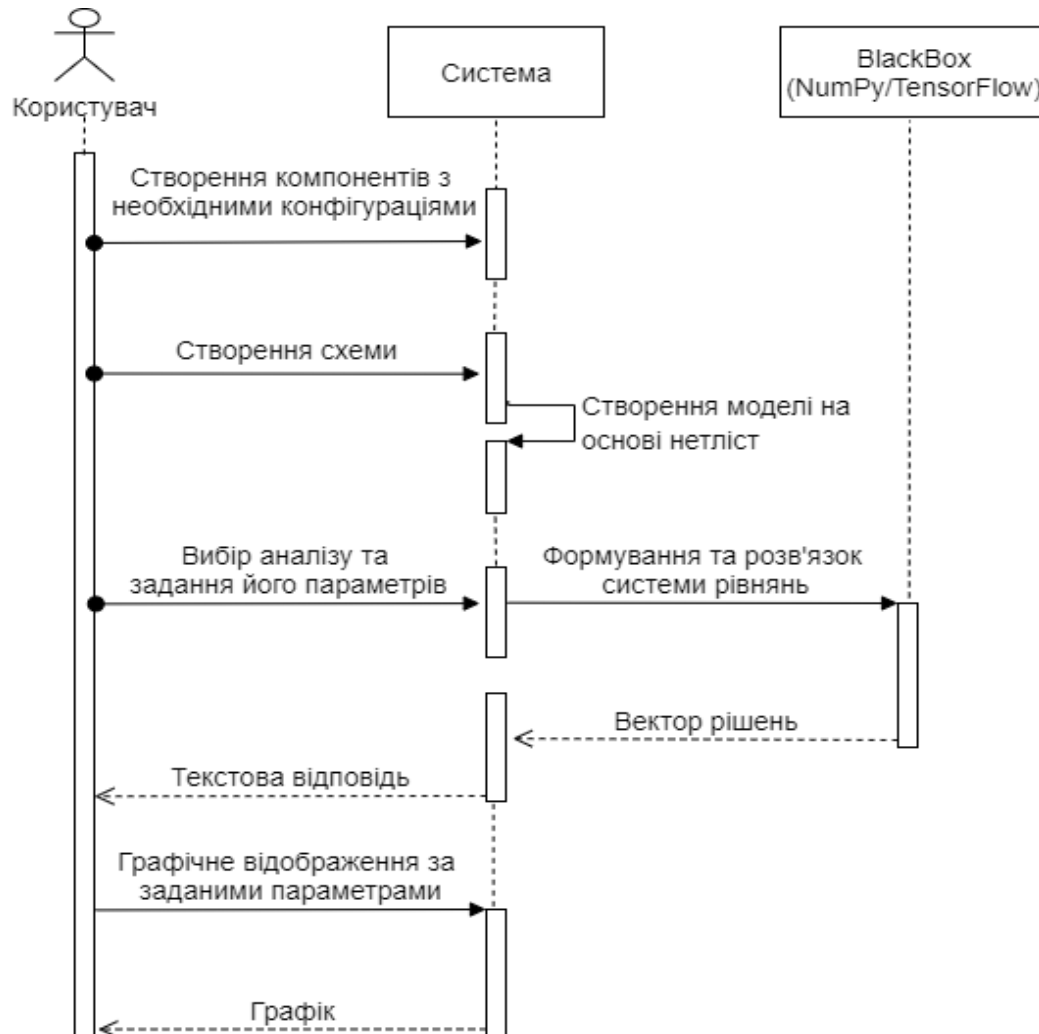


Рисунок 3. Діаграма послідовності роботи програми

Зобразимо детальний опис (Рис. 3) логіки можливого сеансу використання даної програми. Користувач створює компоненти необхідні йому для моделювання. Потім необхідно створити об'єкт схеми Circuit та передати йому список вузлів для формування математичної моделі. При виклику методу для аналізу схеми користувачу буде надано рішення рівнянь рівноваги для даної моделі у вигляді текстової та графічної відповіді.

У додатку присутній базовий набір лінійних елементів та нелінійні елементи, можливість вибору методу побудови рівнянь рівноваги для представлення електричного кола та декілька видів аналізу електричної схеми.

#### **4. ВИСНОВКИ**

На сьогоднішній день існують потужні спеціалізовані програми для моделювання динамічних систем, такі як ALLTED та PSPICE. Даний додаток не має на меті конкурувати з ними, але може стати корисним інструментом для навчання математичного моделювання та більш детальному вивченню фізичних процесів пов'язаних з ним, з можливістю переглядати кожен крок від побудови систем рівнянь за обраною моделлю до аналізу побудованої схеми.

#### **ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. Л.О. Чуа, Пен-Мин Лин Машинний аналіз електронних схем
2. Петренко Анатолій Іванович. Конспект лекцій до курсу „МОДЕЛЮВАННЯ СИСТЕМ” // Київ-2020. Ст.3-5, ст.27-28.
3. Петренко, Чкалов, Кірюша. «МОДЕЛЮВАННЯ СКЛАДНИХ СИСТЕМ» // Київ-2015. ст. 6-18.
4. [https://kpi.ua/web\\_allted](https://kpi.ua/web_allted)
5. <https://www.pspice.com/>

# СИСТЕМА ВІДСТЕЖЕННЯ ПОГЛЯДУ В РЕАЛЬНОМУ ЧАСІ ДЛЯ ВЗАЄМОДІЇ З КОМП'ЮТЕРОМ НА ОСНОВІ МЕТОДІВ ГЛИБОКОГО НАВЧАННЯ

Гончарук О.П.<sup>1</sup>, Недашківська Н.І.<sup>2</sup>

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

<sup>1</sup>honcharuk.oleksandr@lil.kpi.ua,

<sup>2</sup>nedashkovskaya.nadezhda@lil.kpi.ua [0000-0002-8277-3095]

**Різноманіття кількості способів вводу даних та покращення їх точності збільшує продуктивність та зручність взаємодії людини та будь-якої системи. У випадках коли доступ до звичайних пристроїв вводу відсутній, обмежений або неможливий, то їх доповненням або заміником можуть стати системи, що дозволяють керувати пристроєм за допомогою погляду. Метою роботи є вдосконалення методів відстеження погляду користувача, використовуючи методи глибокого навчання. Реалізовано систему відстеження погляду, використовуючи мікросервісну архітектуру.**

**Ключові слова:** відстеження погляду, глибоке навчання, згорткові нейронні мережі, механізм уваги, обробка зображень, комп'ютерний зір.

## 1. ВСТУП

Із розвитком технологій та їх активним впровадженням в повсякденне життя людини, все частіше виникає питання про ефективність взаємодії із цифровими пристроями. Такі параметри, як швидкість набору символів на клавіатурі або швидкість переміщення миші, мають конкретні межі. Також існують люди, які через фізіологічні причини не можуть повноцінно використовувати певні пристрої вводу. Враховуючи це, можливість управління пристроями за допомогою погляду є одним із найперспективніших напрямків досліджень та розробки.

Причин, щоб користуватися таким способом може бути доволі багато та вони можуть бути індивідуальними для конкретного користувача. Починаючи з особливостей будови пристрою (Наприклад, сервери в дата центрах), які не мають жодної периферії, закінчуючи виходом з ладу пристрою вводу, але гострою необхідністю використання цього пристрою.

Більшість технологій, що були створені для зчитування погляду використовують дорогі датчики. Вони надають достатньо точні виміри, але є надмірно дорогими для впровадження на глобальному ринку.

Однією із задач, що була поставлена при розробці даної системи, є надання можливості керувати пристроєм в реальному часі із присутньою камерою.

## 2. ОПИС НАБОРУ ДАНИХ

В межах цього дослідження було використано набір даних Gaze Capture [1], який містить 2.5 мільйони зображень від 1450-ти різних людей. Вказаний датасет містить інформацію, яка дозволяє ідентифікувати де саме на моніторі сконцентрований погляд, використовуючи лише зображення обличчя людини.

Мінливість даних дає загальну оцінку репрезентативності даних. Для цього в роботі було використано метод Dataset Characteristics [1], що оцінює позицію голови ( $h$ ) та напрям погляду ( $g$ ) для кожного зображення. Порівняння кінцевих розподілів величин  $h$  та  $g$  для обраного набору даних GazeCapture та інших наборів MPIIGaze [2] та TabletGaze [3] (Рис. 1) підтверджує варіативність положення голови та погляду у різних наборах, що безпосередньо впливає на якість навчання згорткових нейронних мереж.

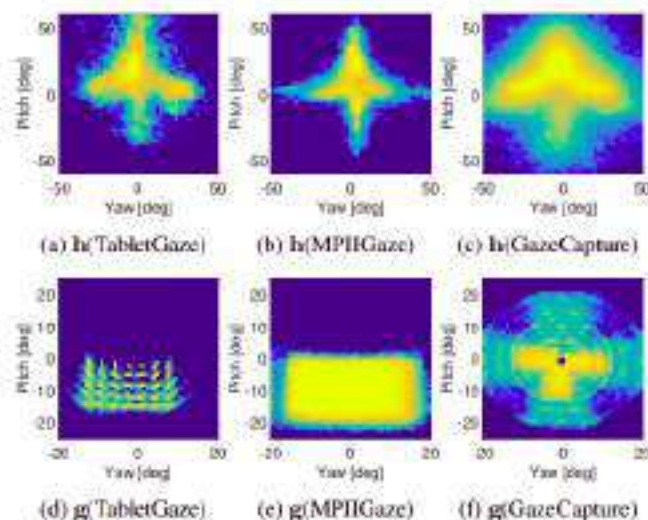


Рисунок 1. Розподіл положення голови і напрямку погляду в порівнянні з існуючими наборами даних [1]

### 3. АРХІТЕКТУРА НЕЙРОННОЇ МЕРЕЖІ ТА ОСОБЛИВОСТІ НАВЧАННЯ

У цій роботі розглядається підхід, що використовує інформацію представлену у вигляді зображення для передбачення координати погляду на моніторі персонального комп'ютера. Для отримання такого прогнозу пропонується модифікація відомої архітектури нейронної мережі [1] на основі згорткових нейронних мереж шляхом додавання механізму уваги [4].

На вхід моделі передається: зображення лица із відповідним його розміщенням на зображенні, іменується як маска лица; і додатково зображення лівого та правого ока відносно розміщення лица (голови). Комбінуючи цю інформацію, модель здатна визначати координати погляду користувача. Загальна архітектура нейронної мережі із відповідними розмірностями зображена на Рис. 2.

Варто звернути увагу, що зображення лівого і правого ока передаються на вхід мережі як окремі входи, хоча саме лице вже містить дану інформацію. Даний підхід було використано з метою збільшення роздільної здатності зображення, що спрощує процес ідентифікації напрямку погляду нейронною мережею.

Оскільки спостереження включають використання різних цифрових пристроїв: телефонів, планшетів та персональних комп'ютерів, необхідно було створити універсальну цільову зміну. В даному випадку - це двовимірний вектор з координатами  $x$ ,  $y$  відносно камери, які приймають значення в сантиметрах.

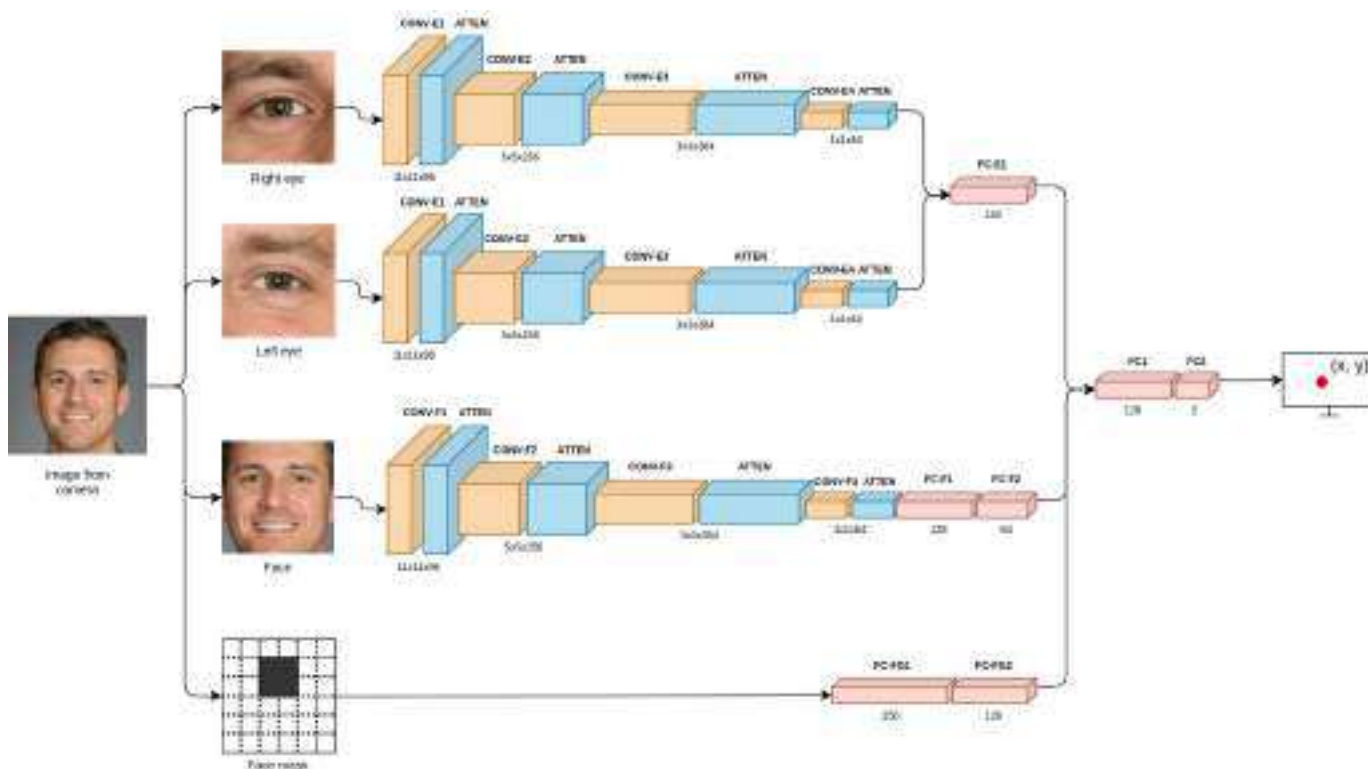


Рисунок 2. Високорівнева архітектура нейронної мережі

Датасет GazeCapture містить 2 445 504 зображень, серед яких лише 1 490 059 використовуються для навчання, валідації та тестування. А саме, спостереження отримані від 1471 людини використовувались для навчання нейронної моделі, зображення отримані від 50 людей – для валідації, і зображення від 150 людей – для перевірки кінцевої якості моделі.

Враховуючи, що розміри кожного зображення відрізняються, було проведено їх трансформацію до єдиного розміру 224 на 224 пікселів. Архітектуру нейронної мережі було реалізовано за допомогою фреймворку Torch для глибокого навчання [5].

Пропонована на Рис. 2 архітектура моделі розбита на функціональні блоки (Табл. 1 і 2).

У Табл. 1 у позначенні Conv2d(a/b/c/d) використано: *a* - кількість фільтрів, *b* - розмір ядра згортки, *c* - крок зміщення, *d* - padding; після кожного згорткового шару Conv2d використовується активаційна функція ReLU [6]; у позначенні об'єднуючого шару MaxPool2d (a/b): *a* - розмір вікна, *b* - крок зміщення; у BatchNorm2d (*a*) під *a* розуміється кількість фільтрів; у шарі MultiHead (a/b): *a* - кількість фільтрів, *b* - кількість шарів уваги.

У позначенні Linear(a/b): *a* - кількість вхідних нейронів, *b* - кількість вихідних нейронів. У кожному повнозв'язному шарі застосовується активаційна функція ReLU.

Таким чином, на виході нейронної мережі отримуємо два значення, які відповідають за координату погляду *x* та *y* відповідно.

Оскільки дана задача відстеження погляду формулюється як регресійна задача, то в якості функції втрат, що оптимізується, було обрано функцію похибки Губера [7]:

$$l_n = \begin{cases} 0.5 x_n - y_n^2, & \text{if } x_n - y_n < \text{delta} \\ \text{delta} * x_n - y_n - 0.5 * \text{delta}, & \text{otherwise} \end{cases}$$

Зазначена функція об'єднує переваги Манхетенської метрики (L1), яка менш чутлива до викидів, а також Евклідової метрики (L2), яка забезпечує згладжування в околі нуля.

Таблиця 1. Модуль обробки зображень

Номер шару	Складові модуля
1	Conv2d(96/11/4/0) MaxPool2d(3/2) BatchNorm2d(96) MultiHead(96/3)
2	Conv2d(256/5/1/2) MaxPool2d(3/2) BatchNorm2d(256) MultiHead(96/3)
3	Conv2d(384/3/1/1) Conv2d(64/1/1/0)

Таблиця 2. Повнозв'язні шари для агрегування даних

Номер шару	Eyes Module	Face	Grid	Aggregate Module
1	Linear(18432, 128)	Linear(9216, 128)	Linear(784, 256)	Linear(320, 128)
2		Linear(9216, 128)	Linear(256, 128)	Linear(128, 2)

В якості алгоритму навчання (optimizer) було вибрано Adam [8] із коефіцієнтом навчання  $learning\ rate = 0.0003$ , параметрами  $\beta_1 = 0.9$  та  $\beta_2 = 0.999$ . Додатково використовувався параметр  $weight\_decay = 0.0001$ , як аналог L2 регуляції із адаптацією коефіцієнта навчання.

Процес навчання зображено на Рис. 4. Батч містив 128 зображень, навчання проводилося протягом 100 епох. З Рис. 4 бачимо, що функція втрат, в даній задачі це функція похибки Губера (Huber Loss), стрімко зменшує своє значення на перших епохах навчання, і після сорокової епохи виходить на плато як на навчальній (Train), так і на валідаційній (Valid) підмножинах.

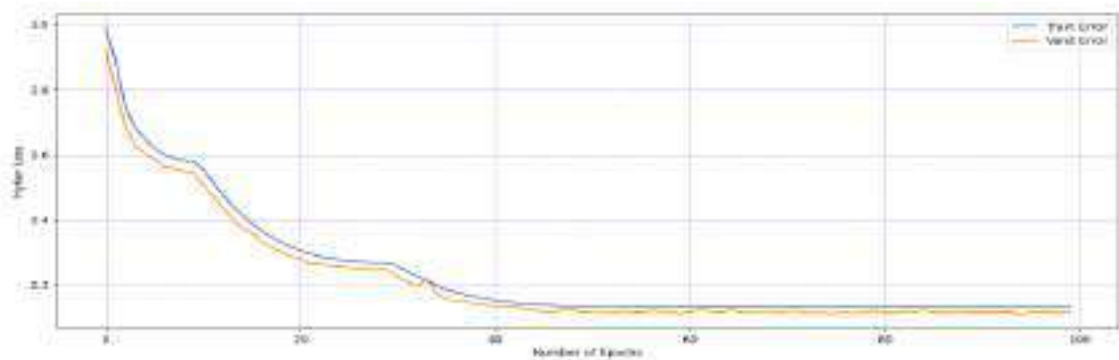


Рисунок 4. Прогрес навчання нейронної мережі

## 4. РЕЗУЛЬТАТИ

В Табл. 3 наведено значення показника точності MAE (mean absolute error) на валідаційній підмножині окремо для кожної координати прогнозованого цільового вектора. Для порівняння, в табл. 3 наведено значення для запропонованої моделі і для іншої відомої моделі iTracker [1] лише для планшетів без калібрування.

Таблиця 3. Порівняння отриманих результатів

Модель	Значення MAE, см	
	координата x	координата y
iTracker [1]	1.04	2.31
Пропонована модель	0.98	2.17

Як видно з Табл. 3, запропонована модель показала кращі результати порівняно з відомою iTracker: значення MAE є меншими на 5.769% та 6.061% для координати x та y, відповідно.

## 5. ВИСНОВКИ

В ході роботи було досліджено і встановлено інструменти прогнозування погляду користувача на основі зображення обличчя.

Запропоновано модифіковану модель глибокої нейронної мережі для відстеження погляду і взаємодії з комп'ютером в реальному часі на основі згорткових шарів із додаванням механізму уваги. Розроблено систему відстеження погляду у реальному часі з використанням фреймворку Torch Python.

В результаті порівняльного аналізу встановлено, що розроблена система за показником MAE (значення 0.98 см і 2.17 см для координат x і y), не поступається перед існуючою системою iTracker, у якої MAE сягають для координат x та y значення 1.04 см та 2.31 см, відповідно.

Перспективи подальших досліджень полягають у пришвидшенні роботи системи. Також одним із вдосконалень є додавання можливості розгортання частини системи у хмарі, що дозволить перенести ресурсоємні обчислення з клієнтської частини системи.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Eye Tracking for Everyone [Електронний ресурс] / [К. Kraffka, А. Khosla, Р. Kellnhofer та ін.]. – 2016. – Режим доступу до ресурсу: <https://gazecapture.csail.mit.edu/>.
2. Appearance-based gaze estimation in the wild. [Електронний ресурс] / Х.Zhang, Y. Sugano, М. Fritz, А. Bulling. – 2015. – Режим доступу до ресурсу: [https://openaccess.thecvf.com/content\\_cvpr\\_2015/papers/Zhang\\_Appearance-Based\\_Gaze\\_Estimation\\_2015\\_CVPR\\_paper.pdf](https://openaccess.thecvf.com/content_cvpr_2015/papers/Zhang_Appearance-Based_Gaze_Estimation_2015_CVPR_paper.pdf).
3. Huang Q. TabletGaze: Unconstrained Appearance-based Gaze Estimation in Mobile Tablets [Електронний ресурс] / Q. Huang, А. Veeraraghavan, А. Sabharwal. – 2015. – Режим доступу до ресурсу: <https://arxiv.org/abs/1508.01244>.
4. Islam K. Face Pyramid Vision Transformer [Електронний ресурс] / К. Islam, М. Zaigham, А. Mahmood. – 2022. – Режим доступу до ресурсу: <https://arxiv.org/abs/2210.11974>.
5. PyTorch Foundation [Електронний ресурс] – Режим доступу до ресурсу: <https://pytorch.org/foundation>.
6. Fred A. Deep Learning using Rectified Linear Units (ReLU) [Електронний ресурс] / Abien Fred. – 2018. – Режим доступу до ресурсу: <https://arxiv.org/abs/1803.08375>.
7. Gokcesu K. Generalized Huber Loss for Robust Learning and its Efficient Minimization for a Robust Statistics [Електронний ресурс] / К. Gokcesu, G. Hakan. – 2021. – Режим доступу до ресурсу: <https://arxiv.org/abs/2108.12627>.
8. Kingma D. Adam: A Method for Stochastic Optimization [Електронний ресурс] / D. Kingma, В. Jimmy. – 2014. – Режим доступу до ресурсу: <https://arxiv.org/abs/1412.6980>.

# ВИЗНАЧЕННЯ ЕМОЦІЙНОГО ЗАБАРВЛЕННЯ ГОЛОСУ ЗА ДОПОМОГОЮ ТЕХНІК DEEP LEARNING У РЕАЛЬНОМУ ЧАСІ

Загарницький Д.В.<sup>1</sup>, Шаповал Н.В.<sup>2</sup>

КПІ ім. Ігоря Сікорського, Київ, Україна

<sup>1</sup> dmitriy.zaharnitskiy@gmail.com, <sup>2</sup> shovgun@gmail.com

Метою дослідження є аналіз можливостей сучасних глибоких нейронних мереж для вирішення задач аналізу емоційного забарвлення голосу. За основу була взята глибока нейронна мережа AlexNet і за допомогою transfer learning вона була донавчена на розпізнавання емоцій на спектрограмах аудіо файлів, які були перетворені у RGB зображення. За дані було взято Berlin Emotional Speech (ЕМО-DB) датасет з якого були створенні відрізки даних довжиною до 1.04 секунди. У результаті ми отримали середню точність класифікації емоцій у 80.5%. Основне практичне значення роботи, це можливість удосконалення сервісів типу «людина-машина» для покращення їх взаємодії у відповідність до емоційного стану людини.

**Ключові слова:** емоційне забарвлення голосу, мережа AlexNet, transfer learning, RGB, голосові запити клієнтів.

## 1. ВСТУП

Розпізнавання емоцій голосу, або speech emotion recognition (SER) — це завдання розпізнавання емоційних аспектів мовлення незалежно від семантичного змісту. Хоча люди можуть ефективно виконувати це завдання як природну частину мовного спілкування, здатність виконувати це автоматично за допомогою програмованих пристроїв все ще є предметом досліджень.

Дослідження систем автоматичного розпізнавання емоцій спрямовані на створення ефективних методів виявлення емоцій у режимі реального часу користувачів мобільних телефонів, операторів колл-центру та клієнтів, водіїв автомобілів, пілотів та багатьох інших користувачів спілкування «людина-машина». Додавання емоцій до машин було визнано критичним фактором, що змушує машини виглядати та діяти схожими на людину [1].

Традиційно машинне навчання (ML) передбачає обчислення ознак (features) із необроблених даних (наприклад, мова, зображення, відео, ЕКГ, ЕЕГ). Ознаки використовуються для навчання моделі, яка вчиться класифікувати їх. Поширеною проблемою, з якою стикається цей підхід, є вибір цих ознак. Загалом невідомо, які ознаки можуть призвести до найбільш ефективної класифікації даних у різні категорії (або класи). Деяке розуміння можна отримати, протестувавши велику кількість різних ознак, об'єднавши різні ознаки в загальний вектор ознак або застосувавши різні методи вибору ознак. Якість створених вручну елементів може мати значний вплив на продуктивність класифікації.

Завдяки появі класифікаторів на базі глибоких нейронних мереж (DNN) було запропоновано елегантне рішення, яке обходить проблему оптимального вибору ознак. Ідея полягає у використанні мережі, яка приймає необроблені дані як вхідні дані та генерує мітку класу як вихідні дані. Немає потреби ні обирати вручну ознаки, ні визначати, які параметри є

оптимальними з точки зору класифікації. Ціною цього дуже зручного рішення є набагато більші вимоги до мічених зразків даних порівняно зі звичайними методами класифікації.

У багатьох випадках, і це включає SER, лише мінімальні дані доступні для цілей навчання. Як показано в цьому дослідженні, проблему обмежених навчальних даних значною мірою можна подолати підходом, відомим як трансферне навчання. Воно використовує існуючу мережу, попередньо навчену на великих даних, для вирішення загальної проблеми класифікації. Потім ця мережа додатково навчається з використанням невеликої кількості доступних даних для вирішення більш конкретного завдання.

Враховуючи, що на даний момент найпотужніші попередньо навчені нейронні мережі навчені для класифікації зображень, щоб застосувати ці мережі до проблеми SER, сигнал голосу потрібно перетворити у формат зображення [2]. Це дослідження описує кроки, пов'язані з переходом від голосу до зображення; у ньому пояснюються процедури навчання та тестування, а також умови, які необхідно виконати, щоб досягти розпізнавання емоцій у режимі реального часу з безперервної потокової мови.

## 2. МЕТОДИ ОБРОБКИ ТА ПІДГОТОВКИ ДАНИХ

Враховуючи, що доступні обчислювальні ресурси були обмежені, і була доступна лише невелика база даних емоційно позначених зразків мовлення, метою було визначити обчислювально ефективний підхід, який міг би працювати з невеликим набором навчальних даних. Для кожного блоку був розрахований масив спектрограм, який перетворювався у формат зображення RGB і передавався як вхідні дані до попередньо навченого CNN. Після відносно короткого навчання навчена CNN була готова класифікувати емоції з нерозміченої (потокової) мови, використовуючи той самий процес перетворення мови в зображення. У представлених тут експериментах продуктивність SER перевірялася з використанням двох різних частот дискретизації (16 і 8 кГц) і процедури компандування  $\mu$ -low. Система SER була реалізована за допомогою мови програмування Python та робочої станції із процесором AMD Ryzen 5 2600X, 3,6 ГГц, 32 ГБ оперативної пам'яті та NVIDIA GeForce 1060 3GB.

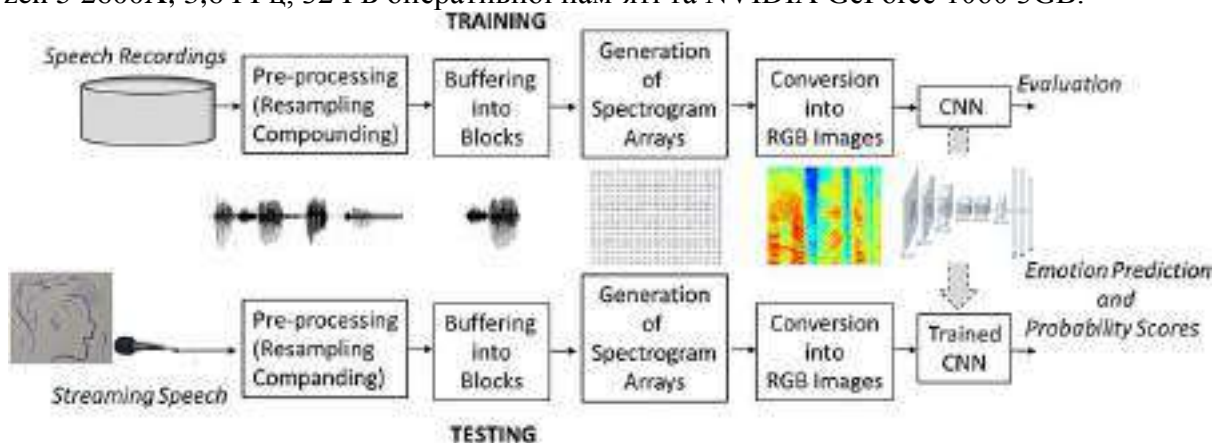


Рисунок 1. Огляд методу SER у реальному часі

У традиційних вузькосмугових системах передачі даних смуга пропускання мовного сигналу була обмежена, щоб збільшити швидкість передачі. У телефонії, наприклад, діапазон частот мови раніше обмежувався діапазоном від 300 Гц до 3,4 кГц. Цього було достатньо, щоб забезпечити базовий рівень розбірливості мови, але ціною високої якості голосу. Цілком ймовірно, що таке серйозне скорочення пропускну здатності призвело до значного зменшення емоційної інформації, що передається мовцями.

Щоб перевірити цю можливість, систему SER було навчено з двома різними частотами дискретизації: початкова 16 кГц відповідає широкій смузі пропускання 8 кГц, а зменшена частота дискретизації 8 кГц відповідає вузькій смузі частот 4 кГц.

Враховуючи оригінальні зразки мовлення  $x$ , компресія відрізків голосу  $F(x)$  була розрахована як

$$F(x) = \frac{\ln(1+\mu|x|)}{\log(1+\mu)^{\text{sgn}(x)}}$$

Тоді як реконструйовані мовні зразки  $\tilde{x}$  були розраховані як

$$\tilde{x} = F^{-1}(F(x)) = \text{sgn}(F(x)) \frac{((1+\mu)^{|F(x)|} - 1)}{\mu}$$

Значення параметра стиснення  $\mu$  було встановлено на 255 (стандарт у США та Японії) [3].

Потокове або записане мовлення було буферизовано в блоки по 1 с для проведення поблочної обробки. Між наступними блоками застосовувався короткий крок тривалістю 10 мс. Рівні амплітуди були нормалізовані в діапазоні від -1 до 1.

Процедура, яка використовується для генерації масивів спектрограм, проілюстрована на рисунку 2. Короткочасне перетворення Фур'є було виконано для кожного 1-секундного блоку мовних сигналів з використанням кадрів 16 мс, створених із застосуванням функції вікна Хеммінга зі зсувом у часі.

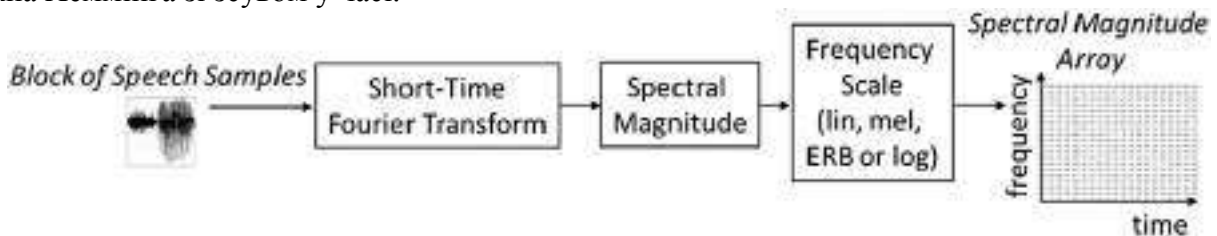


Рисунок 2. Генерація масивів спектрограм

Масиви спектральних величин  $257 \times 259$  дійсних чисел були перетворені у формат кольорового зображення RGB, представлений трьома масивами кольорових компонентів (рис. 3).

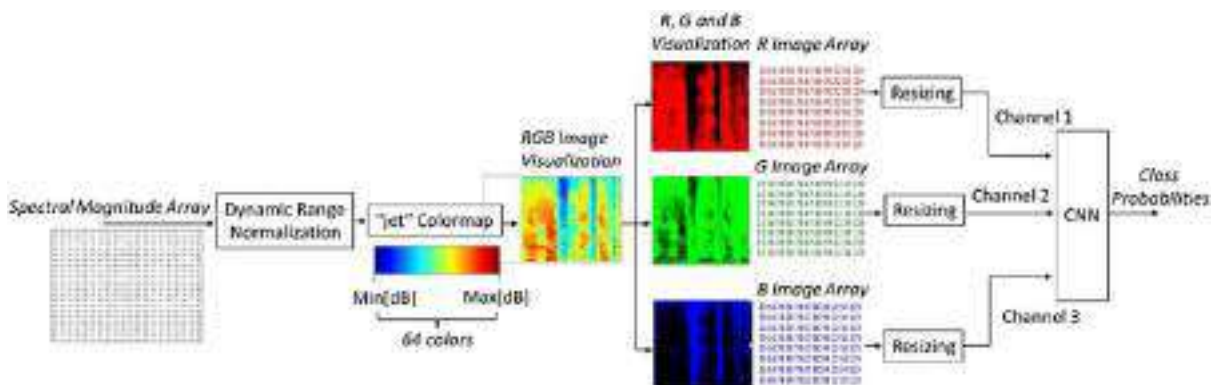


Рисунок 3. Перетворення масивів спектральних амплітуд у масиви зображень R, G та B.

### 3. МЕРЕЖА КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ ALEXNET

AlexNet — це згорткова нейронна мережа (CNN), представлена Крижевським та ін. (2012). Її було попередньо навчено на понад 1,2 мільйонах зображень із набору даних ImageNet Стенфордського університету, щоб розрізняти 1000 категорій об'єктів. Він складається з 3-канального вхідного шару, що дозволяє вводити три 2-вимірні масиви, кожен розміром  $256 \times 256$  пікселів. Після вхідного рівня йдуть п'ять згорткових шарів (Conv1–Conv5), кожен з яких має шар максимального об'єднання та нормалізації (рис. 4). Двовимірні вихідні об'єкти з останнього згорткового шару Conv5 перетворюються в одновимірні вектори та подаються в три повністю з'єднані шари (fc6–fc8). У той час як згорткові шари витягують характерні ознаки з вхідних даних, повністю пов'язані шари вивчають параметри моделі класифікації даних. Експоненціальна функція SoftMax відображає вихідні значення fc8 у нормалізований вектор дійсних значень, які потрапляють у діапазон  $[0,1]$  і в сумі дають 1. Ці значення надаються на вихідному рівні та представляють ймовірності кожного класу. Остаточна класифікаційна позначка присвоюється класу, який отримав найвищий бал вірогідності.

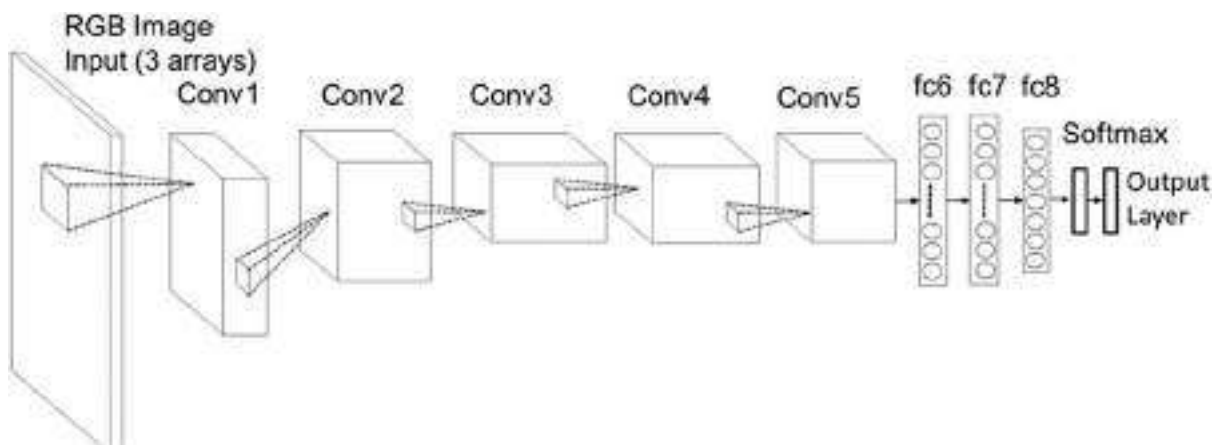


Рисунок 4. Структура AlexNet

Після адаптації до класифікації семи емоцій AlexNet було навчено (fine-tuned) на позначених даних. Оскільки мережа вже була попередньо навчена, процес навчання був набагато швидшим і “можливим” порівняно з тим, що потрібно було б під час навчання тієї самої структури мережі з нуля. Однак можливо, що за наявності необхідних ресурсів навчання з нуля могло б привести до кращих результатів. Хоча в останні роки AlexNet конкурували зі значно складнішими мережевими структурами [4].

У transfer learning процес навчання має на меті досягнути найвищого впливу навчання на кінцевих, повністю пов'язаних рівнях мережі, залишаючи попередні рівні майже недоторканими. Щоб досягти швидшого навчання в нових модифікованих шарах і повільнішого в старих шарах, початкова швидкість навчання (learning rate) була встановлена на невелике значення, а значення weight learning rate та bias learning rate були збільшені лише для кінцевих шарів.

### 4. ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

Усі експерименти адаптували техніку 5-кратної cross-validation з 80% розподілу даних для навчання AlexNet і 20% для тестування. Зразки даних тестування ніколи не використовувалися під час процедури навчання мережі. Експерименти не залежали від мовця та статі.

Таблиця 1. Інформація про набір даних

Емоція	Кількість зразків	Загальна тривалість (сек.)	Кількість згенерованих зображень
Злість	129	335	27220
Нудьга	79	220	18125
Огида	38	127	11010
Страх	55	123	5463
Задоволення	58	152	12400
Нейтральна	78	184	14590
Сум	53	210	18455
Загально	390	1207	111425

Метриками для оцінки виступили Accuracy, Precision, Recall та F-score.

Застосування SER в режимі реального часу було досягнуто шляхом поблочної обробки. Для кожного блоку була створена класифікаційна мітка, яка вказує на одну з семи категорій емоційного класу. Після навчання моделі були проведені наступні експерименти:

- Експеримент 1: частота дискретизації 16 кГц (смуга пропускання = 8 кГц);
- Експеримент 2: частота дискретизації 8 кГц (смуга пропускання = 4 кГц);
- Експеримент 3: частота дискретизації 16 кГц,  $\mu$ -low компандування;
- Експеримент 4: частота дискретизації 8 кГц,  $\mu$ -low компандування.

Таблиця 2. Результати експериментів

Експеримент	Weighted precision (%)	Weighted recall (%)	Weighted F-score (%)	Weighted accuracy (%)
1	80.3	80.8	79.6	80.5
2	76.8	78.0	76.1	76.8
3	75.8	77.1	75.3	75.8
4	73.7	76.1	73.1	73.7

Таблиця 3. Середній час обчислення в мілісекундах (мс)

Експеримент	Час інференсу	Створення ознак			Загальний час
		Створення масивів спектрограм	Перетворення у RGB зображення	Загальний час створення ознак	
1	18.7	8.0	3.6	11.6	30.3
2	18.6	4.7	3.6	8.3	26.9
3	18.4	7.5	3.6	11.1	29.5
4	18.5	4.6	3.6	8.2	26.7

#### 4. ВИСНОВОК

Підсумовуючи, ми виявили, що використання CNN для класифікації емоцій голосу дає точність у 80.5% і швидкість інференсу достатня для аналізу емоцій у реальному часі. Також було показано, що обидва фактори, зменшення смуги пропускання мовлення та впровадження процедури компандування мовлення  $\mu$ -low, мають шкідливий вплив на результати SER. За рахунок зменшення частоти дискретизації з 16 до 8 кГц (тобто, зменшення смуги пропускання з 8 до 4 кГц) спостерігалось невелике зниження середньої точності SER на (близько 3,3%). Процедура компандування зменшила результат на таку саму величину (приблизно на 3,8%), а сукупний вплив обох факторів призвело до зниження приблизно на 7% порівняно з базовими результатами. У всіх експериментальних випадках SER виконувався в режимі реального часу з емоційними мітками, що генерувалися кожні  $\sim 1,04$  с.

#### ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Андре, Е., Рем, М., Мінкер, В., і Бюлер, Д. «Наділення систем розмовного мовного діалогу емоційним інтелектом», у навчальному та дослідницькому семінарі з афективних діалогових систем, ADS 2004, ред. Е. Андре, Л. Дибк'єр, П. Хайстеркамп і В. Мінкер (Німеччина: Kloster Irsee), 2004. 178–187.
2. Столар, М. Н., Лех, М., Боля, Р. Б., і Скіннер, М. «Розпізнавання емоцій голосу у реальному часі за допомогою класифікації зображень RGB і transfer learning», 2017. 1–6.
3. Методи кодування сигналу. URL: <https://www.cisco.com/c/en/us/support/docs/voice/h323/8123-waveform-coding.html> (дата звернення: 15 листопада 2022 р.).
4. Жегеді, С., Луї, В., Джіа, І., Серманет, П., Рід, С., Ангелов, Д. та ін. «Поглиблюючись із CNN», у матеріалах Конференції IEEE з комп'ютерного бачення та розпізнавання образів, 2015. 1–9.
5. EmoDB Dataset. URL: <https://www.kaggle.com/datasets/piyushagni5/berlin-database-of-emotional-speech-emodb> (дата звернення: 10 листопада 2022 р.).

# ПРОГНОЗ ТРАЄКТОРІЇ УЧАСНИКІВ ДОРОЖНЬОГО РУХУ З ВИКОРИСТАННЯМ КІЛЬКОХ КАМЕР

Загній Є.В.<sup>1</sup>, Данилов В.Я.<sup>2</sup>

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

<sup>1</sup> egor.zagny@gmail.com,

<sup>2</sup> danilov1950@ukr.net [0000-0003-3389-3661]

**Для роботи автопілота машини або допоміжної системи водія непогано знати траєкторії руху об'єктів навколо автомобіля. Для прогнозування траєкторій об'єктів навколо авто потрібно на зображеннях з камер розпізнати об'єкти, оцінити відстані до них щоб визначити їх місцезнаходження і відслідковувати переміщення об'єктів у часі. Для цього потрібно застосувати цілий комплекс нейронних мереж, що і буде досліджено в даній роботі.**

**Ключові слова:** машинне навчання, коп'ютерний зір, розпізнавання об'єктів, оцінка відстаней, відслідковування об'єктів, прогноз траєкторій.

## 1. ВСТУП

За останні 10 років людство досягло значного успіху у галузі штучного інтелекту. Це стало можливим завдяки значному зростанню обчислюваних потужностей комп'ютерів, завдяки чому з'явилася можливість тренувати великі нейронні мережі. Штучний інтелект вже є частиною звичного життя людей. Одним з напрямком штучного інтелекту є створення автопілотів для машин. Створення автопілоту неможливо без розв'язання задачі розпізнавання машин і пішоходів навколо машини, знаходження відстаней до них, відслідковування переміщення цих об'єктів, побудови карти з висоти пташиного польоту з розміщенням на ній цих об'єктів, та прогнозуванням траєкторій сусідніх машин.

Метою роботи є дослідження й порівняння різних методів, моделей, структур нейронних мереж для вирішення кожної з підзадач: розпізнавання об'єктів навколо машини, знаходження відстаней до них, побудови виду з висоти пташиного польоту навколо машини, відслідковування переміщення кожного об'єкта та прогнозування траєкторій об'єктів.

## 2. ЕТАПИ ЗАДАЧІ ПРОГНОЗУВАННЯ ТРАЄКТОРІЙ

Для того щоб працювати з траєкторіями об'єктів, спочатку треба розпізнати об'єкти на зображеннях що видає камера, визначити їх місцезнаходження, та відслідкувати переміщення цих об'єктів відносно попередніх кадрів з камери. [1-3]

Нейронні мережі складаються з тіла та голови. Існують багато видів тіл і голів. Доволі часто тіло і голова мережі вибираються окремо. Тіло нейронної мережі вибирається в залежності від того яка швидкість і точність роботи нейронної мережі потрібна. Чим більше тіло тим менша швидкість роботи, але тим кращу точність зможе видати нейронна мережа. Існують такі види тіл як MobileNet, VGG, ResNet. В задачах комп'ютерного зору на вхід нейронних мереж в більшості випадків подається просто зображення. Проте вихід нейронної мережі залежить від задачі і може дуже сильно відрізнитися, тому в першу чергу нас буде цікавити голова нейронної мережі.

Задача розпізнавання об'єктів полягає у тому щоб знайти на зображенні об'єкти певного класу. Зазвичай кожен об'єкт буде позначатися прямокутником з координатами його розміщення на зображенні, а також ідентифікатор класу об'єкта.

Для задачі розпізнавання об'єктів існують такі голови: CenterNet, SSD, Yolo. Розглянемо центернетівську архітектуру.

Суть центернетівської архітектури полягає в тому, що нейронна мережа повертає для кожного класу об'єкта матрицю в якій вказані ймовірності того що в цій частині зображення є даний об'єкт. Цю матрицю називають тепловою картою. Всі локальні максимуми які перевищують певне значення й будуть центрами цих об'єктів. В додаткових матрицях буде ширина й висота об'єктів.

Отже на вході зображенням  $I \in \mathbb{R}^{W \times H \times 3}$  ширини  $W$  і висоти  $H$ . На виході для кожного класу розпізнаваних об'єктів тепла карта  $Y \in [0,1]^{(W/R \times H/R \times C)}$ , ширина та висота виходу нейронної мережі менше вхідного зображення у 2 або 4 рази. Хочемо, щоб у точці, що є центром об'єкта  $Y(x,y,c) = 1$ , тоді як  $Y(x,y,c) = 0$  означає, що тут нічого немає. Тому під час тренування мережі при генеруванні ground truth даних значення на теплових картах навколо центрів об'єктів генеруємо за допомогою Гаусівського ядра середнє квадратичне якого залежить від розміру об'єкта

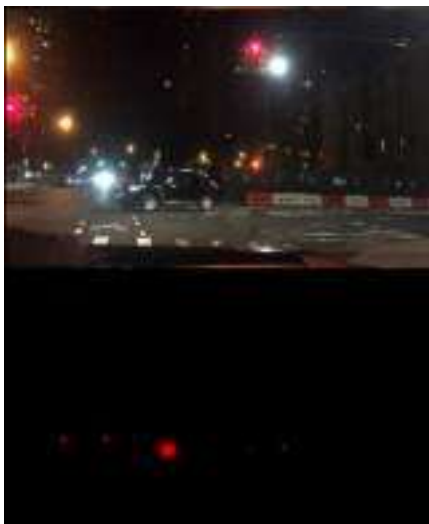


Рисунок 1. Генерація ground truth теплової карти для зображення

На цьому рисунку можна побачити приклад генерації ground truth гаусівськими ядрами, які розміщені в центрі машин та людей.

Окрім центрів об'єктів нам потрібно знайти їх ширину та висоту. Для цього до виходу моделі додаємо ще 4 шари, для визначення відстаней до правого, лівого, верхнього і нижнього країв прямокутника, що позначає об'єкт на зображенні. Тому після того як ми визначили центри об'єктів, визначаємо довжину і ширину зі значень по координатах центрів об'єктів з цих чотирьох шарів. Тому вихід CenterNet буде мати кількість каналів, що дорівнює кількості класів плюс чотири.

Функцією втрат під час тренувань буде сума функцій втрат з теплових карт та з шарів для визначення розмірів об'єкта, при цьому функція втрат для розмірів об'єкта працює тільки з точками що є центрами об'єктів, усі інші точки вона ігнорує.

Одним з головних плюсів CenterNet є те, що так само як визначається розміри об'єкту можна визначити будь які характеристики об'єкта просто додаючи додаткові шари до виходу, наприклад таким чином можна дізнатися орієнтацію машини і відстані до неї, що знадобляться до нашої задачі.

Щоб знайти відстань до об'єктів можна спробувати декілька варіантів. Наприклад знаючи середню висоту машини, можна знаходити відстань за висотою машин, а бо якщо ми знаємо розміщення камери можна пробувати знаходити відстань за розміщенням об'єкта у кадрі. Проте такі методи можуть видавати значні похибки, тому якщо є датасет в якому розмічені відстані до об'єктів можна просто додати їх до виходу CenterNet ще один канал з відстанями до об'єктів, що буде працювати аналогічно визначенню розміру об'єкта.

Задача відслідковування об'єктів полягає в тому щоб у відеопотоці зрозуміти що об'єкт, який присутній на декількох кадрах є одним і тим самим об'єктом, відслідкувати переміщення об'єкта у відео [4-6].

Найпростіший спосіб відслідкувати об'єкти це використовувати жадібний алгоритм, тобто найближчий об'єкт цього класу на попередньому кадрі і буде цим об'єктом. Або використовувати фільтри Калмана. Якщо частота кадрів велика то в принципі проблем не буде, проте існують і більш точні методи. Один з них це модифікація CenterNet CenterTrack. Головною його особливістю є те що замість одного зображення на вході ми беремо два: нинішній і попередній кадр а також теплову карту на якій розміщуємо центри об'єктів по аналогії з тим як це робилось у CenterNet. Тобто тепер вхід моделі буде не трьох каналним, а семи-каналним. На виході CenterTrack додатково додаємо два шари для визначення вектора зміщення відносно попереднього кадру.



Рисунок 2. Вхід і вихід CenterTrack

Потрібно зазначити що під час тренування на вхідній тепловій карті додатково будемо генерувати шуми, симулюючи помилки, бо якщо цього не робити, моделі буде складніше помічати нові об'єкти. Для співставлення об'єктів після врахування вектору зміщення використовуємо жадібний алгоритм.

Отже після того як ми розпізнали об'єкти знаємо на якій глибині в зображенні вони розташовані і можемо створити мапу об'єктів навколо машини. Далі можна застосувати рекурентну нейронну мережу що приймає координати об'єктів на кожному кадрі і намагається видати координати об'єкту на майбутньому кадрі.

### 3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Розділили задачу на декілька етапів. Спочатку розпізнавання об'єктів, їх глибини та відслідковування їх переміщень по кадрах відеопотоку. Потім розміщення розпізнаних об'єктів на карті навколо нашої машини, у якій знаходиться камери, які і знімають відео. Розпізнавання об'єктів, відстаней до них можна робити паралельно. Це заслуга CenterNet, що дозволяє легко прикріплювати до голови будь які характеристики об'єктів, при цьому майже не збільшуючи нейронну мережу. Завдяки тому що на вході використовується не одне зображення, а також попереднє та результати роботи мережі на попередньому кадрі це також трішки дозволяє поліпшити розпізнавання об'єктів проте не набагато. Тому будемо розв'язувати однією і тією ж самою нейронною мережею декілька задач, що зекономить час роботи. До одного і того тіла нейронної мережі будемо додавати декілька голів. Функцією втрат в такому випадку буде сума функцій втрат з кожної голови помножених на коефіцієнти. Щодо розпізнавання об'єктів в основному будемо експериментувати з

центернетівською головою оскільки вона легше в реалізації, її легше змінювати ніж Yolo, чи SSD. При цьому будемо використовувати не оригінальний CenterNet, а його модифікацію TTF. Де замість того щоб видавати ширину, висоту і зміщення центра об'єкта, буде відстань лівого, правого, нижнього і верхнього країв об'єкту. На точність це не вплине, а якщо і вплине то в кращу сторону проте в межах похибки, але при цьому це інтуїтивно більш зрозуміла голова. Для відслідковування об'єктів або як це ще називають трекінгом об'єктів можна використовувати CenterTrack, що дозволить нам об'єднати задачі розпізнавання об'єктів, знаходження відстаней до них та трекінг в одну мережу з трьома головами. Розпізнавати будемо 3 класи об'єктів автомобілі, мотоцикли/велосипеди, люди. Отже голова з розпізнавання об'єктів буде складатися з  $3+4=7$  каналів, відстані 1 канал та трекінг 2 канали, тобто всього 10 каналів.



Рисунок 3. Візуалізація роботи розпізнавання об'єктів разом з відстеженням

Отримана точність з розпізнавання об'єктів. Евалюація проводилася на датасеті BDD100K [7]: AP-integral - 0.190; AP-0.5: 0.407.

#### 4. ВИСНОВКИ

Задачу з прогнозування траєкторії учасників дорожнього руху було розкладено на декілька підзадач: розпізнавання об'єктів, знаходження відстаней до них, відслідковування об'єктів у відеопотоці, нанесення об'єктів на карту навколо машини, прогнозування траєкторій об'єктів. Перші етапи задачі вдалося поєднати в один, тобто вирішувати однією нейронною мережею, що значно прискорює швидкість роботи. Також було враховано те що робота йде з відеопотоком, що дозволяє моделям використовувати більше інформації та спиратися на результати роботи на попередніх кадрах відеопотоку. Було натреновано нейронну мережу для вирішення кожної з підзадач та продемонстровано результати її роботи

#### ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Objects as Points. URL: <https://arxiv.org/abs/1904.07850>
2. SSD: Single Shot MultiBox Detector. URL: <https://arxiv.org/abs/1512.02325>
3. Tracking Objects as Points. URL: <https://arxiv.org/pdf/2004.01177.pdf>
4. Orthographic Feature Transform for Monocular 3D Object Detection. URL: <https://arxiv.org/pdf/1811.08188.pdf>
5. Vehicle Trajectory Prediction Using Hierarchical Graph Neural Network for Considering Interaction among Multimodal Maneuvers. URL: <https://www.mdpi.com/1424-8220/21/16/5354>
6. Vehicle Trajectory Prediction with Lane Stream Attention-Based LSTMs and Road Geometry Linearization/ URL: <https://www.mdpi.com/1424-8220/21/23/8152>
7. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. URL: <https://arxiv.org/pdf/1805.04687.pdf>

# МОДЕЛЬ ДЛЯ РОЗПІЗНАВАННЯ ВІЙСЬКОВОЇ ТЕХНІКИ ТА ВІЙСЬКОВОСЛУЖБОВЦІВ ІЗ КАМЕРИ БПЛА В РЕАЛЬНОМУ ЧАСІ ДЛЯ СИСТЕМ З ОБМЕЖЕНИМИ ОБЧИСЛЮВАЛЬНИМИ РЕСУРСАМИ

Іванов С.І.<sup>1</sup>, Шубенкова І.А.<sup>2</sup>

Національний Технічний Університет України  
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

<sup>1</sup> sergiy.ivanov0210@gmail.com,

<sup>2</sup> shubenkova1959@gmail.com [0000-0002-7433-2070]

**Безпілотні літальні апарати широко використовуються в сучасній військовій сфері, одним із суттєвих факторів підвищення ефективності їх застосування є використання системи розпізнавання об'єктів із камери дрону для допомоги оператору. Метою роботи є збір та попередня обробка вхідних даних та розробка моделі для розпізнавання військової техніки та військовослужбовців із зображення з камери БПЛА в реальному часі. Результатом дослідження є навчена модель у вигляді згорткової нейронної мережі, адаптована під практичне використання із системами з обмеженими обчислювальними ресурсами. В роботі було використано емпіричні й теоретичні методи дослідження.**

**Ключові слова: комп'ютерний зір, машинне навчання, згорткові нейронні мережі, розпізнавання об'єктів, глибоке навчання, БПЛА**

Системи розпізнавання образів та об'єктів із зображень широко використовуються у багатьох сферах сучасного життя, й одним із перспективних напрямів їх застосування є системи розпізнавання об'єктів з камер дронів, що набули стрімкої популярності як в цивільній, так і у військовій сфері [1, 2].

У порівнянні із цивільними застосуваннями, розробка й практичне використання систем розпізнавання образів у військовій сфері пов'язані із додатковими складнощами. Серед них обмежений доступ до даних, складні умови використання таких систем, що вимагають точності, надійності й відповідності нормам безпеки [3]. Із цього випливає необхідність у розробці моделей машинного навчання для розпізнавання об'єктів, що можуть бути застосовані на інтегрованих системах та робочих станціях операторів за умов низької обчислювальної потужності та адаптовані до реальних умов.

## 1. РОЗПІЗНАВАННЯ ОБ'ЄКТІВ У ВІЙСЬКОВІЙ СФЕРІ

Системи розпізнавання образів використовуються у військовій сфері досить широко, однак й досі зацікавленість в їх використанні не падає. Серед застосувань слід виділити розпізнавання стратегічних об'єктів на супутникових знімках, розпізнавання техніки та особового складу на фото та відео із соціальних мереж та інших відкритих джерел [4], застосування із системами камер наземної техніки та безпілотних літальних апаратів для маневрування, наведення, автономного пересування, виявлення цілей та допомоги оператору.

У військовій сфері БПЛА можуть виконувати різноманітні задачі, серед яких розвідка, патрулювання, знищення цілей та перевезення легкого вантажу. Оснащення БПЛА системами комп'ютерного зору може значно спростити роботу таких дронів, наприклад, пошвидшивши виявлення та класифікацію цілей, покращивши наведення на ціль, спростивши виявлення потенційних загроз.

Значним стримуючим фактором розвитку досліджень у сфері військового застосування систем розпізнавання об'єктів з БПЛА є наявність даних та інформації. Дуже часто моделі доводиться навчати на публічно доступних еталонних наборах даних, таких як ImageNet, COCO, публічних наборах даних для дронів для цивільних задач як VisDrone, MOR-UAV[1] та інші, після чого проводити донавчання на обмеженій кількості власних даних із використанням transfer learning [5]. Доступні у відкритому доступі набори даних часто є придатними лише до окремих способів використання та вкрай обмеженими, недієвими для навчання моделей у реальних умовах [6]. Ускладнюючими розпізнавання із камер БПЛА у військовій сфері факторами є низька якість вихідного зображення, часті спотворення форм, кутів огляду, розмиття, той факт, що під час виконання бойових задач дрони значну частину часу проводять у русі, та мають водночас слідкувати за рухомими цілями. Окрім цього, об'єкти часто можуть не повністю потрапляти в кадр, знаходитись на великій відстані що, враховуючи низьку роздільну здатність значної частини монтованих камер, значно ускладнює як виявлення так і класифікацію об'єктів на зображенні.

Додаткові складності для використання систем машинного навчання для задач розпізнавання створює те, що більшість робочих станцій операторів дронів мають досить обмежені обчислювальні ресурси. Переважна більшість використовуваних у військових цілях дронів в останні роки – це легкі дрони, які мають лише компактну камеру та іноді також певний вантаж або бойовий заряд [7]. Це виключає можливість проведення обчислювальних операцій використовуючи обладнання самого дрону. Так само неможливе й застосування хмарних обчислювальних систем [8] через необхідність постійного зв'язку із віддаленим обчислювальним пристроєм та проблеми щодо захисту даних та їх потенційної стратегічної значимості. Натомість, обчислення оптимально проводити безпосередньо на робочому пристрої оператора дрону, які мають відповідати вимогам мобільності, особливо в умовах використання безпосередньо в районах ведення бойових дій, та часто не мають достатніх обчислювальних ресурсів для використання на них більшості сучасних моделей глибокого навчання.

Таким чином існує необхідність в масштабованому рішенні, тобто створенні моделі, що була б здатна відповідати вимогам за точністю розпізнавання об'єктів, швидкості роботи для використання в режимі реального часу та могла би використовуватись на різних робочих станціях від командних пунктів до мобільних пристроїв для максимальної мобільності оператора.

## **2. РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ**

Для дослідження було використано набір даних, зібраний студентами та викладачами Факультету прикладних наук Українського Католицького Університету [9]. Набір даних включає в себе більше 100 відеозаписів, зібраних із відкритих джерел за перші пів року повномасштабної війни Росії проти України. Відеозаписи являють собою реальні кадри із дронів різних класів та напрямів застосування, зібрані безпосередньо під час виконання бойових завдань. Об'єкти інтересу в кадрах із набору даних поділено на 3 класи: військові транспортні засоби, військовослужбовці та невизначені рухомі об'єкти.

Однією з головних переваг набору даних є те, що відеозаписи добре відображають типові фактори, що ускладнює розпізнавання об'єктів із камери дронів за реальних бойових

умов. Кадри часто мають низьку роздільну здатність, розмиті, зашумлені, мають певні викривлення форм об'єктів, включають кадри із складними природними умовами, як туман та сніг, ускладнюючі техногенні фактори як задимлення, горіння. Об'єкти інтересу також використовують природне й рукотворне маскування, що ускладнює розпізнавання та складне оточення, рух як об'єктів так і дрону, перекриття об'єктів іншими об'єктами інтересу або оточенням (Рис.1).



Рисунок 1. Кадри із набору даних, що демонструють деякі типові викривлення та ускладнюючі розпізнавання об'єктів фактори

Для розпізнавання об'єктів на основі вибраного набору навчальних даних було обрано модель згорткової нейронної мережі EfficientDet компактної модифікації Lite0.

Мережі EfficientDet відрізняються ефективним поєднанням високої точності розпізнавання, протестованої на таких еталонних наборах даних як COCO [10], та подекуди в рази, вищою швидкістю та компактністю моделей за аналогічні детектори [11]. Ще одна важлива особливість моделей це їх повна масштабованість, лінійка моделей включає як надточні та важкі, так і легкі, швидкі моделі із достатньою для практичного застосування точністю.

Архітектура мережі складається з 3 частин: «хребта» - попередньо навченої на наборі даних ImageNet мережі EfficientNet, що відповідає за виділення основних ознак на вихідному зображенні, двосторонньої пірамідальної мережі ознак (Bi-FPN), що дозволяє мережі краще обробляти й знаходити ознаки у об'єктах різних розмірів шляхом масштабування ознак в обох напрямках, та мережі прогнозування класів і меж об'єкта за допомогою softmax-класифікаторів та регресорів (Рис. 2) [11].

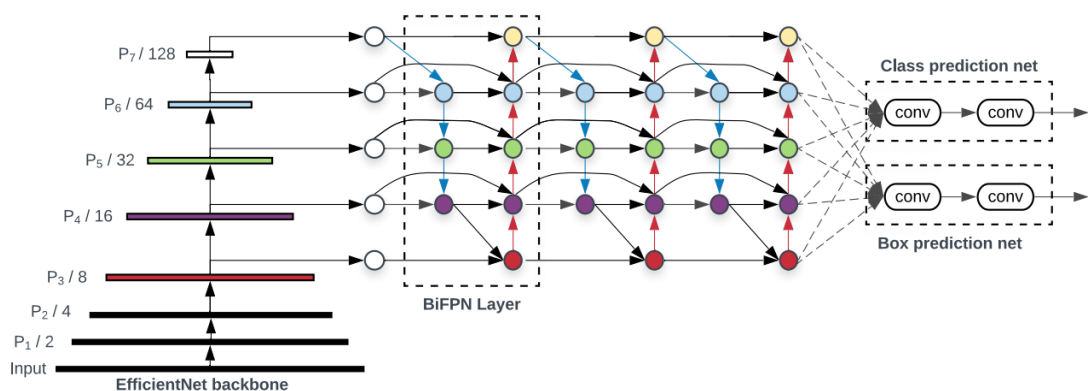


Рисунок 2. Структурні елементи архітектури мереж iEfficientDet

Модифікація мережі Lite0 відповідає найлегшому типу мережі, із найнижчим фактором масштабування, що має всього 3.9 мільйона параметрів й важить всього 4 мегабайти. Відповідно до фактору масштабування, «хребтом» мережі виступає EfficientNet-Lite0, мережа має 3 блоки BiFPN із 64 фільтрами та 3 Box/Classблоки відповідно. Головною перевагою цієї мережі є її швидкість та легкість, що дозволяє використовувати її на пристроях із низькими обчислювальними можливостями. На Google Pixel мережа показала середню швидкість обробки кадру в 36 мс, що відповідає 28 кадрам за секунду, тобто дозволяє використовувати модель в реальному часі навіть на смартфоні.

За результатами навчання побудованої моделі на наборі реальних даних із БПЛА із відкритих джерел було отримано хороші показники середньої точності як поміж класами, так і для ключового класу об'єктів Військовий транспортний засіб, що є домінантним у вихідному наборі даних (Табл. 1).

Таблиця 1. Показники точності мережі на тестовому наборі даних із 1168 зображень

Показник	mAP	AP50	AP транспортний засіб	AP військовослужбовець
Значення	22.4	43.6	41.8	10.6

Модель також показала дуже високу швидкість, витрачаючи на обробку кадру всього 23.125 мілісекунд, що відповідає роботі в реальному часі на 43 кадрах за секунду. Експерименти проводились на графічному процесорі nVidiaTesla із максимальним обсягом пам'яті графічного процесору у 32Гб, однак в процесі навчання для досягнення наведеної швидкості було достатньо лише незначної частини цього обсягу (Табл. 2).

Таблиця 2. Показники швидкості мережі на тестовому наборі даних із 1168 зображень

Показник	Затримка, мс	Кадри на секунду (FPS)	Використаний об'єм пам'яті графічного процесору, Гб
Значення	23.1	43.24	3.56

Кінцевим результатом роботи мережі є спрогнозовані околиці об'єктів інтересу на вихідному зображенні та класифікація цих об'єктів (Рис. 3).

Як ми можемо бачити, модель добре розпізнає та класифікує об'єкти навіть за вкрай складних реальних умов, включаючи перекриття об'єктів, погодні умови, викривлення й низьку якість зображення, маскування, великі відстані до об'єктів.

Розроблена модель може використовуватись на мобільних пристроях на системах Android та iOS, можливе використання на компактних мікрокомп'ютерах RaspberryPi, де достатня швидкість для розпізнавання в реальному часі досягається з версії 4 для 4 Гб вбудованої пам'яті. Модель також портується для використання на компактних процесорах EdgeTPU та може легко працювати на високій швидкості на більшості сучасних ноутбуків середньої потужності та багатьох планшетах. Можливість застосування на компактних процесорах та в інтегрованих системах також дозволяє застосовувати модель для проведення обчислень безпосередньо на обчислювальних ресурсах деяких дронів. Це значно спрощує її використання в реальних умовах на максимально мобільних робочих станціях.



Рисунок 3. Результати роботи мережі на кадрах із камер дронів

Розроблена модель може використовуватись для допомоги операторам дронів будь-якого класу, оснащених камерою, для виконання різноманітних бойових та небойових задач. Низькі вимоги до обчислювальних ресурсів, висока швидкість роботи та достатня точність розпізнавання роблять модель надійним інструментом допомоги оператору БПЛА, що готова до використання в реальних умовах.

### 3. ВИСНОВКИ

Розпізнавання об'єктів із камер дронів у військовій сфері має значні складності, пов'язані із доступністю даних, обчислювальними потужностями пристроїв, вимогами до безпеки, точності та швидкості роботи системи в реальному часі. Існує потреба у

використанні легких моделей, адаптованих під системи із обмеженими обчислювальними ресурсами, на мобільних станціях операторів.

В результаті дослідження було реалізовано модель у вигляді згорткової нейронної мережі для розпізнавання військової техніки, військовослужбовців та інших рухомих об'єктів на кадрах з камери дрону. Модель була навчена на реальних даних, зібраних із відкритих джерел, що враховують основні спотворення зображення та складності для розпізнавання що відповідають використанню системи в реальних умовах. Розроблена модель показала хороші точність та швидкість, здатність працювати в реальному часі, а також є легкою і може бути застосована на інтегрованих системах, мобільних пристроях, планшетах, ноутбуках та інших робочих станціях операторів БПЛА, котрі мають обмежені обчислювальні ресурси. Модель може використовуватись для допомоги операторам дронів у проведенні розвідки, патрулюванні території, а також для бойових задач, як система визначення та класифікації цілей.

Враховуючи масштабованість моделі, як напрям подальшого дослідження можлива розробка моделей із різними показниками точності та вимогами до обчислювальних ресурсів. Можливе розширення набору даних для покращення результатів та додаткової диверсифікації даних. Також можливе поєднання системи розпізнавання об'єктів із системами керування дронами на основі комп'ютерного зору для створення автономних систем, наприклад для задач патрулювання.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Wu X., Li W., Hong D., Tao R., Du Q. Deep learning for unmanned aerial vehicle-based object detection and tracking: a survey. *IEEE Geoscience and Remote Sensing Magazine*. 2018. Vol. 10, No. 1. P.91–124.

2. Nex F. et. al. UAV in the advent of the twenties: Where we stand and what is next. *ISPRS Journal of Photogrammetry and Remote Sensing*. 2022. Vol. 184. P. 215–242.

3. Svenmarck P., Luotsinen L., Nilsson M., Schubert J. Possibilities and challenges for artificial intelligence in military applications. In proceedings of the NATO Big Data and Artificial Intelligence for Military Decision Making Specialists' Meeting. Bordeaux, France, 30 May –1 June, 2018. P. 1–16.

4. Hiipala T. Recognizing military vehicles in social media images using deep learning. *ISI 2017: Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatic*, Beijing, China, July 22–24, 2017.

5. Yang Z., Yu W., Liang P., Guo H., Xia L., Zhang F., Ma J. Deep transfer learning for military object recognition under small training set condition. *Neural Computing and Applications*. 2019. Vol. 31, No. 10. P.6469–6478.

6. Schumacher R., Rosenbach Kh. ATR of Battlefield Targets by SAR – Classification Results Using the Public MSTAR Dataset Compared with a Dataset by QinetiQ, UK. *RTO SET: In proceedings of Symposium on “Target Identification and Recognition Using RF Systems”*. Oslo, Norway, October 11–13, 2004. P.31–42.

7. Pong B. The Art of Drone Warfare. *Journal of War & Culture Studies*. 2022. Vol. 15. P. 377–387.

8. Jangwon L. et. al. Real-time object detection for unmanned aerial vehicles based on cloud-based convolutional neural networks. *Journal Concurrency and Computation: Practice and Experience*. 2017. Vol.29.

9. Факультет прикладних наук УКУ. На передовій безпілотники відіграють надзвичайно важливу роль, тому дуже важливо вміти швидко і безперешкодно аналізувати отримані дані.

URL: <https://www.facebook.com/100063512295462/posts/pfbid02uEWNEKcAVT4 набаSgR1DJCfXnrKX1jcCMzQHnoivoNUPbD5iGv93ZTTYREFeRbBl/> (дата звернення: 02.11.2022).

10. LinTsun-Yiet. al. Microsoft COCO: Common objects in context. *ECCV 2014: Proceeding of the 13th European Conference on Computer Vision*. Zurich, Switzerland, September 6-12, 2014. P. 740–755.

11. Mingxing T., Ruoming P., Quoc V. Le. EfficientDet: Scalable and Efficient Object Detection. *IEEE/CVF: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, Washington, USA, June 13–19, 2020. P. 10781–10790.

# РЕКОМЕНДАЦІЙНА СИСТЕМА НА ОСНОВІ ОГЛЯДІВ КОРИСТУВАЧІВ І ПРОДУКТІВ

Кравченко О.В.<sup>1</sup>, Недашківська Н.І.<sup>2</sup>

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

<sup>1</sup>kravchenko.olexandra@lil.kpi.ua,

<sup>2</sup>nedashkovskaya.nadezhda@lil.kpi.ua [0000-0002-8277-3095]

Рекомендаційні системи займають важливе місце в процесі життєдіяльності багатьох великих компаній. Рекомендаційні системи підвищують лояльність користувачів до різних інтернет-сервісів. Рекомендаційні системи допомагають користувачам знайти фільми, музику, інше за їх вподобаннями, рекомендують нові проекти. Одним з актуальних напрямків розвитку рекомендаційних систем є використання оглядів для рекомендацій. Метою роботи є аналіз і порівняння існуючих алгоритмів надання рекомендацій на основі оглядів, а також пропонування модифікації методу для покращення рекомендацій.

**Ключові слова:** рекомендаційна система, огляди для рекомендацій, DeepCoNN, eCommerce, нейромережа

## 1. ВСТУП

Рекомендаційні системи займають важливе місце в процесі життєдіяльності майже будь-яких інтернет-сервісів. Вони допомагають збільшити дохід бізнесу, покупцям – обрати найбільш підходящий для них продукт, а також зменшити навантаження на живих продавцях-консультантах. Ці системи пройшли шлях від набору заздалегідь сформованих рекомендацій для всіх покупців одразу до персоналізованих, що враховують не тільки загальні інтереси користувача, але і настрої і контекст сеансу.

Останніми роками деякі дослідники припустили, що текст огляду може допомогти більше, ніж характеристики товару і вподобання користувача. Таким чином з'явився відносно новий напрямок дослідження систем рекомендацій – засновані на оглядах.

## 2. МЕТОДИ НАДАННЯ РЕКОМЕНДАЦІЙ НА ОСНОВІ ОГЛЯДІВ

Рекомендаційні системи все ще знаходяться на етапі стрімкого розвитку, в наукових журналах постійно з'являються статті, які пропонують нові алгоритми надання рекомендацій і способи покращення вже існуючих.

На сьогоднішній день існує кілька популярних напрямків досліджень рекомендаційних систем для покращення точності наданих рекомендацій. Серед них можна виділити наступні:

1. Використання оглядів для рекомендацій. Загалом у цьому напрямку стверджується, що оскільки огляди «пояснюють» думки користувачів, вони повинні бути корисними для визначення базових параметрів, які передбачають оцінки чи покупки. Деякі дослідники [1] навпаки стверджують, що такі моделі показують себе мало не гіршими за еталонні моделі, якщо помістити їх в повністю однакові умови.

2. Передбачення рейтингу переходів. За допомогою рейтингу переходів (click-through rate, CTR) ми можемо в певній формі виміряти скільки кліків отримали рекомендації. Основне припущення полягає в тому, що більша кількість кліків на рекомендованих елементах означає, що рекомендації були більш релевантними для користувачів.

3. Контекстно-залежна система рекомендацій (CARS) застосовує визначення й аналіз контексту користувача для надання персоналізованих послуг. Контекстну інформацію можна отримати від датчиків, щоб підвищити точність рекомендацій.

4. Використання автокодерів для рекомендацій. В той час як традиційні моделі працюють лише з одним джерелом даних (оцінка чи текст), моделі на основі автоматичного кодувальника можуть опрацьовувати неоднорідні джерела даних (рейтинг, аудіо, відео).

5. Послідовні рекомендації. Системи послідовних рекомендацій намагаються зрозуміти поведінку користувача з часом і моделюють таким чином послідовності дій користувача.

В статті розглянемо лише перший напрямок дослідження: використання оглядів для рекомендацій. Серед них можна виділити такі моделі.

DeepCoNN [2] – це глибока модель спільного вивчення властивостей предмета та поведінки користувача на основі тексту огляду. Модель складається з двох паралельних нейронних мереж, з'єднаних на останніх рівнях. Одна з мереж зосереджена на вивченні поведінки користувачів, використовуючи відгуки, написані користувачем, а інша вивчає властивості предмета з відгуків, написаних для товару. Спільний рівень об'єднує ці дві мережі. Спільний рівень дозволяє латентним факторам, отриманим користувачами та елементами, взаємодіяти один з одним у спосіб, подібний до методів машинної факторизації.

NARRE [3] аналізує важливість кожного огляду перед використанням. Автори стверджують, що менш корисні відгуки шкодять продуктивності моделі, а також менш значущі для користувача. Для дослідження важливості оглядів модель використовує механізм уваги.

HFT [4] – особлива модель у цьому списку, використовує огляди як регуляризатор. По суті, така модель не просто використовує текст огляду для визначення прихованих вподобань, але й визначає, наскільки об'єктивною відносно тексту огляду є відповідна оцінка.

TransNets [5] – розширення моделі DeepCoNN, що вводить додатковий латентний рівень, що представляє пару користувач-елемент. Під час навчання цей рівень впорядковується, щоб бути схожим на інше приховане представлення огляду цільового користувача цільового елемента.

MPCN [6] – модель з багатоієрархічною парадигмою, що ґрунтується на припущенні, що не всі огляди однакові, тобто важливі лише кілька обраних. Важливість, однак, має динамічно виводитися залежно від поточної цілі. З цією метою пропонується схема навчання на основі вказівника рецензії за рецензією, яка виділяє важливі рецензії, а потім зіставляє їх слово за словом. Це дає змогу не лише використовувати найбільш інформативні огляди для прогнозування, але й глибше взаємодіяти на рівні слів.

NRPA [7] припускає, що важливість відгуків різна для різних користувачів і елементів, а також одне і те ж слово або схожі відгуки можуть мати різну інформативність для різних користувачів і товарів. Тому модель використовує персоналізовану увагу, щоб врахувати це.

DAML [8] – модель, що використовує локальну та взаємну увагу згорткової нейронної мережі, щоб спільно вивчати особливості оглядів для покращення інтерпретації. Потім рейтингові функції та функції перегляду інтегруються в уніфіковану модель нейронної мережі, а нелінійна взаємодія ознак вищого порядку реалізується машинами нейронної факторизації для завершення прогнозу остаточного рейтингу.

### 3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для дослідження було обрано чотири датасети з набору Amazon product data: Apps for Android, Video Games, Cell Phones and Accessories, Office Products (розташовані в порядку спадання розміру датасету). Для спрощення процесу навчання був обраний 5-core варіант даних. Для дослідження набори даних були попередньо оброблені на розділені по схемі 80-10-10 на сети для навчання, тестування і перевірки.

Для подальшого дослідження обрано дві моделі, описані вище – DeepCoNN і NARRE. Також розглянуто NeuMF – проста і розповсюджена модель нейронної колаборативної фільтрації, що використовує машину факторизації для заповнення пропущених зв'язків. Алгоритм NeuMF не використовує текст огляду і обраний для об'єктивності дослідження.

Також для дослідження запропоновано модифікацію моделі DeepCoNN – DeepAtnCoNN. Припущення полягає в тому, що за умови використання Textual Preprocessing (64-вимірні word2vec вбудовування, навчені за допомогою Gensim), а також додавання додаткового Attention шару результат може бути набагато кращим за початковий. Припущення базується на кількох альтернативних статтях в напрямку рекомендацій за допомогою оглядів [7, 8]. Також замість машини факторизації на останньому етапі спробуємо дослідити різного роду зміщення: глобальні, користувачів і елементів, і додати їх до виходу нейронної мережі.

Показниками точності обрано MSE, nDCG@5 і HR@1 [1]. Перші дві метрики використовувалися і під час навчання, і під час тестування моделей, а остання – тільки для тесту моделей після закінчення навчання для загальної оцінки їх ефективності.

Проведено два дослідження: значень втрат протягом 10 епох навчання, і дослідження точності залежно від значення дропауту (dropout). Значення MSE для відомих моделей DeepCoNN, NARRE і NeuMF та запропонованої DeepAtnCoNN на епохах 1, 2, ..., 10 наведені на Рис.1 і 2.

Можна побачити, що запропонована модель DeepAtnCoNN працює краще за інші розглянуті моделі за умови великого набору даних. Крім того, бачимо, що моделі починають перенавчатися в середньому після п'ятої епохи, тому у подальших дослідженнях будемо переривати навчання на цьому етапі. DeepCoNN має повільно спадаючу тенденцію по метриці MSE протягом всього навчання. Запропонована модель має кращу точність ніж розглянуті три інші на наборах даних Android Apps, Cells Phones і Video Games, а на останньому наборі Office Products дає кращі результати ніж DeepCoNN і NARRE, однак ці результати майже не відрізняються від NeuMF, яка не використовує огляди для рекомендацій.

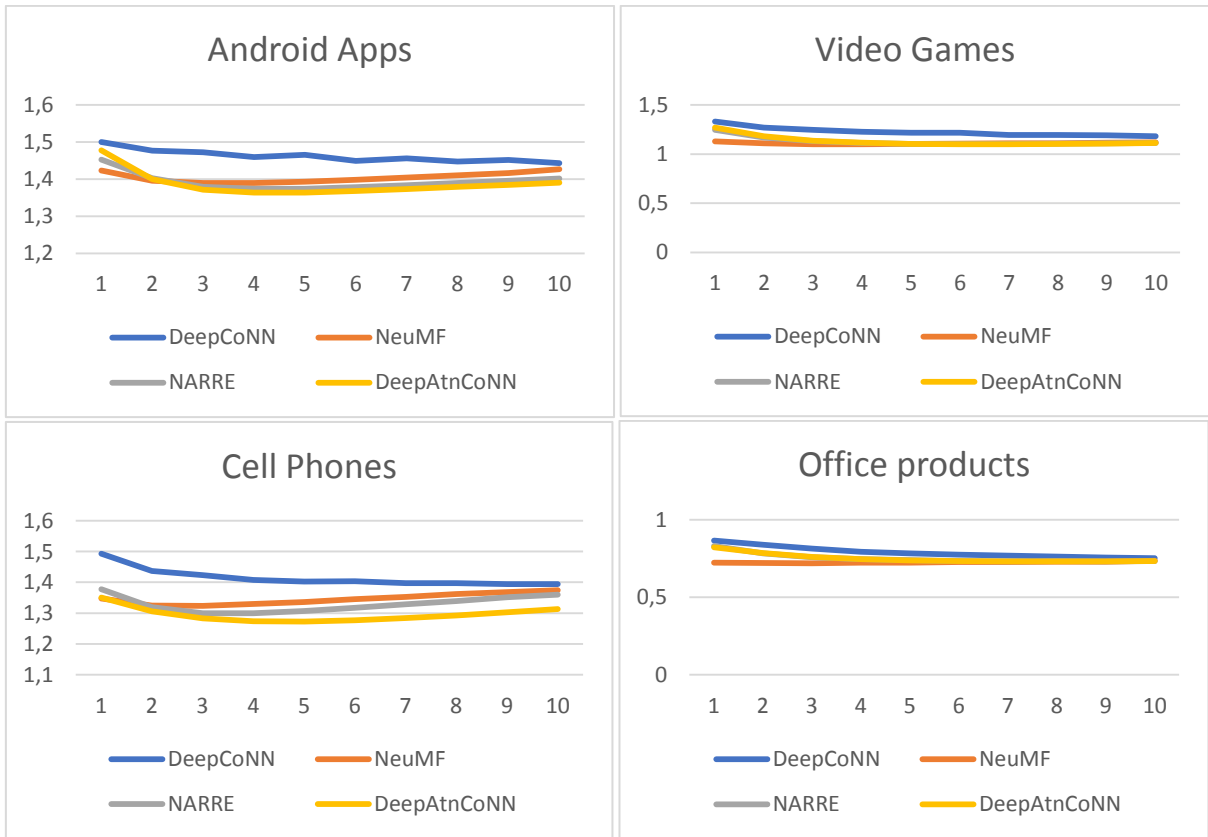


Рисунок 1. Залежність MSE від кількості епох

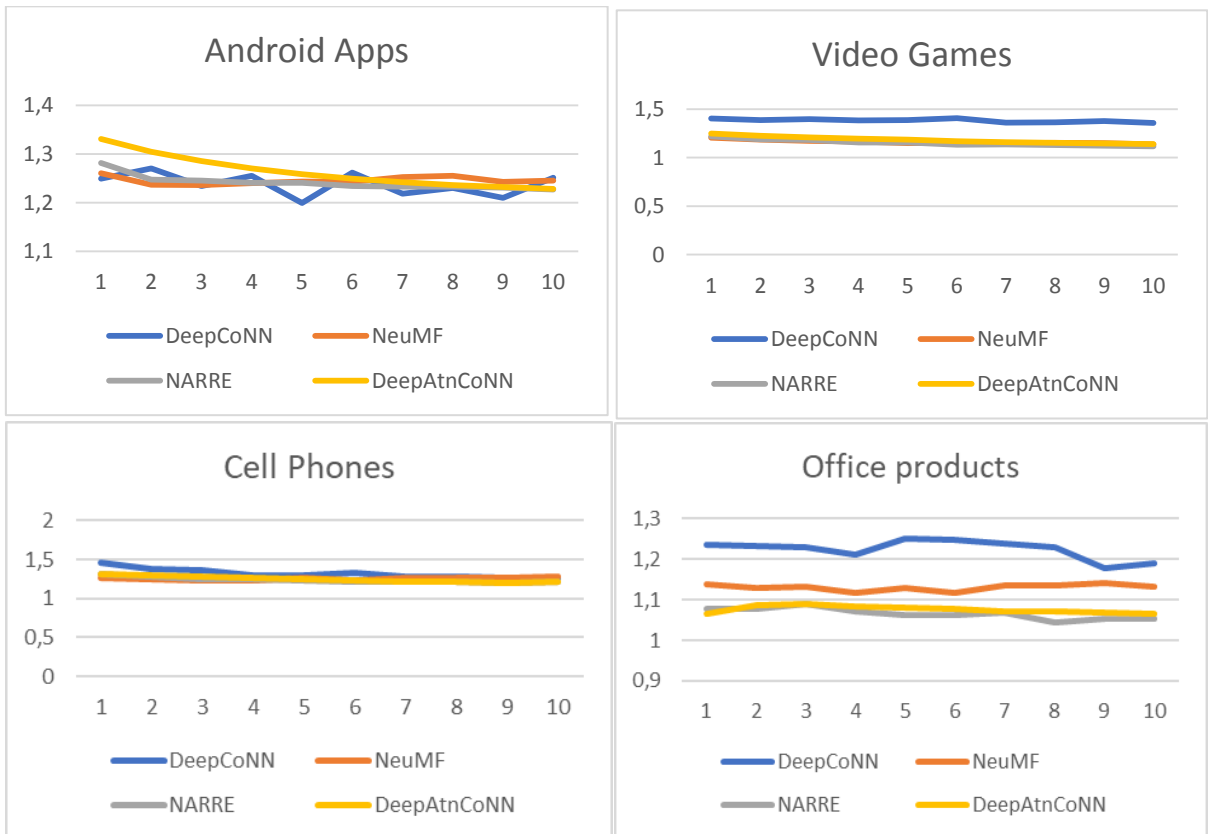


Рисунок 2. Залежність NDCG від кількості епох

Таким чином можна зробити додатковий проміжний висновок, що для маленьких розріджених наборів даних немає необхідності використовувати огляди.

В Таблиці 1 наведено фінальні результати тестування найкращих моделей на чотирьох наборах даних.

#### 4. ВИСНОВКИ

В сучасному світі постійно з'являються нові шляхи покращення механізму рекомендацій. Одним з них є рекомендації на основі оглядів. Припускається, що огляди можуть містити додаткову інформацію про приховані вподобання користувача та специфічні характеристики продукту.

У ході дослідження проаналізовано і порівняно дві класичні моделі рекомендацій на основі оглядів: DeepCoNN і NARRE, а також ту, що не використовує текст огляду – NeuMF. Також запропоновано модифікацію методу DeepCoNN – DeepAtnCoNN з механізмом уваги. Обрано підходящий набір даних Amazon Products Data і проведено його попередню обробку. Проведено дослідження кількох класичних алгоритмів обраного напрямку і порівняно їх точність за показниками MSE і nDCG@5. На базі цього запропоновано модифікацію обраного методу і проведено його дослідження і порівняння з початковими.

Таблиця 1. Значення показників точності MSE, nDCG@5 і HR@1 при фінальному тестуванні моделей DeepCoNN, NeuMF, NARRE та DeepAtnCoNN

	DeepCoNN				NeuMF			
	Apps	Game	Phone	Office	Apps	Game	Phone	Office
MSE	1.4475	1.1856	1.3621	0.7313	1.3789	1.1019	1.3364	0.7129
nDCG@5	1.2393	1.3913	1.2697	1.2025	1.2218	1.1584	1.2171	1.1294
HR@1	31.340	39.090	36.360	44.440	33.330	44.170	70.00	27.270

Продовження таблиці 1

	NARRE				DeepAtnCoNN			
	Apps	Game	Phone	Office	Apps	Game	Phone	Office
MSE	1.3599	1.1013	1.3134	0.7398	1.3708	1.1026	1.3166	0.7042
nDCG@5	1.2323	1.1847	1.1987	1.1146	1.2574	1.2053	1.2474	1.1752
HR@1	37.620	31.970	27.270	50.00	37.310	40.910	45.450	33.330

Пропонована модифікація виявилася точнішою на 5-7% в основному для всіх обраних наборів даних порівняно з іншими розглянутими алгоритмами.

#### ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Noveen Sachdeva and Julian McAuley. How Useful are Reviews for Recommendation? A Critical Review and Potential Improvements. SIGIR '20: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval July 2020 Pages 1845–1848 <https://doi.org/10.1145/3397271.3401281>

2. Lei Zheng, Vahid Noroozi, Philip S. Yu Joint Deep Modeling of Users and Items Using Reviews for Recommendation WSDM '17: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining February 2017 Pages 425–434 <https://doi.org/10.1145/3018661.3018665>

3. Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2018. Neural Attentional Rating Regression with Review-level Explanations. In *The Web Conference 2018*, April 23–27, 2018, Lyons, France. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3178876.3186070>
4. Julian McAuley, Jure Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. *RecSys '13: Proceedings of the 7th ACM conference on Recommender systems* October 2013 Pages 165–172 <https://doi.org/10.1145/2507157.2507163>
5. Rose Catherine, William Cohen. TransNets: Learning to Transform for Recommendation. *RecSys '17: Proceedings of the Eleventh ACM Conference on Recommender Systems* August 2017 Pages 288–296 <https://doi.org/10.1145/3109859.3109878>
6. Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Multi-Pointer Co-Attention Networks for Recommendation. In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, August 19–23, 2018, London, United Kingdom. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3219819.3220086>
7. Hongtao Liu, Fangzhao Wu, Wenjun Wang, Xianchen Wang, Pengfei Jiao, Chuhan Wu, and Xing Xie. 2019. NRPA: Neural Recommendation with Personalized Attention. In *42nd Int'l ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'19)*, July 21–25, 2019, Paris, France. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3331184.3331371>
8. Donghua Liu, Jing Li, Bo Du, Jun Chang, Rong Gao. DAML: Dual Attention Mutual Learning between Ratings and Reviews for Item Recommendation. *KDD '19: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* July 2019 Pages 344–352 <https://doi.org/10.1145/3292500.3330906>

# ДОСЛІДЖЕННЯ ГІБРИДНИХ МЕРЕЖ ГЛИБОКОГО НАВЧАННЯ В ЗАДАЧАХ ПРОГНОЗУВАННЯ У ФІНАНСОВІЙ СФЕРІ

Кузьменко О.В.<sup>1</sup>, Зайченко Ю.П.<sup>2</sup>

КПІ ім. Ігоря Сікорського, Київ, Україна

<sup>1</sup>oleksii.kuzmenko@ukr.net [0000-0003-1581-6224],

<sup>2</sup>zaychenko Yuri@ukr.net [0000-0001-9662-3269]

**У роботі розглядаються проблеми короткострокового та середньострокового прогнозування у фінансовій сфері. Для їх розв'язання запропоновано інтелектуальні методи прогнозування: LSTM та гібридні мережі глибокого навчання на основі МГУА. Знайдено оптимальні параметри LSTM та гібридних мереж. Побудовано оптимальні структури гібридних мереж. Проведено експериментальні дослідження LSTM та гібридних мереж, а також порівняно їх точність.**

**Ключові слова: короткострокове прогнозування, середньострокове прогнозування, LSTM, гібридна мережа глибокого навчання, оптимізація.**

## 1. ВСТУП

Проблемам прогнозування курсів акцій і ринкових індексів на фондових біржах приділяється велика увага інвесторів і різних грошових фондів. Для їх вирішення були розроблені та досліджені потужні інтелектуальні методи та технології, серед яких нейронні мережі та системи нечіткої логіки.

Ефективним інструментом моделювання для короткострокового та середньострокового прогнозування нестационарних часових рядів є мережі LSTM. Вони давно розроблені та успішно застосовуються для прогнозування на біржах [1-5].

В якості альтернативного підходу для прогнозування у фінансовій сфері застосовують різні типи нейронних мереж: MLP [6], нечітких нейронних мереж [7,8], нео-нечітких мереж [9] та мереж глибокого навчання (DL) [10]. Новим трендом в області мереж DL є новий клас мереж – гібридні мережі DL на основі методу МГУА [11]. Застосування самоорганізації в цих мережах дозволяє навчати не тільки ваги нейронів, але й будувати оптимальну структуру мережі. Завдяки методу навчання в цих мережах ваги коригуються не одночасно, а шар за шаром. Це запобігає феномену вибуху градієнта, що дуже важливо для мереж з багатьма шарами.

У перших роботах в цій області у якості вузлів гібридної мережі використовувалися нейрони Ванга-Менделя з двома входами [11]. Але недоліком таких нейронів є вимога навчати не лише нейронні ваги, але й параметри нечітких функцій приналежності (ФП) в антецедентах правил. Це потребує значних розрахункових витрат і тривалого часу на навчання. Тому пізніше були розроблені нео-фазі-мережі DL, в яких у якості вузлів використовуються нео-фазі-нейрони Ямакави [12,13]. Основна властивість таких нейронів полягає в тому, що необхідно навчати лише ваги нейронів, а не нечіткі ФП. Це вимагає менше обчислень порівняно з нейронами Ванга-Менделя і значно скорочує час навчання в цілому. У роботі [13] було проведено дослідження обох класів гібридних мереж DL та порівняно їх ефективність при прогнозуванні у фінансовій сфері.

Тому виникає проблема порівняння ефективності гібридних мереж DL та LSTM в задачі прогнозування у фінансовій сфері [14].

Метою даної роботи є дослідження точності гібридних мереж DL і LSTM у задачі прогнозування ринкових індексів на біржі, порівняння їх ефективності на різних інтервалах та визначення класів задач прогнозування, для яких застосування відповідних обчислювальних інтелектуальних технологій є найбільш перспективними.

## 2. ОПИС ГІБРИДНОЇ МЕРЕЖІ МГУА-НЕО-ФАЗИ

Архітектура гібридної еволюційної DL-мережі представлена на рис. 1. На вхідний рівень системи подається  $(n \times 1)$ -вимірний вектор вхідних сигналів. Після цього сигнал передається на перший прихований шар. Цей шар містить нейрони, кожен з яких має лише два входи.

На виходах першого прихованого шару формуються вихідні сигнали. Потім ці сигнали подаються на блок селекції першого прихованого шару.

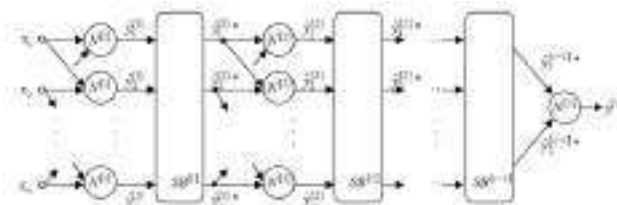


Рисунок 1. Еволюційна гібридна мережа МГУА-нео-фазі

Серед вихідних сигналів обираються найточніші  $y_l^1 \cdot n_1^*$  ( $n_1^* = F$  – так звана свобода вибору) за певним критерієм (здебільшого за середнім квадратом помилки  $\sigma_{y_l^1}^2$ ). Серед цих  $n_1^*$  найкращих результатів першого прихованого шару утворюються попарні комбінації  $y_l^{1*}, y_p^{1*}$ . Вони подаються на другий прихований шар, який утворений нейронами  $N^{[2]}$ . Після навчання цих нейронів вихідні сигнали рівня  $y_l^2$  передаються до блоку селекції  $SB^{[2]}$ , який вибирає  $F$  найкращих нейронів за точністю (наприклад, за значенням  $\sigma_{y_l^2}^2$ ), якщо найкращий сигнал другого шару кращий за найкращий сигнал першого прихованого шару  $y_l^{1*}$ . Інші приховані шари формують сигнали аналогічно. Процес еволюції системи триває до тих пір, поки найкращий сигнал блоку селекції  $SB^{s+1}$  не виявиться гіршим, ніж найкращий сигнал попереднього рівня. Потім ми повертаємося до попереднього рівня та обираємо його найкращий вузловий нейрон  $N^{[s]}$  з вихідним сигналом  $y^{[s]}$ . І рухаючись від цього нейрона (вузла) по його зв'язках назад, послідовно проходячи всі попередні шари, ми врешті отримуємо структуру МГУА-нео-фазі мережі.

Слід зазначити, що таким чином можна побудувати не тільки оптимальну структуру мережі, але й добре навчену мережу завдяки алгоритму МГУА. Крім того, оскільки навчання виконується послідовно шар за шаром, то вдається уникнути проблем високої розмірності та вибухового градієнта.

## 3. ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

### 3.1. Набір даних

В якості вихідних даних було взято корпоративний індекс прибутковості Emerging Markets Bond total Return Index (EMBRI) фондової біржі NASDAQ за період з січня по

серпень 2022 року. Вибірка складалася з примірників, які були розділені на навчальну та тестову підвибірки.

Графік EMBRI представлений на рис. 2.

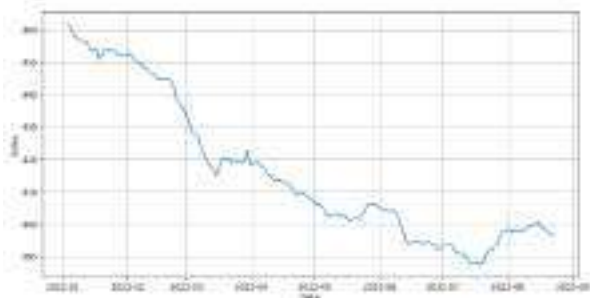


Рисунок 2. Emerging Markets Bond total Return Index (EMBRI)

### 3.2. Експериментальні дослідження гібридних мереж DL

Першу серію експериментів проводили з гібридною мережею глибокого навчання з нео-фазі нейронами у якості вузлів. Під час експериментів змінювалися наступні параметри: співвідношення навчальна/тестова вибірка, кількість входів 3-5, кількість нечітких множин на змінну 3-5 та функції належності (MF): Bell, Gaussian та Triangular. Метою експериментів було знайти оптимальні значення параметрів.

Прогнозний період був прийнятий 5 днів, а показники точності – MSE та MAPE.

У першому експерименті було досліджено Bell MF. Після експерименту були знайдені оптимальні параметри гібридної мережі DL: кількість входів – 3, кількість нечітких множин – 3, співвідношення навчання/тест – 0,8. При цих значеннях параметрів була досягнута найкраща точність на дослідному зразку: MSE = 0,424 та MAPE = 0,155.

Наступні експерименти були проведені для гібридної мережі з Gaussian MF. Після проведення експериментів знайдено оптимальні параметри гібридної мережі DL: кількість входів – 3, кількість нечітких наборів – 4, співвідношення навчання/тест – 0,6. З цими параметрами результати прогнозування представлені в табл. 1.

Таблиця 1. Прогноз із Gaussian MF (кількість входів – 3, кількість нечітких множин – 4, співвідношення навчання/тест – 0,6)

Дата	Реальні значення	Прогноз	MSE	MAPE
17.08.2022	399,15	398,44	0,504	0,178
18.08.2022	399,12	398,17	0,902	0,238
19.08.2022	397,97	398,50	0,281	0,133
22.08.2022	396,82	397,40	0,336	0,146
23.08.2022	396,62	396,93	0,096	0,078
		Мінімум:	0,096	0,078
		Середнє:	0,424	0,155
		Максимум:	0,902	0,238

Наступний експеримент проводився з гібридною мережею DL з Triangular MF. Після проведення експерименту були знайдені оптимальні параметри та структура гібридної мережі: 5 входів, 3 MF та співвідношення 0,8. Після експериментів порівняли точність гібридних мереж з різними MF. Результати представлені на рис. 3.

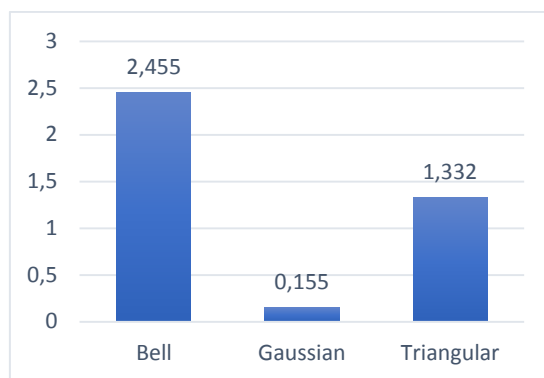


Рисунок 3. Порівняння найкращих значень MAPE (середнє) для різних MF

У наступній серії експериментів були досліджені мережі LSTM. Метою експериментів було знайти оптимальні параметри. Змінювалися такі параметри: кількість входів (3-5), співвідношення навчальна/тестова вибірка (0,6; 0,7; 0,8). Знайдено оптимальні значення параметрів: 3 входи, співвідношення 0,7.

У наступних експериментах було досліджено ефективність найкращих моделей гібридної мережі DL (МГУА-нф) і LSTM та порівняно з різними співвідношеннями навчальна/тестова вибірка. Відповідні результати представлені в табл. 2.

Таблиця 2. MSE для різного співвідношення навчання/тест та 3 входів

Вибірка даних	3 входи		4 входи		5 входів	
	МГУА-нф	LSTM	МГУА-нф	LSTM	МГУА-нф	LSTM
60/40	0,424	4,074	2,894	4,715	5,556	1,961
70/30	1,242	1,378	1,783	2,107	7,414	4,628
80/20	1,248	2,023	3,491	5,412	5,902	2,152

Аналізуючи ці результати, можна зробити висновок, що МГУА-нео-фазі мережа має кращу точність прогнозування на інтервалі 5, ніж LSTM для різних співвідношень.

У наступних експериментах було досліджено ефективність прогнозування гібридних мереж DL та LSTM на різних інтервалах прогнозування.

У табл. 3 представлені результати прогнозування МГУА-нео-фазі мережі та LSTM на інтервалі 7 днів.

Таблиця 3. Кращий прогноз на інтервалі 7 днів – MAPE

Дата	Реальні значення	МГУА-нео-фазі	LSTM
15.08.2022	400,32	0,412	1,806
16.08.2022	400,70	0,317	1,899
17.08.2022	399,15	0,704	1,518
18.08.2022	399,12	0,807	1,511
19.08.2022	397,97	0,859	1,224
22.08.2022	396,82	0,935	0,932
23.08.2022	396,62	0,630	0,877

На рис. 4 представлено порівняння точності прогнозів МГУА-нео-фазі мережі та LSTM на інтервалі 7 днів.

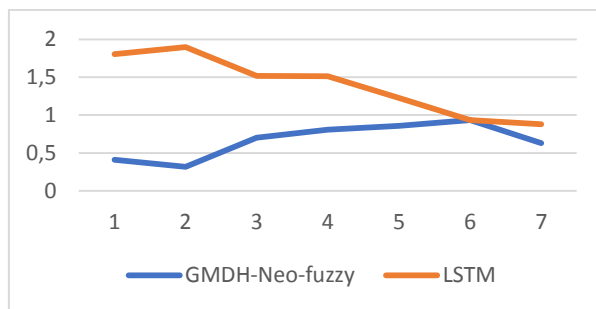


Рисунок 4. Порівняльний графік точності прогнозів на інтервалі 7 днів – MAPE

У наступних експериментах точність прогнозування обох мереж досліджувалася при середньостроковому прогнозуванні з інтервалом 20 днів. Точність за MAPE представлена на рис. 5.

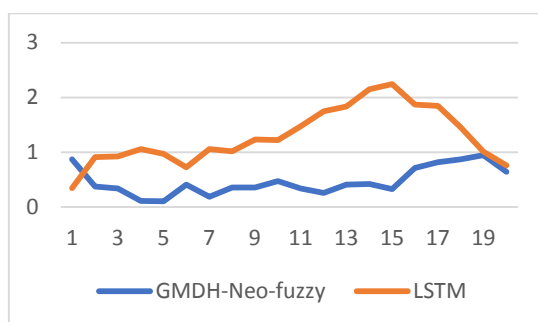


Рисунок 5. Найкращий прогноз з інтервалом 20 днів – MAPE

### 3.3. Порівняння результатів експериментів гібридних мереж DL та LSTM

У підсумкових експериментах порівнювали точність прогнозування гібридної мережі МГУА-нео-фазі та LSTM на різних інтервалах прогнозування (короткострокових і середньострокових). Відповідні результати за MSE та MAPE представлені в табл. 4.

Таблиця 4. Значення MSE та MAPE (середні) для різних інтервалів

Інтервал	Мережа	5 днів		7 днів		20 днів	
		MSE	MAPE	MSE	MAPE	MSE	MAPE
3	МГУА-нео-фазі	0,319	0,094	0,678	0,192	1,251	0,262
	LSTM	0,702	0,201	0,408	0,127	2,374	0,349
5	МГУА-нео-фазі	1,178	0,111	2,819	0,337	0,953	0,224
	LSTM	2,674	0,316	3,054	0,358	2,374	0,352
7	МГУА-нео-фазі	0,238	0,231	2,714	0,475	0,863	0,203
	LSTM	5,888	0,489	6,325	0,510	2,983	0,364
20	МГУА-нео-фазі	2,241	0,392	1,238	0,368	0,595	0,172
	LSTM	6,182	0,453	4,178	0,408	3,134	0,416

Аналіз цих результатів показує, що в цілому гібридна мережа DL має кращу точність, ніж LSTM на різних коротких та середніх інтервалах прогнозування.

## 4. ВИСНОВОК

У роботі розглянуто задачу прогнозування на фінансовому ринку з різними інтервалами прогнозування (короткострокове та середньострокове прогнозування). Для її вирішення було запропоновано застосувати гібридні мережі глибокого навчання (DL) на основі МГУА та LSTM.

Експериментальні дослідження проводились над задачею прогнозування Emerging Markets Bond Total Return Index (EMBRI) на фондовій біржі NASDAQ за період з січня по серпень 2022 року.

Під час експериментів проведено оптимізацію параметрів LSTM та гібридних мереж. Методом МГУА побудовано оптимальну структуру гібридної мережі DL.

Експериментальні дослідження оптимізованих LSTM та гібридних мереж проводились на різних інтервалах прогнозування та порівнювалася їх точність.

У результаті було встановлено, що застосування гібридних мереж DL має набагато кращу точність, ніж LSTM, у задачах короткострокового та середньострокового прогнозування на фондових біржах.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. S. Hochreiter and J. Schmidhuber Long short-term memory, *Neural Computation*, vol. 9, 1997, pp. 1735–1780. doi: 10.1162/neco.1997.9.8.1735.
2. B. Hammer, On the approximation capability of recurrent neural networks, *Neurocomputing*, vol. 31, 1998, pp. 107–123. doi: 10.1016/S0925-2312(99)00174-5.
3. C. Olah Understanding LSTM networks, 2020. Available from: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>.
4. A. Graves Generating sequences with recurrent neural networks, *CoRR*, vol. abs/1308.0850, 2013. doi: 10.48550/arXiv.1308.0850.
5. A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, Verlag, Berlin, Heidelberg: Springer, 2012. doi: 10.1007/978-3-642-24797-2.
6. S.S. Haykin, *Neural networks: a comprehensive foundation*, 2nd ed. Upper Saddle River, N.J: Prentice Hall, 1999.
7. Ossovsky S. *Neural networks for information processing*, transl. from Polish.–M. – Finance and Statistics, 2002 – 344p.
8. Wang F. *Neural Networks Genetic Algorithms and Fuzzy Logic for Forecasting* // Proc. Intern. Conf. Advanced Trading Technologies. – New York, 1992, pp. 504–532.
9. Yamakawa T., Uchino E., Miki T., Kusanagi H. A neo-fuzzy neuron and its applications to system identification and prediction of the system behavior // Proc. 2nd Intern. Conf. Fuzzy Logic and Neural Networks LIZUKA-92. – Lizuka, 1992, pp. 477–483.
10. I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT PRESS, 2016. Available from: <http://www.deeplearningbook.org>.
11. Yuriy Zaychenko, Yevgeniy Bodyanskiy, Oleksii Tyshchenko, Olena Boiko, Galib Hamidov Hybrid GMDH-neuro-fuzzy system and its training scheme. *Int. Journal Information theories and Applications*, 2018. vol.24, Number 2, pp. 156–172.
12. Yu. Zaychenko, Galib Hamidov The Hybrid Deep Learning GMDH-neo-fuzzy Neural Network and Its Applications. *Proceedings of 13-th IEEE Intern Conference Application of Information and Communication Technologies-AICT2019*. 23–25 October 2019, Baku, pp. 72–77.
13. Evgeniy Bodyanskiy, Yuriy Zaychenko, Olena Boiko, Galib Hamidov, Anna Zelikman Structure Optimization and Investigations of Hybrid GMDH-Neo-fuzzy Neural Networks in Forecasting Problems. *System Analysis & Intelligent Computing*. Ed. Michael Zgurovsky, Natalia Pankratova. *Book Studies in Computational Intelligence, SCI*, vol.1022. Springer, 2022, pp. 209–228.
14. Yuriy Zaychenko and Oleksii Kuzmenko Investigation of hybrid deep learning networks and LSTM in the short term forecasting in financial sphere, *IEEE 17th Int. Conference CSIT*, 11 November 2022, 4 p.

# РОЗПІЗНАВАННЯ ПОЖЕЖІ ЗА ДОПОМОГОЮ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ

Маркін І.Д.<sup>1</sup>, Шаповал Н.В.<sup>2</sup>

КПІ ім. Ігоря Сікорського, Київ, Україна

<sup>1</sup>ivanmarkin2019@gmail.com, <sup>2</sup>shovgun@gmail.com

**У роботі розглянуто проблему виявлення пожежі за допомогою сучасних технологій та алгоритмів машинного зору. Для розв'язання поставленої задачі запропоновано використати згорткову нейронну мережу. Побудована згорткова нейронна мережа, проведено експерименти з побудованою нейронною мережею на вибірці зображень.**

**Ключові слова:** згорткова нейронна мережа, CNN, виявлення пожежі, класифікація.

## 1. ВСТУП

Виявлення пожежі на ранній стадії до її переростання в катастрофічну подію має вирішальне значення для нейтралізації пожеж та порятунку життя та майна людей. Через швидке поширення пожежі важливо виявити його на стадії перших проявів вогню.

Незважаючи на важливість цієї проблеми, вона залишається невирішеною на даний момент. Датчики, основним принципом яких є виявлення диму, добре працюють, коли пожежа триває протягом певного часу і утворилася достатня кількість диму для спрацьовування сигналізації [1]. Ці пристрої не можуть бути використані у великих масштабах на відкритому повітрі, наприклад, у лісі. Супутникові системи, стаціонарні віддалені камери та пілотовані літаки - це звичайні технології дистанційного моніторингу пожеж, проте ці технології мають низку обмежень з точки зору вартості, тимчасового та просторового дозволу.

Також для вирішення поставленої задачі пропонується використання згорткових нейронних мереж. На даний момент, технології нейронних мереж досягли високого рівня розвитку і вже дозволяють знаходити рішення специфічним завданням, працюючи за принципом, схожим на процес прийняття рішення людиною. Використання таких методів на даному етапі розвитку технологій не може зробити прийняття рішень повністю автоматизованим, проте воно суттєво знижує вплив людського фактора та забезпечує високу результативність за менших витрат часу та зусиль.

Метою даної роботи є розробка програмного забезпечення для виявлення лісової пожежі за допомогою нейронних мереж.

## 2. ОПИС ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ

Відповідно до принципу алгоритмів виявлення об'єктів, алгоритм виявлення пожежі на основі згорткових нейронних мереж зображено на (Рис. 1) [2]. Згорткова нейронна мережа має функції пропозицій регіону, виділення ознак і класифікації. По-перше, ЗНМ приймає зображення як вхідні дані та виводить пропозиції регіонів шляхом згортання, об'єднання тощо. По-друге, система виявлення об'єктів на основі регіону визначає наявність чи відсутність вогню в регіонах пропозицій за допомогою згорткових шарів, шарів об'єднання, повноз'єднаних шарів. і т.д.

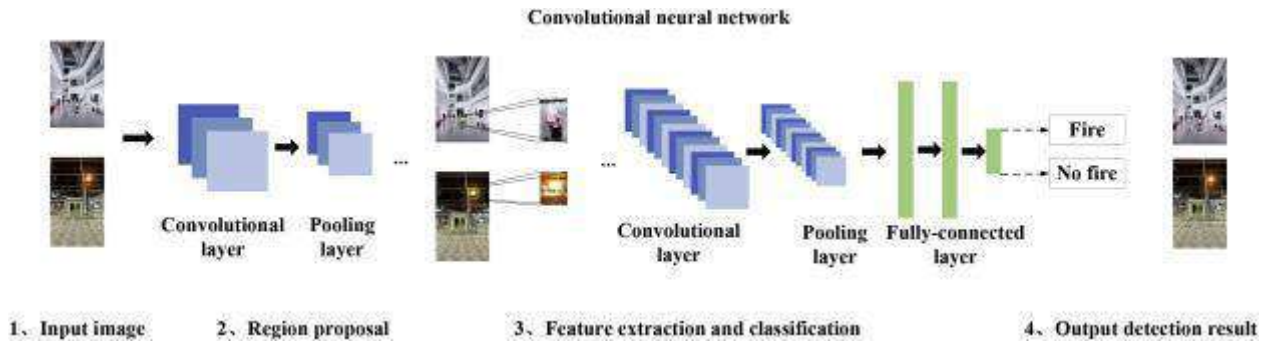


Рисунок. 1. Блок-схема алгоритмів виявлення пожежі за зображенням на основі CNN виявлення.

Згортковий рівень є основною частиною CNN. На відміну від інших нейронних мереж, які використовують ваги зв'язків і зважені суми, згортковий рівень використовує фільтри перетворення зображень, які називаються ядром згортки, для генерації карт властивостей вихідних зображень. Згортковий шар — це набір ядер згортки. Ядро згортки ковзає по зображеннях і обчислює новий піксель за зваженою сумою пікселів, над якими воно плаває, щоб створити карту функцій. Карта функцій відображає особливості вихідного зображення. Формула розрахунку шару згортки:

$$y = \sum_{j=0}^{J-1} \sum_{i=0}^{I-1} w_{ij} x_{m+1, n+j} + b, (0 \leq m \leq M, 0 \leq n \leq N)$$

де  $x$  позначає вхідне зображення певного розміру,  $w$  позначає ядро згортки певного розміру,  $b$  позначає зміщення, а  $y$  позначає вихідні карти ознак. На практиці значення  $w$  і  $b$  визначається шляхом навчання.

На рисунку 2 показано 32 ядра першого згорткового шару в Inception Resnet V2 [3] (сучасна ЗНМ) і 32 карти функцій зображення пожежі, згенерованого цими ядрами. Кількість карт ознак і ядер згортки однакова. Наприклад, у цьому шарі є три ядра згортки, тому генеруються три карти функцій. Колір пікселів відображає ступінь активації. Білі пікселі в певному місці на карті функцій вказують на те, що пікселі сильно позитивно активовані в тій же позиції на вихідному зображенні. Чорні пікселі вказують на сильно негативні активації. Сірі пікселі означають несильні активації. Порівняно з вихідним зображенням, карта функцій, згенерована згортковим ядром 14 цього шару, активована на краях. Темні/світлі області активуються позитивно на верхніх/нижніх краях, а світлі/темні області активуються негативно на верхніх/нижніх краях. Карта характеристик, створена згортковим ядром 26, активується на помаранчевих пікселях, оскільки більш білі пікселі на карті відповідають помаранчевим областям на вихідному зображенні. Це вказує на те, що ядра на попередніх рівнях в основному вивчають і витягують прості функції, такі як колір, краї тощо. Аналізуючи ці карти функцій, виявлено, що прості функції не можуть розрізнити пожежу та збурення, коли сцени складні, а також багато перешкод. Таким чином, необхідно розробити алгоритми виявлення пожежі за зображенням, які можуть отримувати складні характеристики зображення для виявлення пожежі в реальних умовах. Глибокі згорткові нейронні мережі кращі в цьому аспекті. На рис. 3 показані зразки ядер у першому, третьому та шостому шарі згортки Inception Resnet V2. Це вказує на те, що мережі виділяють складніші функції на наступних рівнях. Тому для вилучення складних характеристик зображення потрібна глибока мережа.

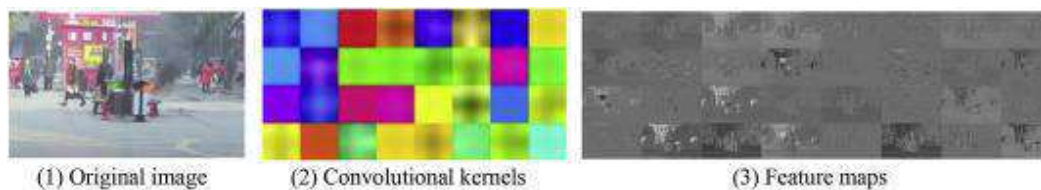


Рисунок 2. Об'єкти, виділені першим згортковим шаром

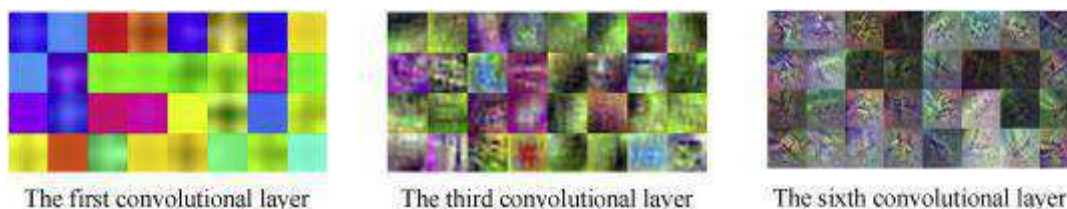


Рисунок 3. Зразки ядер у деяких згорткових шарах

### 3. ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

#### 3.1. Вхідні дані

Набір даних був створений під час конкурсу NASA Space Apps Challenge у 2018 році. Метою було використання набору даних для розробки моделі, яка може розпізнавати зображення вогнем. Було зібрано дані які містять вогонь (зображення вогню), і звичайні зображення (зображення без вогню).

Дані поділено на 2 папки, папка fireimages містить 755 зображень пожежі (Рис. 4), деякі з них містять сильний дим, інша – зображення без пожеж (Рис. 5), які містять 244 зображення природи (наприклад: ліс, дерево, трава, річка, люди, туманний ліс, озеро, тварина, дорога та водоспад).



Рисунок 4. Приклад зображення з вогнем



Рисунок 5. Приклад зображення без вогню

Дані будуть розділені на три різні категорії: навчання, перевірка та тестування. Навчальні дані використовуватимуться для навчання моделі CNN глибокого навчання, а її параметри будуть точно налаштовані за допомогою даних перевірки. Нарешті, продуктивність даних буде оцінена за допомогою тестових даних (даних, які модель раніше не бачила).

#### 3.2. Побудова згорткової нейронної мережі

Загальна схема побудованої нейронної мережі зображена на рисунку 6.

Тепер, коли ми вказали архітектуру моделі, ми скопіюємо модель для навчання. Для цього нам потрібно вказати функцію втрат, оптимізатор і метрику (як ми будемо оцінювати

продуктивність моделі). В якості оптимізатора використовуватимуться градієнтний спуск і стохастичний градієнтний спуск, в якості метрики – точність (accuracy).



Рисунок 6. Структура побудованої мережі

### 3.3. Аналіз результатів

Нижче наведено графіки точності (Рис. 7) та функції втрат (Рис. 8) для розробленої нейронної мережі, а також матриця невідповідностей (Рис. 9).

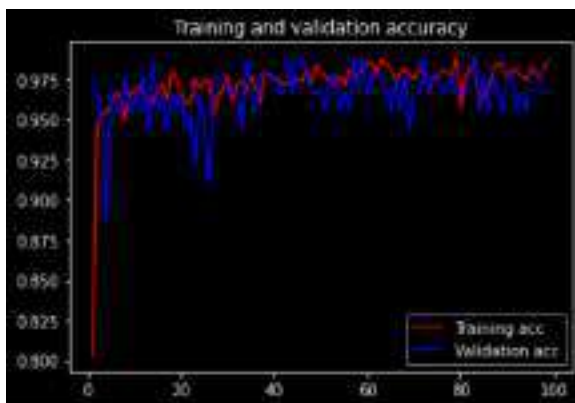


Рисунок 7. Точність

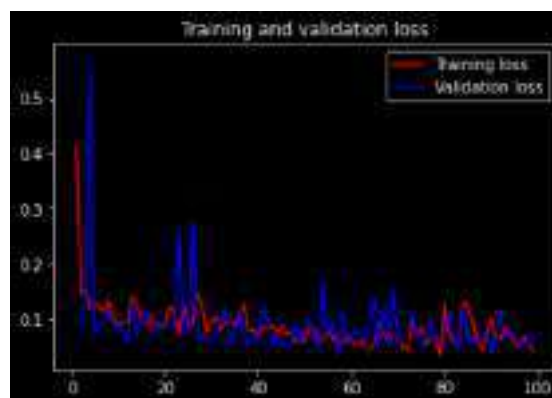


Рисунок 8. Функція втрат

Ми можемо спостерігати, як точність покращується з часом, зрештою стабілізуючись. Відповідно, втрати з часом зменшуються. Подібні графіки можуть допомогти діагностувати перенавчання. Якби ми побачили висхідну криву функції втрати (U-подібна форма на графіку), ми б запідозрили, що модель починає запам'ятовувати тестовий набір і погано узагальнювала б нові дані.



Рисунок 9. Матриця невідповідностей

Запропонована нейронна мережа показала високу точність у виявленні пожежі на зображенні. Досягнута точність 0,95.

### **ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. Калиев Д. И., Швец О. Я. Свёрточные нейронные сети для решения задач обнаружения пожаров по данным аэрофотосъемки // Программные системы: теория и приложения. 2022. Т. 13, №1(52). С. 195–213. [http://psta.psiras.ru/read/psta2022\\_1\\_195-213.pdf](http://psta.psiras.ru/read/psta2022_1_195-213.pdf)
2. Pu Li, Wangda Zhao, Image fire detection algorithms based on convolutional neural networks, Case Studies in Thermal Engineering, Volume 19, 2020, pp 2-4.
3. S. Bianco, R. Cadene, L. Ceona, P. Napoletano Benchmark analysis of representative deep neural network architectures pp. 64270-64277.

# МЕТОДИ ПРОГНОЗУВАННЯ ІНДЕКСУ АКЦІЙ НА ОСНОВІ МЕХАНІЗМІВ ШТУЧНОГО ІНТЕЛЕКТУ

Міщенко Д.В.<sup>1</sup>, Мухін В.Є.<sup>2</sup>

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

<sup>1</sup> mischenko\_dasha@ukr.net, <sup>2</sup> v.mukhin@kpi.ua [0000-0002-1206-9131]

Протягом останніх десятиліть об'єктом багатьох досліджень є прогнозування фондових ринків, яке, незважаючи на свою заплутаність, динамічність та нестабільність, є надзвичайно складною задачею. Для побудови ефективної моделі прогнозування необхідно враховувати різноманітні фактори, величезні обсяги даних та тривіальне співвідношення сигнал/шум, що значно ускладнює задачу прогнозування поведінки цін на фондовому ринку. Тим не менш, існує велика кількість різноманітних підходів, спрямованих на досягнення цієї мети. Метою роботи є аналіз фінансових даних, підбір моделей для прогнозування, реалізація методів прогнозування на основі механізмів штучного інтелекту та вибір найкращого методу. Результатом дослідження є програмний продукт, що видає поденний прогноз індексу акцій S&P 500.

**Ключові слова:** фондові ринки, методи прогнозування, нейронні мережі, S&P 500, авторегресія, індекси акцій.

## 1. ВСТУП

Фондовий ринок – це сукупність бірж, через які випускають, купують та продають акції публічних компаній.

Роль фондового ринку полягає в тому, щоб забезпечити компаніям спосіб залучення капіталу шляхом продажу акцій державним інвесторам. У той же час, фондовий ринок дозволяє приватним інвесторам купувати акції публічних компаній і ставати співвласниками їх бізнесу. Сукупна вартість всього фондового ринку часто відстежується та відображається за допомогою ринкових індексів, таких як промисловий індекс Доу-Джонса (Dow Jones Industrial Average) та індекс S&P 500 (S&P 500) [1].

Коли приватна компанія бажає стати публічною, вона, як правило, здійснює первинне публічне розміщення акцій (ІРО). Під час процесу ІРО компанія продає акції публічним інвесторам, щоб зібрати гроші, необхідні для погашення боргу або інвестування у свій бізнес. Після завершення ІРО ці акції починають торгуватися на одній або декількох фондових біржах, які складають фондовий ринок.

Інвестори купують акції в надії на те, що компанія, яка продає ці акції, з часом зросте і стане більш цінною, тим самим збільшуючи ціну кожної акції. Ціни на акції визначаються економічним законом попиту та пропозиції, і їх ціни часто коливаються щодня на основі змін у попиті інвесторів [2].

## 2. АКТУАЛЬНІСТЬ ПРОГНОЗУВАННЯ ФІНАНСОВИХ ДАНИХ

Прогнозування фінансових і економічних часових рядів завжди було і залишається важким завданням через його чутливість до політичних, економічних і соціальних факторів. Хорошими прикладами є пандемія у 2020 році та повномасштабне вторгнення Росії на територію України. У першому випадку, з середини лютого до середини березня 2020 ринки акцій у Європейському Союзі та Сполучених Штатах впали на 30 відсотків. Зміну очікуваного зростання дивідендів на 1 рік можна побачити на рисунку 1 та зміну очікуваного зростання ВВП на 1 рік, де А. – локдаун в Ухані, В. – карантин в Італії, С – заборона США на поїздки до ЄС та D – надзвичайний стан в США.

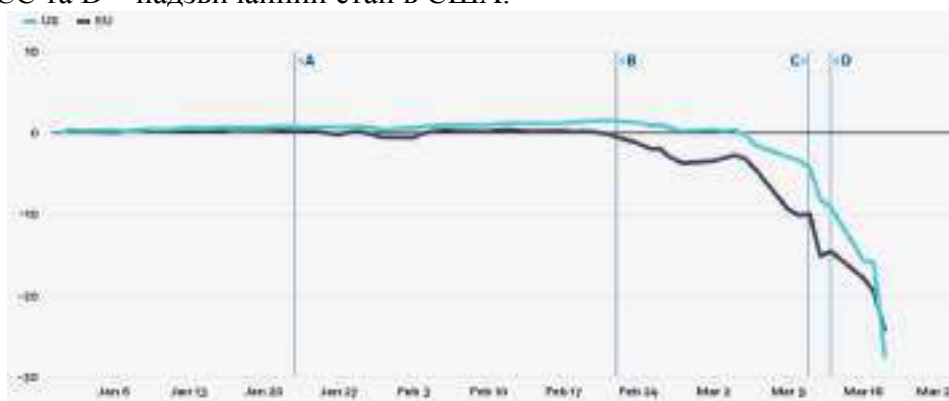


Рисунок 1. Зміна очікуваного зростання дивідендів на 1 рік [3]

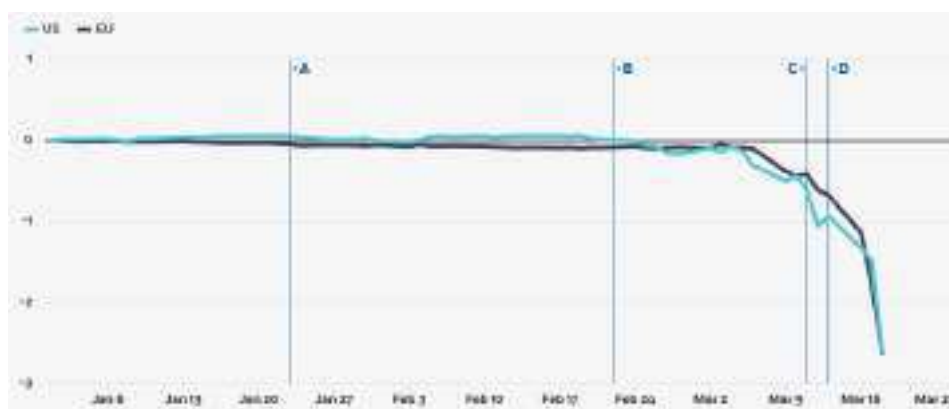


Рисунок 2. Зміна очікуваного зростання ВВП на 1 рік [3]

У другому випадку війна спричинила серйозні коливання ринку і привела до певної нестабільності. Ринки здебільшого проігнорували нещодавні конфлікти, пов'язані з Близьким Сходом та Іраном. Однак війна між Росією, Україною та союзниками по НАТО може мати серйозніший вплив, особливо на ціни на нафту та інші товари. Хоча фондові ринки відновилися до рівня до вторгнення лише за кілька тижнів після вторгнення Росії в Україну, перепади спричинили серйозні фінансові втрати для багатьох інвесторів.

У США індекс S&P 500 впав більш ніж на 7% за кілька днів і тижнів відразу після вторгнення, оскільки США та інші країни посилили жорсткі економічні санкції проти Росії, а інвестори стурбовані впливом цін на сировину. Але через місяць ринки відновилися, і S&P торгувався на рівні, вищому, ніж до вторгнення, навіть якщо ціна нафти залишалася вище 100 доларів за барель. На Рис. 3 зображений графік стану індексу S&P 500 як почалось повномасштабне вторгнення Росії в Україну та місяць після.

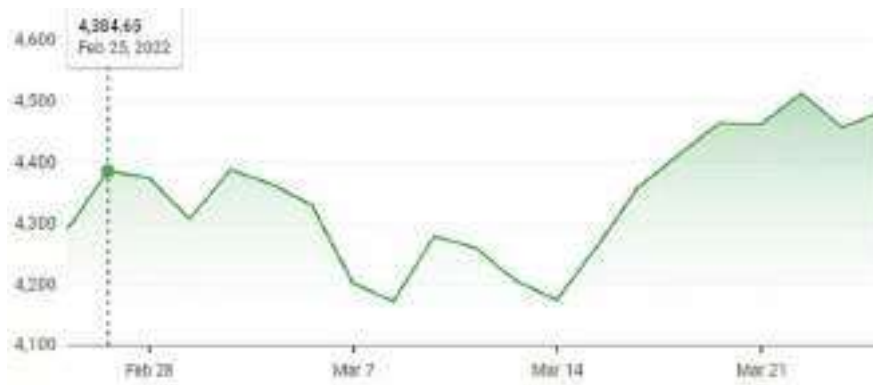


Рисунок 3. Ціна індексу S&P 500 з моменту вторгнення Росії в Україну та місяць потому [4]

Та навіть враховуючи ці випадки люди, які інвестують у фінансові ринки та обмін валюти, зазвичай шукають надійні моделі, які можуть гарантувати їм максимізацію свого профілю та мінімізацію втрат, наскільки це можливо в умовах постійної соціальної і економічної нестабільності.

### 3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

В рамках дослідження було прийняте рішення побудувати моделі авторегресії, довгої короткострокової пам'яті та в подальшому розробка методу прогнозування на основі механізму уваги (Attention network) для порівняння та вибору найкращого методу для прогнозування.

Для дослідження та побудови прогнозу було обрано дані індексу акцій S&P 500, що є зваженим за ринковою капіталізацією індексом 500 провідних публічних компаній США, а саме за період з 31.12.2019 до 31.12.2021.

Спочатку було проведено низку маніпуляцій над вхідними даними, а саме – логарифмування, визначення порядку лагу регресії для побудови вектору ознак та розділення вибірки на навчальну та тестову з подальшою нормалізацією (стандартизацією) ряду.

Далі було реалізовано модель авторегресії порядку 3, у таблиці 1 представлено значення похибок прогнозу даної моделі. Для оцінки якості прогнозу було обрано метрики  $R^2$ , MSE, MAE та MAPE[5].

Таблиця 1. Рначення похибок прогнозу моделі лінійної регресії ARMA

	$R^2$	MSE	MAE	MAPE
Тренувальна	0.979615	0.000330	0.011232	0.001396
Тестова	0.957944	0.000067	0.006250	0.000745

Порівняємо прогноз моделі та справднні дані на рисунку 4. На верхньому графіку відображена візуалізація справжніх даних, а на нижньому отриманий прогноз за допомогою моделі лінійної регресії.

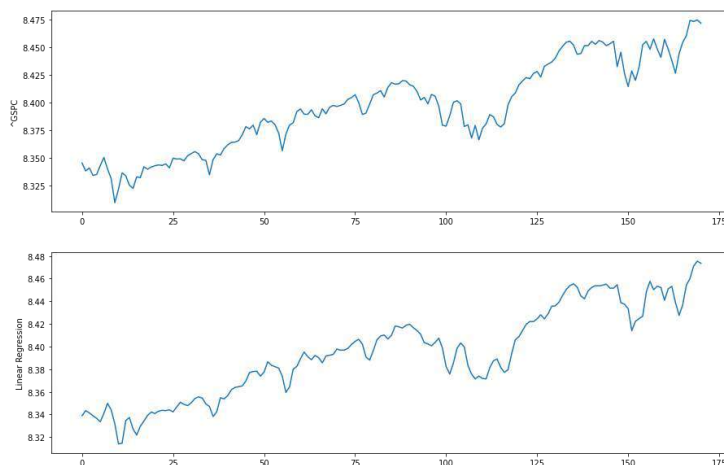


Рисунок 4. Прогноз лінійної авторегресійної моделі в порівнянні зі справжніми даними

Далі було реалізовано модель довгої короткострокової пам'яті в різних конфігураціях для знаходження найкращої моделі прогнозування ряду. У таблиці 2 показані значення похибок прогнозу LSTM моделі для різних конфігурацій.

Таблиця 1. Значення похибок прогнозу моделі LSTM

Конфігурації/ похибка	Датасет	$R^2$	MSE	MAE	MAPE
lstm_64-64_lstm_layers_128- 128_dense_layers	Train	0.965836	0.000553	0.016932	0.210003
	Test	-28.4967	0.046998	0.176939	2.100744
lstm_64-64_lstm_layers_64- 64_dense_layers	Train	0.940538	0.000962	0.024145	0.298472
	Test	-53.6261	0.087038	0.24359	2.892339
lstm_64-64_lstm_layers_32- 32_dense_layers	Train	0.917011	0.001343	0.026719	0.331763
	Test	-10.5617	0.018422	0.114673	1.361971
lstm_64-64_lstm_layers_ 128_dense_layers	Train	0.730207	0.004365	0.052797	0.653975
	Test	-23.7398	0.039419	0.178428	2.12056
lstm_64- 64_lstm_layers_64_dense_layers	Train	0.759237	0.003895	0.050692	0.627004
	Test	-41.0924	0.067067	0.225963	2.684566
lstm_64- 64_lstm_layers_32_dense_layers	Train	0.739095	0.004221	0.046994	0.584426
	Test	-17.2182	0.029028	0.143491	1.704148
lstm_32-32_lstm_layers_128- 128_dense_layers	Train	0.95694	0.000697	0.017505	0.217781
	Test	-9.48553	0.016707	0.108819	1.292486
lstm_32-32_lstm_layers_64- 64_dense_layers	Train	0.959036	0.000663	0.016207	0.202062
	Test	-3.75888	0.007583	0.069879	0.829604
lstm_32-32_lstm_layers_32- 32_dense_layers	Train	0.858923	0.002283	0.040241	0.496908
	Test	-40.314	0.065827	0.222979	2.649009
lstm_32-32_lstm_layers_128_ dense_layers	Train	0.897496	0.001658	0.033583	0.414701
	Test	-40.7559	0.066531	0.223151	2.650915
lstm_32- 32_lstm_layers_64_dense_layers	Train	0.914537	0.001383	0.028771	0.354616
	Test	-19.765	0.033086	0.164021	1.949447
lstm_32- 32_lstm_layers_32_dense_layers	Train	0.812266	0.003037	0.033089	0.413834
	Test	-3.43403	0.007065	0.071202	0.845728
lstm_128_lstm_layers_128- 128_dense_layers	Train	0.860072	0.002264	0.034473	0.427745
	Test	-61.991	0.100366	0.208407	2.470416

lstm_128_lstm_layers_64-64_dense_layers	Train	0.736589	0.004262	0.053753	0.664874
	Test	-12.6901	0.021813	0.13365	1.588622
lstm_128_lstm_layers_32-32_dense_layers	Train	0.841774	0.00256	0.037573	0.465034
	Test	-761.028	1.214169	0.879389	10.43801
lstm_128_lstm_layers_128_dense_layers	Train	0.588498	0.006658	0.059676	0.740511
	Test	-5.19191	0.009866	0.09324	1.108671
lstm_128_lstm_layers_64_dense_layers	Train	0.530279	0.0076	0.075688	0.93088
	Test	-251.953	0.40304	0.53179	6.315124
lstm_128_lstm_layers_32_dense_layers	Train	0.856315	0.002325	0.038668	0.476722
	Test	-44.1063	0.07187	0.23511	2.793376
lstm_64_lstm_layers_128-128_dense_layers	Train	0.817728	0.002949	0.045776	0.566415
	Test	-53.5476	0.086913	0.236663	2.809535
lstm_64_lstm_layers_64-64_dense_layers	Train	0.684549	0.005104	0.058813	0.726362
	Test	-146.799	0.235493	0.397531	4.719956
lstm_64_lstm_layers_32-32_dense_layers	Train	0.914261	0.001387	0.030858	0.380005
	Test	-279.403	0.446777	0.556264	6.605339
lstm_64_lstm_layers_128_dense_layers	Train	0.798097	0.003267	0.048157	0.593979
	Test	-296.757	0.474427	0.568438	6.749331
lstm_64_lstm_layers_64_dense_layers	Train	0.529415	0.007614	0.073541	0.90539
	Test	-271.97	0.434933	0.575537	6.837614
lstm_64_lstm_layers_32_dense_layers	Train	-0.82582	0.029541	0.147431	1.825889
	Test	-31.9915	0.052567	0.213531	2.538882
lstm_32_lstm_layers_128-128_dense_layers	Train	0.882336	0.001904	0.034477	0.427528
	Test	-103.438	0.166405	0.296107	3.512346
lstm_32_lstm_layers_64-64_dense_layers	Train	0.684455	0.005105	0.061368	0.757598
	Test	-103.955	0.167229	0.34645	4.11472
lstm_32_lstm_layers_32-32_dense_layers	Train	0.761386	0.003861	0.047534	0.587962
	Test	-53.5055	0.086846	0.257672	3.06136
lstm_32_lstm_layers_128_dense_layers	Train	-0.09325	0.017688	0.117866	1.454407
	Test	-91.1195	0.146778	0.345196	4.102582
lstm_32_lstm_layers_64_dense_layers	Train	0.872073	0.00207	0.025348	0.316033
	Test	0.786861	0.00034	0.014225	0.169246
lstm_32_lstm_layers_32_dense_layers	Train	-0.28382	0.020772	0.085497	1.067839
	Test	-5.26437	0.009981	0.097304	1.157964

З отриманих результатів робимо висновок, що найкраще себе показала модель з однією LSTM коміркою розміром 32 та одним прихованим повнозв'язним шаром розміром 64.

Порівнюємо прогноз реалізованих моделей та справжні дані на рисунку 5. На верхньому графіку відображена візуалізація справжніх даних, далі – лінійна регресія та на нижньому LSTM.

Отже, виходячи з візуалізацій та метрик якості моделей можна зробити висновок, що найкраще для прогнозування обраних даних показала модель довгої короткострокової пам'яті.

#### 4. ВИСНОВКИ

Мільйони людей по всьому світу щодня інвестують у фондовий ринок. Хороша модель прогнозування цін на акції допоможе інвесторам, керівництву та особам, які приймають рішення, у прийнятті правильних та ефективних рішень.

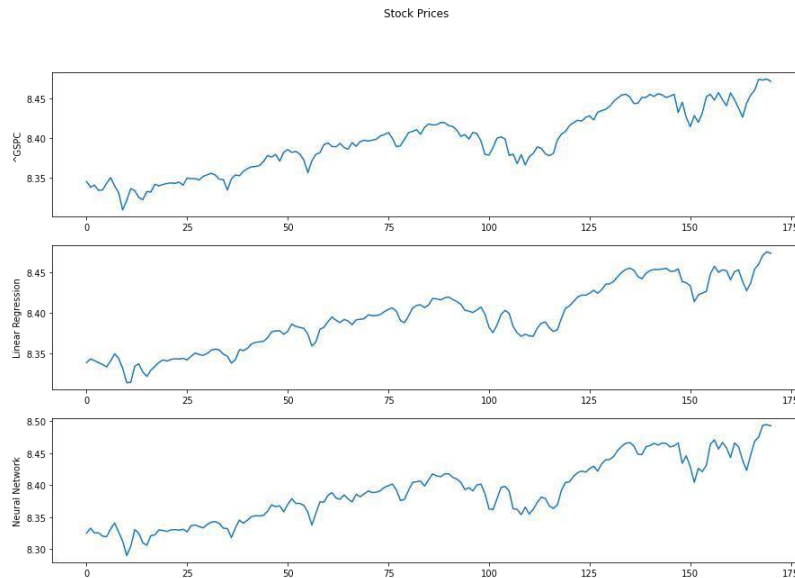


Рисунок 5. Прогноз реалізованих моделей в порівнянні зі справжніми даними

Метою дослідження було проаналізувати дані та реалізувати кілька методів прогнозування на основі механізмів штучного інтелекту та обрати найкращий з них для подальшого використання при вирішенні задачі прогнозування фінансових показників у короткостроковій перспективі.

У ході проведення дослідження були вивчені властивості досліджуваного часового ряду, а саме – показника індексу S&P 500 на період з 31.12.2019 до 31.12.2021, після чого було встановлено нестационарність часового ряду та зроблено висновок щодо автокорельованості ряду та доцільності застосування авторегресійних моделей, в якості референсних моделей.

Після виконання низки маніпуляцій над вхідними даними були проведені експерименти з побудовою та тестування моделей глибокого навчання з метою визначення найкращої моделі за показниками  $R^2$ , MSE, MAE та MAPE.

Як результат дослідження було встановлено, що модель з однією LSTM коміркою розміром 32 та одним прихованим пов'язаним шаром розміром 64 має найкращі метрики як серед досліджуваних моделей глибокого навчання, так і в порівнянні з авторегресійними лінійними моделями.

Результат даної роботи можна застосувати при вирішенні подібних задач короткострокового прогнозування нестационарних часових рядів.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. S&P 500 Index: What It's for and Why It's Important in Investing URL: <https://www.investopedia.com/terms/s/sp500.asp> (дата звернення: 11.11.2022)
2. What Is the Stock Market and How Does It Work? URL: <https://learn.financestrategists.com/finance-terms/stock-market/> (дата звернення: 11.11.2022)
3. How coronavirus affects stock prices and growth expectations. URL: <https://www.chicagobooth.edu/review/how-coronavirus-affects-stock-prices-and-growth-expectations> (дата звернення: 10.11.2022)
4. How War Affects the Modern Stock Market. URL: <https://www.investopedia.com/solving-the-war-puzzle-4780889> (дата звернення: 11.11.2022)
5. A guide on regression error metrics. URL: <https://sefidian.com/2022/06/18/a-guide-on-regression-error-metrics-with-python-code/> (Дата звернення: 09.11.2022).

# **ЗОРОВИЙ ТРАНСФОРМЕР ДЛЯ ЗАДАЧІ КЛАСИФІКАЦІЇ РАКУ ШКІРИ**

Нікітін В.О.<sup>1</sup>, Шаповал Н.В.<sup>2</sup>

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

<sup>1</sup>nvo63911@gmail.com,

<sup>2</sup>shovgun@gmail.com

**Автоматизація класифікації раку шкіри – актуальна проблема сучасності. Метою цієї роботи є класифікація зображень ракових пухлин на шкірі за допомогою зорового трансформера та аналіз можливості використання моделі на основі зорового трансформера як ефективного класифікатора патологічних утворень на шкірі.**

**На основі існуючих досліджень можна зробити припущення, що зоровий трансформер може мати кращі шанси для цієї задачі ніж традиційні методи, що використовують здебільшого згорткові нейронні мережі**

**Ключові слова: класифікація зображень, ViT**

## **1. ВСТУП**

Сьогодні в задачах обробки природної мови (NLP), трансформери стали стандартними моделями (наприклад, BERT, GPT-3 і т. д.). Тим не менш, можливості трансформерів у задачах комп'ютерного зору все ще є досить обмеженими. Більшість дослідників використовують згорткові мережі напряму, чи так чи інакше використовують згорткові шари у мережах

Із зображеннями трансформер використовувати дещо важче. Картинка складається з тисяч чи мільйонів пікселів. Таким чином, якщо трансформер виконує попарну операцію між кожною парою пікселів на зображенні, складність виростає до складно обчислювального рівня. Тому для картинок використовують дещо іншу форму локальної функції уваги замість глобальної.

Автори зорового трансформера використовують функцію уваги, не для всієї картинки, а для багатьох вхідних частин картинки. Тож, на першому етапі картинка розбивається на підкартинки розміром 16 на 16.

В цій роботі реалізований зоровий трансформер для задачі класифікації раку шкіри на прикладі меланому. Мета роботи полягає у тому, щоб побудувати ефективну модель класифікації на основі зорового трансформера за якої можна було б оцінювати чи варто людині із пухлиною на шкірі звернутися до лікаря та оцінити доцільність використання зорового трансформера, як моделі для вирішення цієї задачі [1].

## **2. ЗОРОВИЙ ТРАНСФОРМЕР**

Автори зорового трансформера використовують функцію уваги, не для всієї картинки, а для багатьох вхідних частин картинки. Тож, на першому етапі картинка розбивається на підкартинки розміром 16 на 16.



Рисунок 1. Розбиття картинки на підкартинки [2]

Потім створюється послідовність із цих зображень. Через нумерацію підкартинок моделі надається інформація про їх позиціонування. Це відбувається за допомогою пошукової таблиці, що зберігає вектор для кожного номеру позиції зображення. Тож, для першої підкартинки вектор з таблиці подається у модель з першим зображенням. Аналогічно з іншими підкартинками.

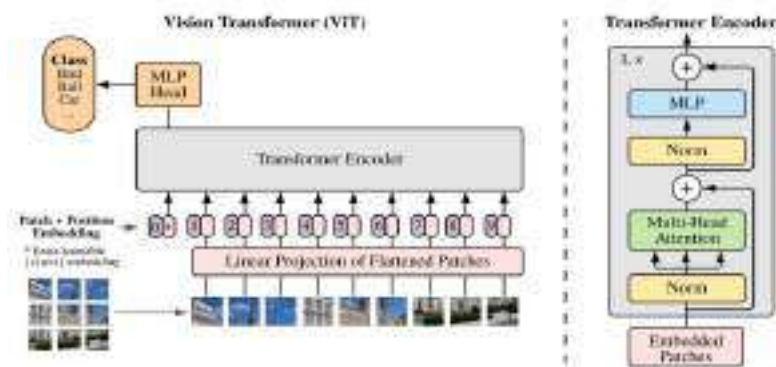


Рисунок 2. Загальна схема зорового трансформера [3]

Щоб створити надійну систему класифікації раку шкіри, потрібні різноманітні дані із усіма видами дерматологічних зображень. Оскільки потреба в ресурсах медичної візуалізації в академічних колах зростає, все більше і більше наборів даних стають загальнодоступними. Далі представлені теоретичні відомості про популярні та найбільш змістовні набори даних, що використовуються чи можуть бути потенційно використані для вирішення поставленої задачі.

Набір даних HAM100004 [4] було зібрано організацією International Skin Imaging Collaboration (ISIC) для вирішення проблеми дисбалансу та обмеження даних у наборах даних про захворювання шкіри. Він містить 10015 дермоскопічних зображень із сімома репрезентативними захворюваннями пігментних уражень шкіри. В роботі вони розділилися тільки на меланомні та здорові [5].

### 3. ТРЕНУВАННЯ МОДЕЛІ ТА РЕЗУЛЬТАТИ ГЕНЕРАЦІЇ

Зоровий трансформер навчався протягом 65 епох. В якості оптимізатора був використаний Адам із затуханням вагів. За метрику якості бралась ассурасу [3]. На рисунках 3 та 4 можна побачити графіки точності та функції втрат відповідно. Як бачимо, вони стабільно покращились протягом всього навчання.



Рисунок 3. Графік ассурасу

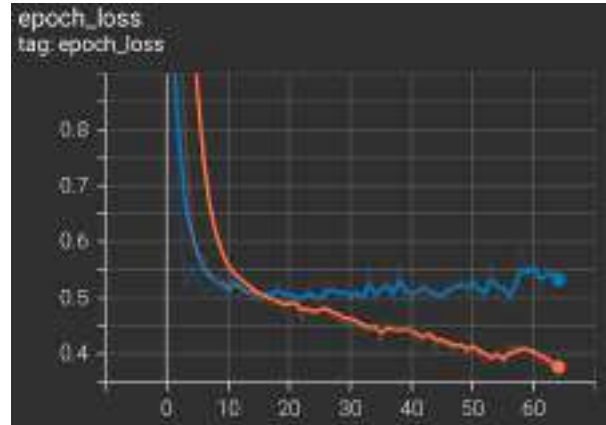


Рисунок 4. Графік функції втрат

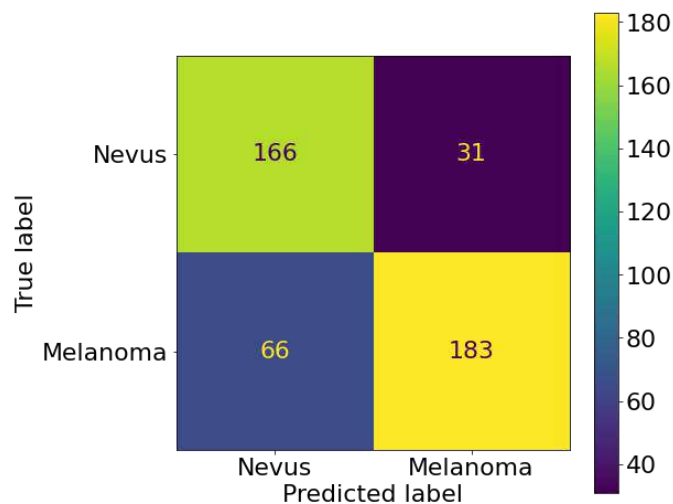


Рисунок 5. Матриця невідповідностей

### 4. ВИСНОВКИ

В результаті було отримано точність 78% на тестовому датасеті. Матриця невідповідностей (знову ж таки, на тестовому датасеті) показує, що більшість як позитивних так і негативних прикладів модель класифікувала правильно. Загалом з 446 тестових прикладів, маємо 66 помилок першого роду та 31 помилки другого роду.

Таким чином можна зробити висновок про придатність моделі зорового трансформера для поставленої задачі. Також, враховуючи тенденцію цієї моделі покращувати показники на більших датасетах краще ніж згорткові мережі [2], припущення про кращу роботу моделі зорового трансформера для класифікації раку шкіри ніж згорткової мережі є вірним.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ashish V., Ashish V., Noam S., Niki P., Jakob U., Llion J., Aidan N. G., Łukasz K.. Attention Is All You Need, 2017. 3–8 с.
2. Alexey D., Lucas B., Alexander K., Dirk W., Xiaohua Z., Thomas U., Mostafa D., Matthias M. An Image is worth 16X16 words: transformers for image recognition at scale, 2021, с 1–4.
3. Keras, ViT: документація. URL: [https://keras.io/examples/vision/image\\_classification\\_with\\_vision\\_transformer/](https://keras.io/examples/vision/image_classification_with_vision_transformer/)
4. HAM10000 : документація. URL: <https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T>
5. Фаундація раку шкіри: вебсайт. URL: <https://www.skincancer.org/skin-cancer-information/>

# АЛГОРИТМИ СОРТУВАННЯ З ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ

Подколзін Г.Б.<sup>1</sup>, Павлюк В.<sup>2</sup>

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

<sup>1</sup>podkolzin.gleb@lil.kpi.ua,

<sup>2</sup>pavliuk.vira@lil.kpi.ua

**Сортування є фундаментальною операцією в обчислювальній техніці. Однак, швидкість найсучасніших алгоритмів сортування в одному потоці досягли своїх меж. Тим часом глибоке навчання продемонструвало його потенціал для значного підвищення продуктивності задачі інтелектуального аналізу даних і машинного навчання. Метою роботи є дослідження доцільності та спроможності використання нейронних мереж у задачі сортування даних. Результатом дослідження є моделі sequence-to-sequence та pointer networks, що виконують сортування масивів чисел різної довжини. У роботі було використано теоретичні та емпіричні методи дослідження.**

**Ключові слова:** алгоритми сортування, нейронні мережі, глибоке навчання, sequence-to-sequence, pointer networks, сортування чисел.

## 1. ВСТУП

Сортування є основною операцією багатьох обчислювальних завдань. Воно впливає на найрізноманітніші параметри: від взаємодії користувачів із програмами до типів проблем, які ми можемо ефективно вирішити. Існує багато відомих послідовних алгоритмів сортування, таких як швидке сортування, сортування злиттям, пірамідалне сортування і інтросортування, які широко використовуються. Вони можуть задовольнити загальні вимоги до сортування в наукових дослідженнях та інженерних проектах. Однак ефективність алгоритмів послідовного знижується, коли вони стикаються з великомасштабними даними [1].

Основними цілями паралельного сортування є мінімізація часу сортування і навантаження процесорів, упорядковуючи дані елементи в бажаному порядку. Нейронна мережа - це відповідна архітектура в деяких паралельних алгоритмах, оскільки вона може обробляти дані одночасно.

## 2. МОДЕЛІ SEQUENCE-TO-SEQUENCE ТА POINTER NETWORKS

Однією з найбільш корисних властивостей, яку ми можемо мати, це здатність для моделі вивчати алгоритм, який узагальнює довші вхідні послідовності. Зокрема, фактична архітектура моделі легко потребуватиме здатності узагальнювати до більш довгих послідовностей, а алгоритм навчання, який її навчає, повинен буде відповідним чином упорядкувати, щоб модель не підганялася до коротших послідовностей, які вона бачить під час навчання. Ця проблема вимагає імплементацію функції, що відображає послідовність у послідовність. Очевидним шляхом вирішення цього є використання архітектури кодера-

декодера рекурентної нейронної мережі, яка використовується для задачі перекладу. З цієї причини для дослідження було обрано моделі Sequence-To-Sequence та Pointer Networks.

Моделі Sequence-To-Sequence (Seq2Seq) — це моделі глибокого навчання, які досягли значного успіху в таких завданнях, як машинний переклад, підсумовування тексту та створення підписів до зображень. Ця модель приймає послідовність елементів (слів, літер, часових рядів тощо) і виводить іншу послідовність елементів. Sequence-To-Sequence була запропонована дослідниками Іл'я Sutskever, Oriol Vinyals, Quoc V. Le для вирішення задачі перекладу тексту [2].

Модель складається з кодера і декодера (Рис 1.). Кодер фіксує контекст вхідної послідовності у вигляді прихованого вектора стану та надсилає його в декодер, який потім створює вихідну послідовність. Оскільки завдання базується на послідовності, і кодер, і декодер, як правило, використовують певну форму RNN, LSTM, GRU тощо. Прихований вектор стану може мати будь-який розмір, хоча в більшості випадків він приймається як ступінь 2 і велике число (256, 512, 1024), яке може певним чином представляти складність повної послідовності, а також домену.

RNN за дизайном приймають два входи, поточний приклад, який вони бачать, і представлення попереднього входу. Таким чином, вихід на кроці часу  $t$  залежить від поточного входу, а також входу на момент часу  $t-1$ . Саме тому вони працюють краще, коли ставляться до послідовних завдань. Послідовна інформація зберігається в прихованому стані мережі та використовується в наступному випадку.

Кодер, що складається з RNN, приймає послідовність як вхідні дані та генерує остаточне вбудовування в кінці послідовності. Потім це надсилається до декодера, який потім використовує його для передбачення послідовності, і після кожного наступного передбачення він використовує попередній прихований стан, щоб передбачити наступний екземпляр послідовності (Рис. 1) [3].

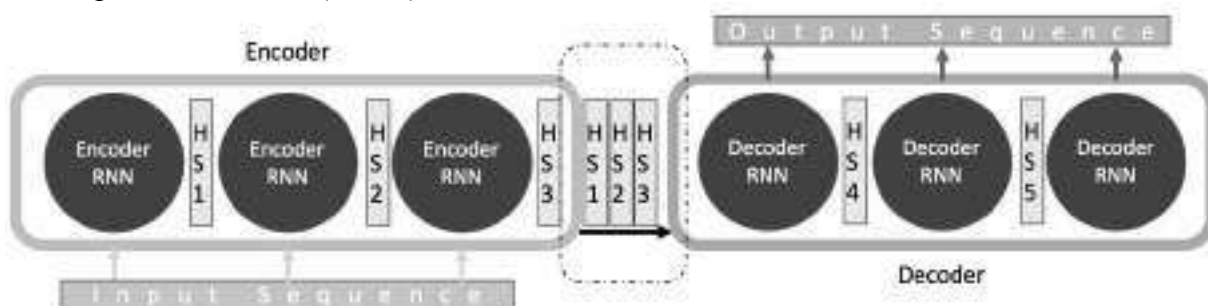


Рисунок 1. Модель кодера-декодера для моделювання Seq2Seq

Мережі вказівників (Pointer network, Ptr-Net) є різновидом моделі послідовності до послідовності з увагою. Замість перекладу однієї послідовності в іншу вони видають послідовність покажчиків на елементи вхідної серії. Основне використання цього — упорядкування елементів послідовності або набору змінної довжини [4].

Мережу вказівників запропонували Oriol Vinyals, Meire Fortunato, Navdeep Jaitly у 2017 році в своїй роботі [5]. Автори описали архітектуру мережі та запропонували її використання для вирішення таких складних геометричних задач, як: пошук плоских опуклих оболонок, обчислення триангуляцій Делоне та плоска задача комівояжера — лише з використанням навчальних даних.

Pointer networks (Рис. 2) ефективно створюють механізм уваги для змінної кількості токенів. Враховуючи послідовність токенів  $(t_0, t_1, \dots, t_{n-1})$ , мережа вказівників обслуговує вхідний словник кандидатів для наступного токена  $t_n$ . Кожен із цих кандидатів на вхід має

пов'язаний вектор вбудовування, створений кодувальником. Подібним чином, кожен з маркерів послідовності має власні вбудовані декодери. На відміну від інших моделей передбачення послідовності, мережі покажчиків побудовані таким чином, що кількість кандидатів на вхід може змінюватися під час висновку [6].

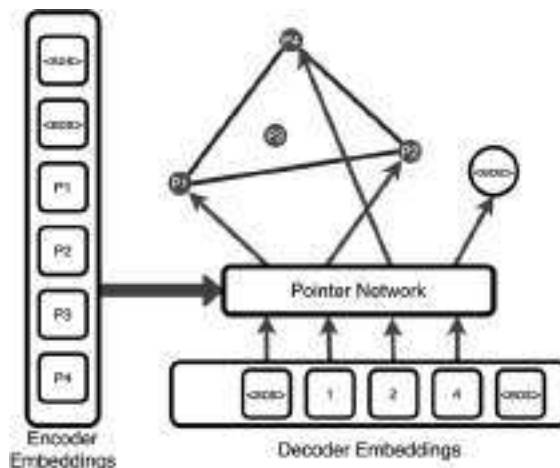


Рисунок 2. Архітектура Pointer Network

### 3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Задля досягнення мети дослідження було виконано програмну реалізацію кодеру-декодеру Sequence-To-Sequence та Pointer networks. Написано скрипт для генерації навчального та тестового наборів даних. Також написано скрипти для тренування та оцінки моделей та репортигу метрик. Проведено навчання моделей з різними налаштуваннями параметрів та зроблено оцінку для кожної моделі. Для реалізації було обрано мову програмування Python, яка найчастіше використовується для задач машинного навчання та містить чимало бібліотек для роботи з нейронними мережами. В нашій роботі використовується бібліотека pytorch для реалізації моделей, numpy для різних обчислень та matplotlib для графічного представлення певних результатів.

Згенеровані дані для навчання моделей містять одновимірні масиви чисел різної довжини. Мінімальна довжина масива береться за 2, а максимальна – 100. В рамках даного експерименту розглядаються цілі числа, які не більше 32. Тестові дані для оцінки моделей згенеровані аналогічним чином. Навчальні дані містять 10000 екземплярів, тестові – 5000.

Під час навчання на кожній ітерації обчислюється функція втрат негативний логарифм правдоподібності втрати (NLLLOSS), приклад значень якої для однієї епохи зображено на рисунку 3 та рисунку 4.

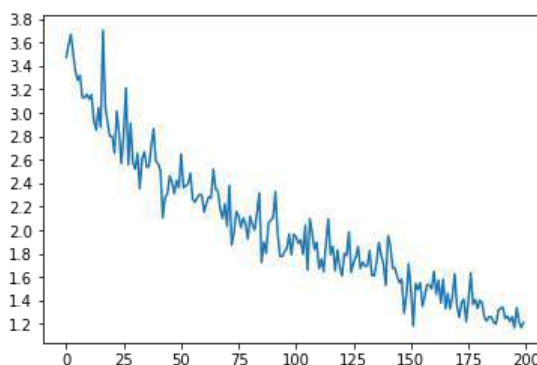


Рисунок 3. Значення NLLLOSS для першої епохи для моделі Pointer Network

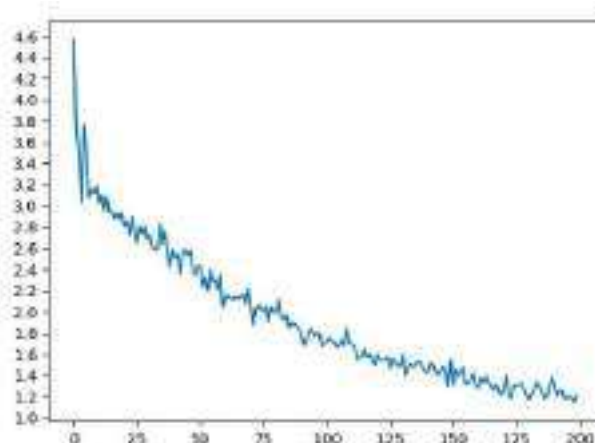


Рисунок 4. Значення NLLLOSS для першої епохи для моделі Seq2Seq

Результати тестування роботи моделей наведені у таблиці нижче (Табл. 1).

Таблиця 1. Робота сортування тестових даних моделей Sequence-To-Sequence та Pointer Networks

Оригінальний масив	Sequence-To-Sequence	Pointer Networks
[6, 23, 1]	[1, 6, 23]	[1, 6, 23]
[24, 7, 19, 28, 1]	[1, 7, 19, 24, 28]	[7, 19, 24, 28]
[22, 12, 12, 29, 5, 24, 1]	[5, 12, 22, 24, 29]	[1, 5, 12, 22, 24, 29]
[8, 10, 22, 19, 17, 6, 11, 1]	[6, 8, 10, 11, 17, 19, 22]	[1, 6, 8, 10, 11, 17, 19, 22]
[18, 11, 30, 15, 26, 11, 23, 4, 1]	[1, 4, 11, 11, 15, 18, 23, 26, 30]	[4, 11, 15, 18, 23, 26, 30]
[24, 2, 13, 21, 5, 6, 1]	[2, 5, 6, 13, 21, 24]	[1, 2, 5, 6, 13, 21, 24]
[25, 17, 5, 8, 18, 20, 18, 1]	[5, 8, 17, 18, 20, 25]	[1, 5, 8, 17, 18, 18, 20, 25]

Як видно з таблиці, модель періодично губить повторюванні значення (рядки 3 та 7 таблиці 1) та найменше значення 1. Тим не менш, немає порушення порядку сортування. Ми вирішили зробити модифікацію оригінального механізму уваги в моделі Pointer Networks, запропонованого авторами, з метою покращення результатів роботи. В таблиці 2 ми наводимо результати метрик, які дозволяють порівняти моделі. Метрика Permutation повертає долю, перестановок. Метрика Nondecreasing повертає долю елементів розташованих у порядку неспадання. Та, нарешті, метрика Accuracy повертає відсоток правильно відсортованих масивів.

Таблиця 2. Значення метрик Permutation, Nondecreasing та Accuracy

Метод	Sequence-To-Sequence	Pointer Networks	Modified Pointer Networks
Метрика			
Permutation	0.00	0.00	0.00
Nondecreasing	0.718	0.966	0.999
Accuracy	90.23%	92.81%	95.48%

Оскільки однією з причин проведення даного дослідження було бажання досягти мінімізації часу сортування даних великої розмірності, ми провели порівняння роботи моделі Pointer Network з послідовними алгоритмами сортування. Результати часу сортування для масиву довжиною 1000000 елементів наведені у таблиці нижче (Табл. 3).

Таблиця 3. Порівняння часу сортування

Алгоритм	Modified Pointer Network	Bubble Sort	Merge Sort	Heap sort	TimSort
Час	25 s	1 h 56 m	2.5 s	4 s	3.2 s

#### 4. ВИСНОВКИ

Сортування є фундаментальною операцією в обчислювальній техніці. Проблема сортування не є новою та має багато різних рішень за допомогою класичних алгоритмів сортування, які базуються на послідовному попарному порівнянні та перестановці елементів. Проте класичні алгоритми не є оптимальними у випадку великої розмірності даних. Це може стати серйозною проблемою при розробці складних рішень.

Тим не менш нейронні мережі, завдяки своїй архітектурі, здатні виконувати паралельні обчислення, що дозволяє оптимізувати витрати часу та навантаження на процесор під час сортування. Оскільки сортування чисел по суті відображає послідовність кластерів у послідовність такої самої розмірності, ми вибрали для дослідження моделі Sequence-To-Sequence та Pointer Networks. Перевагою моделей є здатність сприймати на вхід послідовності нефіксованої розмірності, саме тому їх використовують для проблеми перекладу.

Для дослідження було використано в якості навчальних даних списки цілочисельних значень різної розмірності. В результаті оцінки роботи моделей було виявлено, що вони виконують сортування чисел, проте періодично допускають похибку, зменшуючи довжину послідовності. Запропоновані метрики показали, що модель Pointer Networks краще впоралась з цією задачею.

#### ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Zhao, Y., Dongli, H., Huang, D., Liu, Y., Yang, Z., Mao, L., Liu, C., & Zhou, F. (2020, July 27). *Sorting Data via a Look-Up-Table Neural Network and Self-Regulating Index*. <https://doi.org/10.1155/2020/4793545>
2. Sutskever, I., Vinyals, O., & V. Le, Q. (n.d.). Sequence to Sequence Learning with Neural Networks.
3. Dugar, P. (2019, July 13). Attention — Seq2Seq Models. Towardsdatascience. <https://towardsdatascience.com/day-1-2-attention-seq2seq-models-65df3f49e263>
4. Singh, A. (2020, May 24). Pointer Networks for Deep Learning. Towardsdatascience. <https://towardsdatascience.com/understanding-pointer-networks-81fbbc1ddbc8>
5. Vinyals, O., Fortunato, M., & Jaitly, N. (2017, January 2). *Pointer Networks*. arxiv.org. <https://arxiv.org/pdf/1506.03134v2.pdf>
6. McGough, M. (2021, June 18). Pointer Networks with Transformers. Towardsdatascience. <https://towardsdatascience.com/understanding-pointer-networks-81fbbc1ddbc8>

# ВИКОРИСТАННЯ НЕЙРОННИХ МЕРЕЖ ТА GRADIENT BOOSTING ДЛЯ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ ВИТРАТ

Скоробогатов С.Ю.<sup>1</sup>, Зайченко О.Ю.<sup>2</sup>

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

<sup>1</sup> iskor29@gmail.com [0000-0002-6532-0836],

<sup>2</sup> zaichenko.helen@iill.kpi.ua [0000-0002-4630-5155]

**Будь-яке комерційне підприємство перш ніж розпочати виробництво, аналізує ринкову кон'юнктуру. Знаючи обсяги попиту на продукцію, підприємство може планувати обсяги доходів та, відповідно, обсяг витрат. На сьогодні існує великий масив даних, які можна обробити, використовуючи методи математичного аналізу для підвищення точності прогнозування фінансових результатів підприємства на основі даних про планований обсяг доходів та витрат. Водночас, важливим є вирішення проблеми оптимізації логістичних витрат, тобто визначення оптимального варіанту доставки необхідного обсягу продукції кожному замовнику. Для вирішення даної проблеми пропонуємо використовувати нейронні мережі а також модель Gradient Boosting при прогнозуванні майбутнього попиту на продукцію, обсяги його реалізації та плануванні логістичних витрат. Вибір оптимального рішення пропонується здійснювати на основі аналізу метрик якості.**

**Ключові слова:** максимізація доходу, мінімізація витрат, прогнозування, нейронні мережі, регресійні моделі, оцінка попиту

## 1. ВСТУП

Завдяки достовірності оцінки попиту на продукцію та достатньому обсягу виготовлення продукції для задоволення даного попиту, власники та керівники підприємств можуть отримати інформацію для планування (прогнозування) відповідних доходів та витрат. Недостовірна оцінка попиту позначиться на неточності прогнозування обсягів виробництва. У свою чергу, це призведе до неточності прогнозування відповідних витрат підприємства, включаючи і логістичні. Тому в умовах високого ступеня невизначеності навколишнього середовища вважаємо необхідним підвищити ступінь достовірності прогнозів. Задля цього пропонуємо використовувати методи системного аналізу, включаючи нейронні мережі та регресійні моделі на основі Gradient Boosting.

Завдяки прогнозуванню попиту, компанії можуть виробляти достатню кількість продукції для того, щоб його задовольнити. Тому на даний момент використання аналізу даних і прогнозування є дуже поширене у великих компаніях. За допомогою прогнозування на основі нейронних мереж та регресійних моделей можна досягти високої точності оцінки попиту, що зменшить витрати компанії через похибку у аналізі даних та прийнятих на основі цього рішень. Саме шляхом прогнозування попиту на продукцію, компанії можуть оцінити свій подальший розвиток і зробити певні вкладення або обмеження на виготовлення продукції. Після оцінки попиту можна зробити замовлення на продукцію заздалегідь, що дозволить зменшити витрати на продукцію. Також можна зробити попередній план перевезень продукції, що збільшить швидкість її доставки та зменшить витрати компанії.

Коли підприємства країни отримують прибутки, це позначається на загальному стані економічного розвитку країни. Таким чином, оптимізуючи витрати підприємства, можемо отримати максимум прибутків та підвищити добробут населення країни.

## 2. МЕТОДИ МАКСИМІЗАЦІЇ ТОЧНОСТІ ОЦІНКИ МАЙБУТНЬОГО ПОПИТУ

На даний момент використовуються такі методи для оцінки попиту: опитування споживачів, лабораторні експерименти, рандомізоване дослідження та лінійна регресія. Опитування споживачів можуть бути виконані за допомогою прямих інтерв'ю або анкет з потенційними клієнтами, щоб отримати інформацію. Цей процес включає швидкі короткі інтерв'ю/опитування або поглиблені. Проте не дуже багато людей згодні відповісти на поставлені запитання, через що оцінювання попиту на основі опитів досить часто відхиляється від реальних результатів. Лабораторні експерименти мають на меті перевірити, як споживачі реагують на зміни змінних у функції попиту в гіпотетичній ситуації. Ці експерименти можна використовувати для перевірки поведінки споживачів різними способами, але вони не обов'язково мають відношення до попиту на певний продукт. Рандомізоване дослідження можна проводити, щоб усунути певні упередження, які можуть бути присутніми в іншому випадку, такі упередження можуть включати упередження відбору та упередження розподілу. Рандомізовані випробування проводяться шляхом випадкового розподілу досліджуваних на дві або більше груп. Регресія вимірює ступінь, в якому зміна в деякій залежній змінній спричинена однією або кількома незалежними змінними. Він адаптує лінійне рівняння до спостережених даних і вимірює, наскільки добре рівняння відповідає, або наскільки результати залежної змінної можна пояснити незалежною. Лінійна регресія є потенційним засобом для оцінки того, як попит (залежна змінна) змінюється з будь-якою незалежною змінною [1]. Проте лінійна регресія не є настільки точною, на відміну від повноцінної нейронної мережі, в якій є приховані шари (Рис. 1).

Все більше уваги приділяється оптимізації процесів, в основному, у вигляді зниження витрат та підвищення доходів. Зниження витрат можна досягти модернізацією обладнання, але даний підхід тягне за собою безліч витрат на проектування, купівлю, реконструкцію, а також супроводжується недоотриманим прибутком під час простою об'єкта, що реконструюється. Визначення обсягів доходів можна досягти за рахунок прогнозування. Використання нейронної мережі дозволить визначити обсяг попиту при мінімальній похибці. Зокрема, обсяг попиту на продукцію можна оцінити за допомогою нейронної мережі (рис. 1).

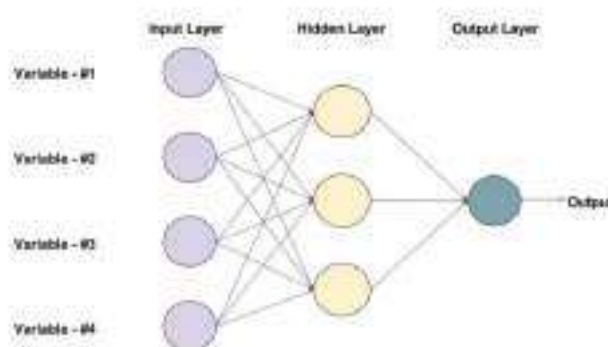


Рисунок 1. Зображення нейронної мережі [2]

Нейрон – базовий елемент нейронної мережі, одинична проста «обчислювальна комірка», здатний сприймати, перетворювати і поширювати сигнали. Тому об'єднання великої кількості нейронів в одну мережу дозволяє вирішувати досить складні завдання.

Навчання нейронної мережі насамперед полягає у зміні «сили» зв'язків між нейронами. Нейронні мережі масштабуються, вони здатні вирішувати досить складні завдання. Аналіз за допомогою нейронної мережі можна уявити як по черговий аналіз даних від менш складних до більш складних із проходженням шарів нейронної мережі [3].

Другий варіант підвищення точності прогнозування – використання регресійних моделей. Для таких цілей зазвичай використовується модель із Gradient Boosting, яка є ансамбль-моделлю. Моделі машинного навчання можна адаптувати до даних окремо або об'єднати в ансамбль. Ансамбль — це поєднання простих окремих моделей, які разом створюють більш потужну нову модель. Прискорення машинного навчання — це метод створення ансамблю. Він починається з підгонки початкової моделі до даних. Потім будується друга модель, яка фокусується на точному прогнозуванні випадків, коли перша модель працює погано. Очікується, що поєднання цих двох моделей буде кращим, ніж кожна з них окремо. Потім даний процес повторюється багато разів. Кожна наступна модель намагається виправити недоліки ансамблю всіх попередніх моделей. Ключова ідея полягає в тому, щоб встановити цільові результати для наступної моделі, щоб мінімізувати помилку (рис. 2).

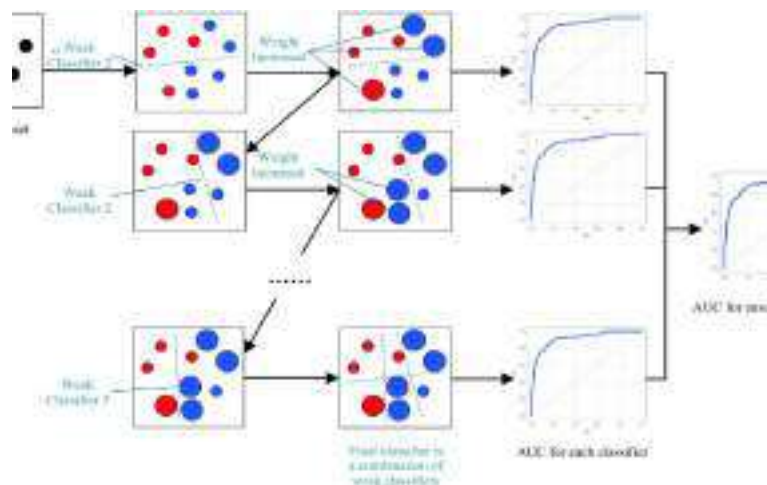


Рисунок 2. Зображення алгоритму оптимізації моделі Gradient Boosting [4]

Цільовий результат для кожного випадку в даних залежить від того, наскільки зміна прогнозу цього випадку впливає на загальну помилку прогнозу:

- Якщо невелика зміна прогнозу для випадку спричиняє значне зниження помилки, тоді наступним цільовим результатом випадку буде високе значення. Прогнози нової моделі, близькі до цілей, зменшать помилку.
- Якщо невелика зміна прогнозу для випадку не спричиняє зміни помилки, тоді наступний цільовий результат випадку дорівнює нулю. Зміна цього прогнозу не зменшує помилку.

Вважаємо доцільним для оцінки прогнозованого обсягу попиту на продукцію, який визначає обсяги доходів підприємства, використання двох методів – нейронні мережі та Gradient Boosting. В подальшому приймається для прийняття рішень про обсяги виробництва та варіанти доставки продукції до споживачів той варіант прогнозової моделі, який дає меншу похибку.

Використавши попередні вхідні дані критеріїв залежності попиту і вихідні змінні попиту, можна навчити моделі нейронної мережі і Gradient Boosting, після чого обрати найбільш точну модель. Є різні критерії визначення точності моделі, але найбільш популярними для оцінки

неперервних значень  $\epsilon$ : Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Accuracy.

Після оцінки майбутнього попиту за одним із варіантів, можна розв'язати задачу знаходження найбільш оптимального перевезення товару від складу товару до місця збуту товару. Для цього можемо використати метод потенціалів або угорський метод для вирішення отриманої транспортної задачі [5].

### 3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

#### 3.1 Визначення точності прогнозування на основі нейронних мереж

За допомогою фреймворку tensorflow було побудовано нейронну мережу, в якій було 3 приховані шари. На першому шарі знаходилось 40 нейронів, на другому прихованому шарі знаходилось 50 нейронів і на останньому 1 нейрон. Модель навчено на 506 спостереженнях. Для навчання було обрано 80% даних для тренування і 20% даних для тестування навченої моделі (рис. 3).

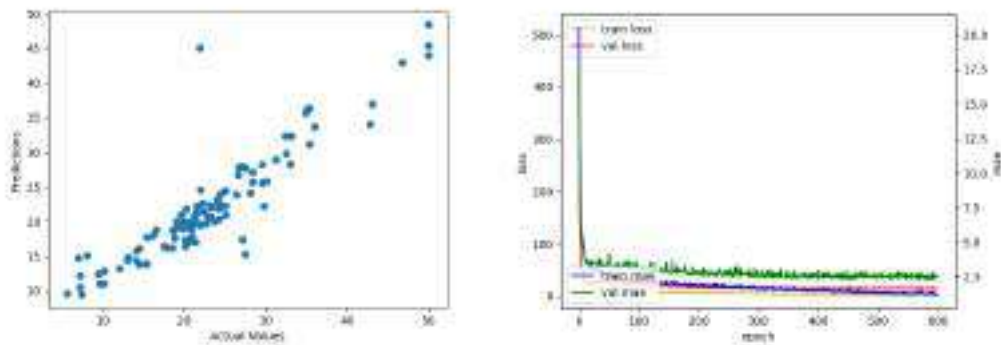


Рисунок 3. Навчання та тестування моделі із використанням нейронної мережі

За графіком помилок можемо побачити, що модель дуже швидко навчалась на початковому наборі даних, але із збільшенням кількості проходів модель зменшила швидкість зпаження функції помилки. Перенавчання на наборі для тренування не відбулось, адже була використана регуляризація у вигляді Dropout, який дозволяє відключати нейрони на обраних шарах, щоб модель не орієнтувалась на конкретні нейрони. За графіком порівняння точних значень і оцінених за допомогою моделі можна зазначити, що модель дуже точно описала дані. Особливо це помітно, бо в ідеальному випадку повинна бути пряма по діагоналі. Є декілька викидів, але, зважаючи на досить маленький датасет, дані оцінки є досить точними також. За допомогою нейронної мережі можемо знайти майбутній попит на товар.

#### 3.2 Визначення точності прогнозування на основі Gradient Boosting Regressor

За допомогою бібліотеки Scikit-learn та зробленого в ній ансамблю моделей Gradient Boosting Regressor було зроблено навчання і прогнозування на 506 спостереженнях. Використано 80% даних для тренування і 20% даних для тестування навченої моделі (рис. 4).

За допомогою пошуку гіперпараметрів із GridSearchCV у бібліотеці Scikit-learn обрано використано такі параметри: learning\_rate=0.05, loss='ls', max\_depth=5, max\_features='log2', min\_samples\_leaf=5, min\_samples\_split=12, n\_estimators=200, subsample=0.8. Перенавчання у натренованої моделі відсутнє, як можна побачити за графіком точних значень і передбачених значень. Оскільки точки знаходяться майже по діагоналі, можна вважати, що ця модель виконала передбачення навіть краще, ніж просто нейронна мережа. Проте так може відбуватись не завжди, адже у нейронної мережі значено більше можливості в плані

налаштування шарів та функції, яку дані шари можуть виконувати. За допомогою Gradient Boosting Regressor можемо знайти майбутній попит на товар.



Рисунок 4. Навчання та тестування моделі із використанням Gradient Boosting Regressor

## 4. ВИСНОВКИ

На сьогодні одним із найбільш важливих критеріїв ефективності витрат компанії є оцінка попиту і подальша закупка і перевіз матеріалів з місця виготовлення на місця продажі або подальшого виготовлення іншої продукції. Правильний підхід до вирішення проблеми оцінки попиту надає можливості для прийняття правильних рішень у розподілі продукції, отриманні максимального прибутку та досягненні максимальних результатів у максимізації доходів у майбутньому.

Завдяки високоточному прогнозуванню попиту за допомогою нейронних мереж та регресійних моделей на прикладі Gradient Boosting Regressor, можна отримати дуже високу точність попиту в наступні рази на основі попередньо зібраних даних. Це дозволить компанії отримати наглядну картину, щоб розуміти фінансову ситуацію і потім зробити необхідні дії для досягнення максимального заробітку та стабільності.

За результатами метрик якості та візуального аналізу прогнозованих значень побудованими моделями, можна отримати попит із точністю більше ніж 95%. Це дозволяє замовити продукцію у виробників заздалегідь та із відповідністю до обмежень потужностей виробника розрахувати оптимальний план для побудованої транспортної задачі.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Wilkinson, N. Управлінська економіка: підхід до вирішення проблем. Кембридж: Cambridge University Press, стор.125.
2. What is a Neural Network? [Електронний ресурс]. – Режим доступу: <https://www.projectpro.io/article/neural-network-projects/440> (дата доступу 19.11.2022)
3. Zell, Andreas. Simulation Neuronaler Netze [Simulation of Neural Networks] (1st ed.). Addison-Wesley. p. 73. ISBN 3-89319-554-8.
4. Gradient Boosting – What You Need to Know [Електронний ресурс]. – Режим доступу: <https://datascience.eu/machine-learning/gradient-boosting-what-you-need-to-know/> (дата доступу 19.11.2022)
5. Ю. П. Зайченко. Дослідження операцій. Підручник. Сьоме видання, перероблене та доповнене. – К.:Видавничий Дім «Слово», 2006. – 816 с.

# МЕТОДИ БАГАТОФАКТОРНОЇ АВТЕНТИФІКАЦІЇ ДО ВЕБ ДОДАТКІВ ЗА ДОПОМОГОЮ ШТУЧНОГО ІНТЕЛЕКТУ ТА ТЕХНОЛОГІЇ БЛОКЧЕЙН

Славінський В.О.<sup>1</sup>, Гуськова В.Г.<sup>2</sup>

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

<sup>1</sup>sevoslav@gmail.com

У інтернеті практично всі веб-сайти вимагають підтвердження особи, однак поточні методи авторизації недосконалі, оскільки вони покладаються на застарілий метод ідентифікації через e-mail і пароль, який легко зламати або ненавмисно передати зловмисникам. У роботі розглянуто альтернативні методи авторизації, такі як авторизація через біометрію, а також новітній підхід - автентифікація через блокчейн. Результатом дослідження є порівняння існуючих методів авторизації та автентифікації, а також перспективи автентифікації за допомогою блокчейну та біометрії у вебі 3.0.

**Ключові слова:** автентифікація, мультифакторна автентифікація, блокчейн, web3.0, розпізнавання облич.

## 1. ВСТУП

Об'єктом дослідження є мультифакторна авторизація та аутентифікація в мережі Web3.0.

Предметами дослідження визначено нейронні мережі для розпізнавання облич та блокчейн для роботи з розподіленими ідентифікаторами. Задача роботи полягає у створенні прикладу використання мультифакторної автентифікації на основі нейронних мереж і блокчейну в мережі Web3.0. Отримані результати надалі можна буде розвинути в самостійну бібліотеку, яку зможуть використовувати незалежні розробники для своїх додатків в розподіленому інтернеті. Дана робота допоможе зробити систему безпечнішою та зручнішою для користувачів, а також дасть змогу перестати зберігати свої дані у великих корпорацій, таких як Google або Facebook.

## 2. МЕТОДИ ІДЕНТИФІКАЦІЇ ОСОБИ

Для ідентифікації особи може використовуватися документ, що посвідчує особу, наприклад, посвідчення особи (також відоме як ІС, ID-картка, картка громадянина), або паспортна картка (якщо вона має невеликий формат, як звичайна кредитна картка). Деякі країни також видають офіційні документи, що посвідчують особу, такі як національні посвідчення особи, які можуть бути обов'язковими або необов'язковими, в той час як інші можуть покладатися на регіональні посвідчення особи або неофіційні документи для підтвердження особи.

Існують і деякі інші прийнятні форми посвідчення особи:

**Щось, що особа знає (SYK):** пароль, PIN-код, дівоче прізвище матері або кодова комбінація замка. Ідентифікація особи за допомогою чогось, що вона вже знає, є, ймовірно, найпростішим варіантом, але одним з найменш безпечних.

**Щось, що є у особи (SYH):** ключ, свайп-карта, картка доступу або бейдж - все це приклади предметів, якими може володіти особа. Цей метод зазвичай використовується для отримання доступу до таких об'єктів, як банки та офіси, але він також може бути використаний для отримання доступу до конфіденційних місць або перевірки системних облікових даних. Це також простий варіант, але ці предмети легко вкрати.

**Щось, чим є людина (SYA):** біометричні дані особи є унікальними і не можуть бути втрачені або викрадені. Використання біометричних даних для ідентифікації особи є найбільш точним і безпечним варіантом.

Кіберзлочинці постійно вдосконалюють свої системні атаки. Як наслідок, команди безпеки мають справу з безліччю постійно мінливих проблем з автентифікацією. Ось чому компанії починають розгортати більш складні плани, які включають автентифікацію. Деякі з найпоширеніших методів аутентифікації, що використовуються для захисту сучасних систем, включають в себе:

**Face Recognition.** Аутентифікація за обличчям - це технологія, яка дозволяє людям отримувати доступ до онлайн-сервісів, фізичних параметрів та інших ресурсів, використовуючи зображення свого обличчя.

Аутентифікація за обличчям, яка також називається розпізнаванням обличчя, спирається на вбудовану технологію зондування мобільних та інших пристроїв. На відміну від інших рішень ідентифікації, таких як паролі, верифікація за допомогою електронної пошти, селфі або зображень, або ідентифікація за відбитками пальців, біометричне розпізнавання обличчя використовує унікальні математичні та динамічні шаблони, які працюють як сканер обличчя, що робить цю систему однією з найбезпечніших та найефективніших.

**Fingerprint Recognition.** Автентифікація за відбитками пальців - це акт перевірки особи на основі одного або декількох відбитків пальців. Ця концепція використовується протягом десятиліть у різних сферах, включаючи цифрову ідентифікацію, кримінальне правосуддя, фінансові послуги та охорону кордонів.

**FIDO 2.** Fast Identity Online (FIDO) - це відкритий стандарт для безпарольної автентифікації. FIDO дозволяє користувачам і організаціям використовувати цей стандарт для входу на свої ресурси без імені користувача або пароля за допомогою зовнішнього ключа безпеки або ключа платформи, вбудованого в пристрій.

Ці ключі безпеки FIDO2, як правило, є USB-пристроями, але можуть також використовувати Bluetooth або NFC. Завдяки апаратному пристрою, який обробляє автентифікацію, безпека облікового запису підвищується, оскільки немає пароля, який можна було б розкрити або вгадати.

**SMS OTP.** Це зручний метод, який не вимагає від користувачів встановлення будь-яких додатків. Замість цього, для аутентифікації на зареєстрований телефон користувача надсилається одноразовий пароль за допомогою SMS, який використовується для аутентифікації.

Загальні проблеми:

- Проблеми з користувацьким інтерфейсом - OTP часто мають обмеження за часом, а обмежений прийом мобільних операторів може спричинити проблеми для користувачів у віддалених районах.
- Вразливість до шкідливого програмного забезпечення, атак SS7 та підміни SIM-карт.

**OATH OTP (Soft Tokens).** Це програмне забезпечення, яке може бути вбудоване в мобільні додатки і використовує криптографічні операції для автентифікації користувача та пристрою. Ці рішення зазвичай забезпечують більш плавний користувацький інтерфейс; немає необхідності перемикатися між додатками або покладатися на апаратний пристрій. З

точки зору безпеки, вони мають значні переваги, оскільки SDK з м'якими токенами підтримують просунуту криптографію, наприклад, цифрові підписи.

Загальні проблеми:

- Поганий користувацький досвід (UX) - користувачі повинні постійно перемикатися між додатками для підтвердження особи/транзакції; користувач втрачає доступ з кожною зміною/втратою/оновленням свого смартфона; немає безпечних варіантів резервного копіювання

**Blockchain.** Блокчейн-аутентифікація відноситься до системи, розробленої для підвищення безпеки користувачів та перевірки особи користувача і дозволяє користувачам підключатися до ресурсів, заснованих на технологіях цифрової валюти, транзакцій, криптовалют тощо.

Вона використовує технологію розподіленого реєстру блокчейн та методи аутентифікації для підвищення конфіденційності та безпеки систем аутентифікації. Вся мережа, заснована на блокчейні, здатна мати власну цілісність даних.

Персональна інформація, яка використовується для перевірки особи користувача, зберігається в хеші блоку, наприклад, ім'я користувача або пароль. Це допоможе досягти суверенної ідентичності.

**Passwords.** Паролі є найпоширенішим методом автентифікації. Паролі можуть бути у вигляді рядка букв, цифр або спеціальних символів. Для захисту необхідно створювати надійні паролі, які включають комбінацію всіх можливих варіантів.

Однак паролі схильні до фішингових атак, що послаблює їх ефективність. Середньостатистична людина має близько 25 різних облікових записів в Інтернеті, але тільки 54% користувачів використовують різні паролі для своїх облікових записів.

Справа в тому, що існує дуже багато паролів, які потрібно пам'ятати. Як наслідок, багато людей обирають зручність, а не безпеку. Більшість людей використовують прості паролі замість того, щоб створювати надійні паролі, тому що їх легше запам'ятати.

**Secret Questions.** Безпечні питання є поширеним методом автентифікації особи, з яким ви, ймовірно, стикалися раніше. Створюючи обліковий запис або реєструючись на послугу в Інтернеті, користувачі конфіденційно передають відповіді на секретні запитання провайдеру.

Як правило, ці питання безпеки та відповіді на них використовуються для самообслуговування відновлення пароля - введення правильної відповіді перевіряє користувача та дозволяє йому скинути пароль - хоча ви також можете використовувати питання безпеки як додатковий фактор автентифікації для входу в систему.

**Personal Info.** Персональна аутентифікаційна інформація - ПІН-код або будь-який інший пароль чи інформація, яку користувачі створюють або приймають для використання з метою аутентифікації своєї особи у додаток. Інші приклади Персональної автентифікаційної інформації можуть включати ім'я, дату народження, поштовий індекс, місце проживання, які можуть використовуватися або вимагатися для здійснення Інтернет-транзакцій або інших операцій.

**Email OTP.** Метод Email OTP дозволяє авторизуватися за допомогою одноразового пароля (OTP), який надсилається на зареєстровану адресу електронної пошти. При спробі авторизації на будь-якому сервісі сервер надсилає OTP на зареєстровану електронну адресу користувача.

Щоб використовувати функцію Email OTP, необхідно спочатку зареєструвати альтернативний ідентифікатор електронної пошти. Реєстрація альтернативного ідентифікатора електронної пошти необхідна для того, щоб OTP могло бути надіслано на цей ідентифікатор, оскільки ви не зможете отримати доступ до основного ідентифікатора

електронної пошти, якщо обліковий запис буде заблоковано або якщо ви забудете пароль до облікового запису.

### 3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Кожен метод автентифікації користувача, як правило, може бути оцінений за допомогою наступних трьох ключових параметрів:

**Зручність:** Наскільки природно і без проблем для кінцевого користувача використовувати цю автентифікацію?

**Безпека:** наскільки складно зловмиснику обдурити автентифікацію?

**Розгортання:** Наскільки легко розгортати для всіх користувачів на різних платформах, пристроях, географічних регіонах тощо? Важливим фактором є те, чи є сценарій використання B2C або B2E, тобто чи є кінцевий користувач споживачем або працівником.

Тож для дослідження візьмемо описані вище методи автентифікації і проаналізуємо їх за обраними факторами та наведемо табличний огляд деяких з найпопулярніших методів автентифікації з оцінками за кожним з трьох параметрів. Зверніть увагу, що ці бали є простими відносними показниками, які дають загальне уявлення про відносні сильні та слабкі сторони кожного методу.

Таблиця 1. Порівняння існуючих методів автентифікації

Методи автентифікації	Вид	Розгортання			Коментарі
		Зручність	Безпека	Розгортання	
Face Recognition	SYA	9	7	2	Загальне розпізнавання облич - непослідовна безпека, сильно залежить від апаратного та програмного забезпечення пристрою. Наприклад, безпека розпізнавання облич на Android значною мірою залежить від марки пристрою.
Fingerprint Recognition	SYA	7	7	4	Загалом безпечніший і більш розповсюджений, ніж розпізнавання обличчя, і з кожним роком стає все більш розповсюдженим. Практично всі смартфони мають досить пристойний сканер відбитків пальців.
FIDO 2	SYH	6	8	3	Використовує біометричні автентифікатори і намагається зробити їх сумісними на різних пристроях, працюючи з усіма гравцями галузі, але має деякий час, щоб дозріти в плані підтримки.
SMS OTP	SYH	3	5	8	Не дуже безпечний, оскільки схильний до атак на підміну SIM-карт, але активно використовується як другий фактор у сценаріях B2C, оскільки кожен користувач має смартфон.
OATH OTP (Soft Tokens)	SYH	3	8	5	Досить безпечний, але потребує програми-автентифікатора, і користувач повинен знати, як її налаштувати. Це може бути корисно, якщо кінцевий користувач технічно підкований, як правило, у сценаріях B2E.

Blockchain	SYH	8	10	5	Дуже безпечний спосіб аутентифікації, і в той же час досить зручний оскільки не вимагає носіння спеціальних пристроїв. Досить мати доступ до гаманця з будь-якого пристрою будь то персональний комп'ютер або телефон. Також легко вбудовується в додатки оскільки спочатку був розроблений в середовищі web і має хорошу інтеграцію з будь-якими веб додатками.
Passwords	SYK	4	3	6	Найбільш розповсюджений через історію та можливість розгортання має добре відомі проблеми з безпекою. Всі фахівці з безпеки хочуть, щоб вона зникла якомога швидше.
Secret Questions	SYK	5	2	7	Можливо, це гірше, ніж паролі, особливо в епоху соціальних мереж, оскільки багато інформації про користувачів є загальнодоступною.
Personal Info	SYK	6	0	8	Запитувати особисту інформацію, таку як дата народження, є жахливим методом в епоху соціальних мереж. Він навіть не повинен бути вказаний як метод аутентифікації, але я бачив, що деякі сервіси використовують його.
Email OTP	SYK	4	5	7	Гідний метод для сценаріїв B2C, таких як SMS, і не залежить від іншого пристрою, але в значній мірі залежить від ефективності та безпеки електронної пошти користувача.

Якщо зобразити ці оцінки на бульбашковій діаграмі, де розмір бульбашки відображає розгортання, безпеку по осі Y та зручність використання по осі X, то вона виглядатиме як зображено на рисунку 1:

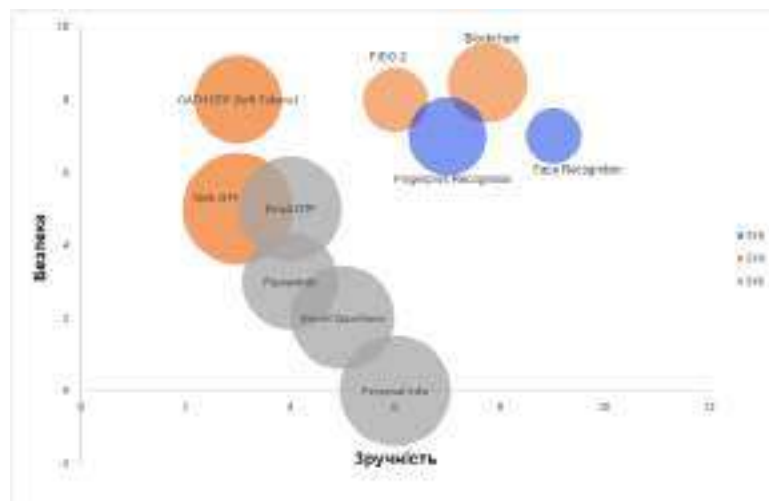


Рисунок 1. Порівняння методів автентифікації користувачів за 3 параметрами - безпека, зручність та розгортання. Розмір бульбашки вказує на можливість розгортання цього методу автентифікації

## 4. ВИСНОВКИ

Дивлячись на отримані результати можна легко побачити, що в більшості ситуацій єдиний метод аутентифікації користувача не може бути панацеєю. Саме для цього на допомогу приходять мультифакторна автентифікація, вона може допомогти убезпечити вхід користувача, використовуючи другий фактор, бажано іншої категорії. Як ми бачимо з результатів, фактор знання найслабший у безпеці та зручності, оскільки вимагають постійного введення даних і їх запам'ятовування. Якщо ж говорити про фактор того, хто ти є, то найпростіше використовувати розпізнавання облич, оскільки воно вимагає найменше дій від користувача і є більш унікальним, оскільки відбиток пальця простіше підробити і складніше розпізнати. І нарешті останній фактор - це те що людина має. З них найкращі показники у блокчейна, оскільки він є найнадійнішим, оскільки блокчейн гарантує, що дані користувача не будуть змінені, оскільки вони вже записані в блокчейн, а також немає необхідності мати спеціальний пристрій, достатньо мати доступ до свого криптогаманця.

### ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. What is Multi-Factor Authentication (MFA) and How Does it Work? URL: <https://www.onelogin.com/learn/what-is-mfa>
2. Two-Factor Authentication (2FA) URL: <https://duo.com/product/multi-factor-authentication-mfa/two-factor-authentication-2fa>
3. Identity and Web3 URL: <https://auth0.com/blog/identity-and-web3/>
4. Cryptocurrency Explained With Pros and Cons for Investment URL: <https://www.investopedia.com/terms/c/cryptocurrency.asp>
5. Advantages and disadvantages of biometrics URL: <https://www.miteksystems.com/blog/advantages-and-disadvantages-of-biometrics>
6. The 5 most common multi-factor authentication (MFA) methods URL: <https://securitybrief.asia/story/the-5-most-common-multi-factor-authentication-mfa-methods>
7. Comparison of User Authentication Methods on Three Parameters [URL: <https://www.cyberark.com/resources/blog/comparison-of-user-authentication-methods-on-three-parameters>

# АГЕНТНА АРХІТЕКТУРА МЕХАНІЗМІВ КОНСЕНСУСУ В БЛОКЧЕЙНІ

Тимошук О.Л.<sup>1</sup>, Лисов Б.С.<sup>2</sup>

<sup>1,2</sup> Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

<sup>1</sup>o.tymoshchuk@kpi.ua [0000-0003-1863-3095], <sup>2</sup>lysov.bohdan@lll.kpi.ua

**Блокчейн технологія створила твердий фундамент для децентралізації проведення фінансових операцій без єдиного контролюючого органу за правилами мережі. «The Merge» - операція переходу з proof-of-work на proof-of-stake механізм консенсусу - створює нове підґрунтя для дослідження ефективності використання ресурсів в технологіях. Мета дослідження полягає у розробці агентної архітектури для аналізу блокчейну Ethereum, роботи його транзакційного механізму в умовах обраного механізму консенсусу та порівняння цих механізмів у вигляді зібраних метрик та як вони впливають на кількість витраченої електроенергії, а отже і впливу на навколишнє середовище. Новизна роботи полягає у підборі специфічних параметрів для забезпечення оптимального використання децентралізованої мережі відповідно до використання електроенергії за допомогою агентної архітектури.**

## 1. ВСТУП

З появою таких P2P платіжних систем, як Bitcoin і Ethereum, збільшується кількість людей, які зацікавлені в покупці та продажі цифрових активів. Інфраструктура не стоїть на місці: з'являються різні аналоги з додатковим функціоналом, але всі вони підпорядковуються суворим стандартам.

Для валідації стану мережі існують механізми консенсусу, під якими мається на увазі досягнення загальної згоди її членами. Що стосується блокчейну Ethereum, процес є формалізованим, і досягнення консенсусу означає, що принаймні 66% вузлів у мережі погоджуються щодо глобального стану мережі.

До 15 вересня 2022 року Ethereum використовував механізм proof-of-work, але зрештою перейшов на proof-of-stake концепт. Операція переходу усунула потребу в енергоємному майнінгу, що натомість дозволило захистити мережу за допомогою накопиченого ETH. Це важливо, оскільки знижується кількість використаного ресурсу для проведення транзакції, що позитивно впливає на довколишнє середовище в порівнянні з минулими роками.

За допомогою штучного інтелекту, а саме інтелектуальних агентів та архітектур, що спираються на це поняття, можна протестувати та проаналізувати якість змін, навантаживши мережу і конфігуруючи її параметри.

Агентом вважаємо все, що може розглядатися як сприймаюче своє середовище за допомогою датчиків і впливає на це середовище за допомогою виконавчих механізмів [1].

Основним напрямком цієї роботи є застосування методів штучного інтелекту для налаштування мережі блокчейну і підбору найефективніших комбінацій параметрів з точки зору швидкості та задіяного ресурсу обчислень. Головною метою виступає збір та аналіз метрик. Такий аналіз надасть можливість зробити висновки, наскільки безпечною та ефективною є децентралізована система, чи правильно обрано вектор розвитку сучасних блокчейнів. Висновок дозволить з більшою впевненістю інвестувати чи утриматись від інвестування в цю сферу, а також запропонувати краще рішення, якщо воно існує.

## 2. ВПЛИВ КРИПТОВАЛЮТ ТА ДЕЦЕНТРАЛІЗОВАНИХ СИСТЕМ НА НАВКОЛИШНЄ СЕРЕДОВИЩЕ

Найвідоміший вплив технології блокчейн на навколишнє середовище пов'язаний із споживанням енергії та, отже, можливим негативним впливом на клімат. Поточний стандартний процес перевірки транзакцій, заснований на алгоритмі підтвердження роботи, є «надзвичайно енергоємним», оскільки вимагає величезної кількості обчислювальних потужностей і, отже, електроенергії для роботи пов'язаних комп'ютерних розрахунків. Більш широке використання технології блокчейн могло б протидіяти зусиллям із пом'якшення кліматичних змін, оскільки електроенергія в усьому світі в основному виробляється з викопного палива. У 2016 році виробництво з горючих видів палива все ще становило 67,3% від загального світового валового виробництва електроенергії.

Приклад Bitcoin є особливо показовим. Порівняно з альтернативними методами оплати, біткоїн був у 20 000 разів більш енергоємним, ніж Visa. У 2019 році, за даними деяких аналітиків, енергія, споживана для кожної транзакції біткоїну, зросла до 635 кВт/год, що дорівнює електроенергії, яка могла б забезпечити енергією приблизно 21 домогосподарство США протягом 1 дня. Згідно з останніми дослідженнями, споживання електроенергії біткоїнами становить від 20 до 80 ТВт-год на рік, або приблизно 0,1-0,3% світового споживання електроенергії. Наприклад, Cambridge Bitcoin Electricity Consumption Index (СВЕСІ) наводить цифру в 64 ТВт-год на рік (СВЕСІ, 2019), що перевищує річне споживання електроенергії Швейцарією або всіма електромобілями в усьому світі (58 ТВт-год) у 2018 році. Ці оцінки слід тлумачити з обережністю через методологічні проблеми, обмежену доступність даних і дуже різноманітні умови в галузі. Крім того, ці цифри все ще набагато нижчі, ніж інші кінцеві споживачі, такі як охолодження, яке спожило 2 020 ТВт-год електроенергії в 2016 році.

Тим не менш, біткоїн - це лише одна криптовалюта, яка є лише одним із застосувань блокчейну. Важливо знайти спосіб зменшити споживання енергії для перевірки транзакцій. Заміна оригінального механізму консенсусу, тобто алгоритму підтвердження роботи, на інші підходи (такі як алгоритми «підтвердження частки», «підтвердження повноважень» або «підтвердження вичерпаного часу») було запропоновано. Деякі криптовалюти та блокчейн-програми вже покладаються на ці альтернативи. Проте все ще потрібна ретельна оцінка кожного механізму та його енергетичного впливу та енергоефективності. Перехід на більш екологічні джерела енергії та розробка обчислень, які потребують менше енергії, є іншими варіантами, які слід. Також можна було б зменшити енергоспоживання біткоїна шляхом реструктуризації способу стимулювання обслуговування блокчейну.

Як і інші нові технології на основі ІКТ, блокчейн також викликає занепокоєння щодо електронних відходів (електронних відходів). Майнерам-конкурентам потрібне все більш ефективне обладнання для майнінгу, що призводить до швидкого старіння приблизно кожні 1,5 року. З моменту свого створення апаратне забезпечення для майнінгу біткоїнів вже перейшло від використання центральних процесорів до графічних процесорів, програмованих вентильних матриць та інтегральних схем для конкретних програм. Приблизні оцінки показують, що біткоїн створює 135 г електронних відходів за транзакцію, що в 30000 разів більше, ніж транзакція Visa [2].

15 вересня 2022 року сталася подія «The Merge», що означало фактично зміну механізму консенсусу в блокчейні Ethereum [3]. Злиття було об'єднанням початкового рівня виконання Ethereum (основної мережі, яка існувала з моменту створення) з його новим консенсусним рівнем підтвердження частки, Beacon Chain. Це усунуло потребу в енергоємному майнінгу, а натомість дозволило захистити мережу за допомогою

розставленого EТН. Це був справді захоплюючий крок у реалізації бачення Ethereum — більше масштабованості, безпеки та стійкості. Тому актуальність дослідження процесу переходу з одного механізму на інший є високою.

Протестувати систему глобально та порівняти результати досліджень не є можливим в масштабах одного дослідника, тому було прийнято рішення працювати з системою локально на машині, імітуючи реальне навантаження. Змінити значення параметрів мережі для знаходження найефективніших комбінацій може допомогти агентна архітектура.

З використанням показників продуктивності, які втілюють в собі критерії оцінки успішної поведінки агентів, можна визначити яким чином використовується обчислювальний ресурс, що в свою чергу призводить до споживання електроенергії.

### **3. ТЕХНІЧНІ ІНСТРУМЕНТИ ДЛЯ РЕАЛІЗАЦІЇ АГЕНТНОЇ АРХІТЕКТУРИ В МЕРЕЖІ БЛОКЧЕЙН**

Для того, щоб розгорнути локальну версію блокчейну були використані наступні технології: Puppeth, Vagrant, Nodejs [4-6].

Створення приватного блокчейну зараз не є тривіальним завданням. Щоб розгорнути мережу, потрібно багато часу та зусиль. Хоча з'являються сервіси, які значно спрощують це завдання, але ми хочемо контролювати приватну мережу та мати повну можливість запускати та підтримувати її всередині. Зробити це на 100% вручну було б надзвичайно трудомістким, якби не зручний інструмент від групи Ethereum під назвою puppeth.

Puppeth — це майстер CLI, який допомагає створити нову мережу Ethereum аж до генезису, завантажувальних вузлів, підписувачів, ethstats, крана, інформаційної панелі тощо, без клопоту, який зазвичай потребує налаштування всіх цих служб одну за одною. Puppeth використовує ssh для підключення до віддалених серверів і створює свої мережеві компоненти з контейнерів докерів за допомогою docker-compose. Користувач керується процесом за допомогою майстра командного рядка, який автоматично виконує важку роботу та конфігурує топологію.

Vagrant — це програмний продукт із відкритим вихідним кодом для створення та підтримки портативних віртуальних середовищ розробки програмного забезпечення, який допоможе нам з розташуванням відповідних вузлів мережі. Він використовує «Provisioners» і «Providers» як будівельні блоки для керування середовищами розробки. Провайдери — це інструменти, які дозволяють користувачам налаштовувати конфігурацію віртуальних середовищ. Puppeth і Chef є двома найпоширенішими провайдерами в екосистемі. Постачальники — це служби, які Vagrant використовує для налаштування та створення віртуальних середовищ. Підтримка віртуалізації VirtualBox, Hyper-V і Docker поставляється з Vagrant, тоді як VMware і AWS підтримуються через плагіни. Для контролювання і автоматизації процесу було обрано варіант написання свого агенту та агентної архітектури, використовуючи мову програмування Node.js.

### **4. АНАЛІЗ РЕЗУЛЬТАТІВ ТЕСТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ**

Вирішення задачі полягало в декількох етапах, тому опишемо їх детально.

Перший етап – створення десяти віртуальних машин, щоб запустити на них програму та додати в спільну мережу. Сконфігуровано та налаштовано порти, які будуть приймати і з яких будуть віддаватись команди та дані.

Другий етап полягає у створенні вузлів блокчейну Ethereum на кожній віртуальній машині. Для цього було застосовано майстра CLI Puppeth, який дав готовий інтерфейс.

Також програма згенерувала свої унікальні адреси в мережі. Коли програми були готові до отримання вказівок, а акаунти ініціалізовані – зроблено додавання їх у мережу.

Третій етап – створення такої ж мережі, але з proof-of-stake механізмом консенсусу. Для цього розподілення було 5 master вузлів та 5 slave вузлів.

Етап номер чотири заключає в собі написання інтелектуального агента, який наче ляльковод буде смикати за ниточки – відправляти запити на мережу, створювати блоки, після чого змінюючи параметри системи в залежності від результату (Рис. 1).

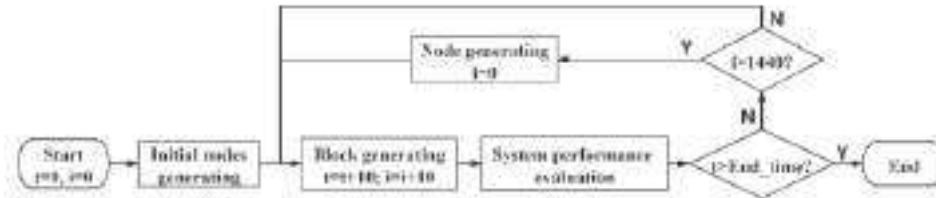


Рисунок 1. Структура агентної моделі

Для того, щоб результат був більш наочним – було додано ще огляд proof-of-authority механізму і сконфігуровано за прикладом proof-of-work.

Ефективність трьох консенсусних протоколів порівнюється на основі наступних параметрів: Transaction Cost and Gas Cost; Transactions Per Second (TPS); Electricity usage per transaction.

Зібрані метрики було оформлено у вигляді таблиць та графіків для наочності. Інструмент для моніторингу стану мережі Ethstats одразу показував кількість транзакцій на секунду для кожного алгоритму (Табл. 1). Також мануально було відображено графік порівняння Total Gas Cost, який показує ефективність того чи іншого підходу (Рис. 2).

Таблиця 1. Кількість транзакцій на секунду в залежності від механізму консенсусу

Механізм консенсусу	Кількість транзакцій на секунду
PoA	2380
PoW	395
PoS	4613

Фінальним порівнянням виступили результати метрик, отримання яких власне і ставилось за завдання: кількість витраченої електроенергії (Табл. 2).

Таким чином можемо сказати, що proof-of-stake концепт дійсно є значно енергоефективним та перехід на нього є доцільним, оскільки шкоди навколишньому середовищу завдається менше. Мінусом цього підходу є більша складність в реалізації та відсутність довготривалих випробувань в реальних умовах, тобто практично не доведено стійкість до зовнішніх або внутрішніх атак, лише описано теоретично.

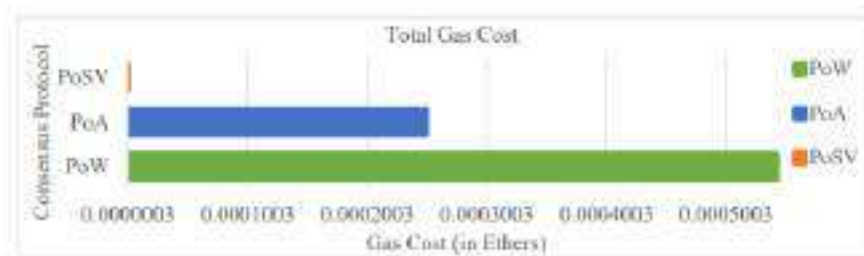


Рисунок 2. Порівняння Total Gas Cost кожного механізму консенсусу

Таблиця 2. Кількість електроенергії для проведення однієї транзакції в залежності від механізму консенсусу

Механізм консенсусу	Кількість використаної електроенергії (КВт-годин/транзакція)
PoA	1.1783
PoW	1.2321
PoS	0.0201

## 5. ВИСНОВКИ

Поставлена задача є досить складною, оскільки аналіз великою мережі за рахунок екстраполяції локальних результатів може містити похибку. Експеримент проводиться на одній машині, імітуючи багато з'єднань, але реальна мережа набагато більша, з більшими складовими як смарт контракти. Також на отримані результати впливає кількість вузлів та електронних гаманців у системі. Логічно, що реальні умови відрізняються від «лабораторних», але навіть з отриманих результатів можемо побачити різницю у споживанні ресурсів. Офіційні цифри, які наводить головний сайт Ethereum, зі зменшенням кількості витраченої електроенергії кажуть про скорочення витрат на 99.8% [3]. Тобто за словами розробників Ethereum, споживання має зменшитись у 50 разів. Отримані результати кажуть про зменшення споживання майже в 60 разів. Враховуючи похибку «стерильності» умов, можна зробити висновок, що офіційні цифри досить правдиві. Агентна архітектура показала себе як життєздатна для проведення таких експериментів, збору метрик та налаштування мережі, автоматизувавши роботу та значно зменшивши втручання людини, залишаючи їй лише порівняння кінцевих чисел.

## ПЕРЕЛІК ВИКРИСТАНИХ ДЖЕРЕЛ

1. Stuart J.Russel and Peter Norvig. Artificial Intelligence. A modern approach. Second edition. 2006. С. 75-440, 863-1010.
2. Mastercard [Електронний ресурс]. URL: <https://www.mastercard.com/news/perspectives/2021/why-mastercard-is-bringing-crypto-onto-our-network/>
3. Ethereum платформа [Електронний ресурс] – URL: <https://ethereum.org>
4. Official golang implementation of the Ethereum protocol [Електронний ресурс]. - URL: <https://github.com/puppeth>
5. Офіційна сторінка програмного забезпечення Vagrant [Електронний ресурс] – URL: <https://www.vagrantup.com/>
6. Офіційна сторінка мови програмування Node.js [Електронний ресурс] – URL: <https://nodejs.org/en/>

# ВИЯВЛЕННЯ АНОМАЛІЙ МЕТОДАМИ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

Хомич О.Р.<sup>1</sup>, Недашківська Н.І.<sup>2</sup>

КПІ ім. Ігоря Сікорського, Київ, Україна

<sup>1</sup>khomych.alexander@iit.kpi.ua [0000-0002-1977-3973],

<sup>2</sup>nedashkovskaya.nadezhda@iit.kpi.ua [0000-0002-8277-3095]

**Проведено порівняльний аналіз результатів виявлення різних видів аномалій різними методами інтелектуального аналізу даних. Колективні аномалії описуються групою векторів даних, де кожен окремий вектор може бути допустимим. Контекстні аномалії часто описуються метаданими, ними можуть бути часові, просторові чи просторово-часові ознаки навколишнього середовища. Показано, що моделі на основі рекурентних нейронних мереж LSTM та гібридні з елементами LSTM більш точно виявляють контекстні, колективні аномалії, викиди у часових рядах порівняно з іншими методами інтелектуального аналізу даних.**

**Ключові слова:** розпізнавання аномалій, рекурентні нейронні мережі, LSTM, ансамбль нейронних мереж LSTM, штучний інтелект

## 1. ВСТУП

Аномалії створюють проблему в різних областях застосування. Вони часто призводять до зниження продуктивності системи і можуть викликати нестабільність і збої. Часто причинами аномалій є невідомі ефекти в складних системах. Завдяки прогресу досліджень у сфері машинного навчання останніми роками було запропоновано широкий спектр нових підходів до виявлення аномалій. Різні архітектури глибоких нейронних мереж, зокрема архітектури на основі довготривалої короткострокової пам'яті (LSTM), були розроблені і виявилися здатними вирішувати різноманітні складні задачі розпізнавання [1, 2].

На сьогоднішній день аномалії виявляють в основному методами теорії статистики, інформації, алгоритмами класифікації та кластеризації [1, 3]. Зокрема, для цієї задачі застосовують аналіз головних компонент (PCA), машини опорних векторів (SVM), алгоритм k найближчих сусідів (kNN), різні типи кореляційного аналізу, а також додаткові методи, такі як ковзні вікна, які поєднуються з вищезгаданими для виявлення змінних у часі аномалій.

Для вирішення проблеми розпізнавання складних контекстних аномалій з характеристиками, що змінюються в часі, в останні роки застосовують нові архітектури рекурентних нейронних мереж [3 – 8], нейронні мережі типу машин Больцмана та автокодувальників [6]. Глибокі нейронні мережі використано при виявленні аномалій у медицині [5], системах автоматизації виробництва [4,8], системах інтернету речей (IoT) [7].

Цю роботу присвячено дослідженню та порівняльному аналізу результатів розпізнавання різних видів аномалій різними методами інтелектуального аналізу даних.

## 2. КЛАСИФІКАЦІЯ АНОМАЛІЙ

Точкові аномалії (Рис. 1 ліворуч) можуть виявлятися методами статистики: для цільових параметрів рахуються значення щільності ймовірності та аналізуються процентилі.

Колективні аномалії характеризуються групою векторів даних, де кожен окремий вектор допустимий, але група цих векторів вказує на нерегулярність. Тому виявлення колективних аномалій базується на аналізі послідовностей даних. У [2] запропоновано метрику, яка об'єднує ступені відхилення кожного окремого вектору з групи, для випадку виявлення колективних аномалій у багатовимірних часових рядах. Якщо значення цієї сукупної метрики перевищує поріг, причому поріг змінюється з часом, то робиться висновок про присутність такої аномалії.

Контекстні аномалії (Рис. 1 праворуч) описуються як окремими векторами даних, не будучи при цьому точковими аномаліями, так і групами векторів даних. В другому випадку група вказує на нерегулярність у межах навколишніх векторів даних або груп векторів даних, які називаються контекстом. В цьому полягає відмінність від колективної аномалії.

Також, на відміну від колективних аномалій, які описуються своєю внутрішньою структурою або змістом, виявлення контекстних аномалій сильно залежить від короткострокових і довгострокових характеристик навколишніх зовнішніх структур даних [3]. У випадку багатовимірних часових рядів, контекстом вектору даних або групи векторів даних є об'єднання всіх навколишніх векторів даних, які лежать у визначеному часовому горизонті. Контекстні аномалії також часто описуються метаданими, які називають атрибутами аномалії [5, 6]. Ними можуть бути будь-які часові, просторові чи просторово-часові ознаки навколишнього середовища. У системах автоматизації виробництва атрибутами аномалії розглядають дані про виробничий процес і пов'язані з ним параметри процесу [4,8].

Оцінювання контекстної аномалії проводять на основі показників, що засновані на відстані, та методом ковзного вікна, коли значення відстані до попереднього вікна перераховується за кожною новою вибіркою даних [7]. Якщо показник перевищує динамічний (змінний у часі) поріг, робиться висновок про присутність контекстуальної аномалії.

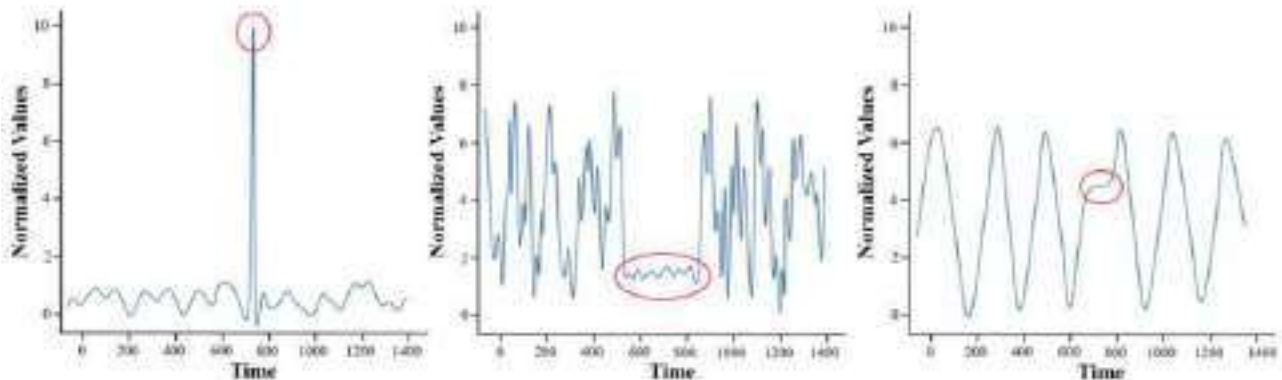


Рисунок 1. Точкова аномалія (ліворуч), колективна аномалія (посередині) і контекстна аномалія (праворуч) [9]

### 3. РЕКУРЕНТНІ НЕЙРОННІ МЕРЕЖІ ТИПУ LSTM

Складову мережі LSTM – комірку пам'яті (Рис. 2) – розроблено в [10] для вирішення проблеми зникаючого градієнту, яка виникає зі звичайними рекурентними мережами і призводить до того що частина ваг припиняє оновлюватися під час процесу навчання, в результаті стає важко вивчати довгострокові залежності. В комірці LSTM вектор стану розщеплюється на два вектори:  $h(t)$  – короткостроковий і  $z(t)$  – довгостроковий стани. Тому мережа LSTM може дізнатися яку інформацію слід зберігати в довгостроковому стані, а яку відкидати. Під час перетину мережі зліва направо довгостроковий стан  $z(t - 1)$  спочатку

проходить через шлюз забування forget gate, де відкидаються деякі спогади, і потім до нього додаються (у шлюзі add gate) нові спогади, що були обрані вхідним шлюзом input gate. Результат  $z(t)$  подається на вихід комірки без будь-яких подальших перетворень.

В комірці LSTM на кожному часовому кроці одні спогади відкидаються, а інші додаються. Ця процедура зумовлена обставиною, що жодна пам'ять не є нескінченною і що людська пам'ять як взірець також має обмежені можливості. Після операції додавання (add gate) довгостроковий стан  $z(t)$  копіюється і пропускається через функцію гіперболічного тангенсу, після чого результат фільтрується вихідним шлюзом. Остаточним результатом є короткостроковий стан  $h(t)$ , який дорівнює виходу комірки  $y(t)$  для даного часового кроку  $t$ .

Отже, стан комірки LSTM вміщує зменшене та зважене представлення історичної вхідної інформації. Останні введені дані не обов'язково домінують при створенні вихідних сигналів, оскільки при створенні виходу враховуються також важливі минулі події.

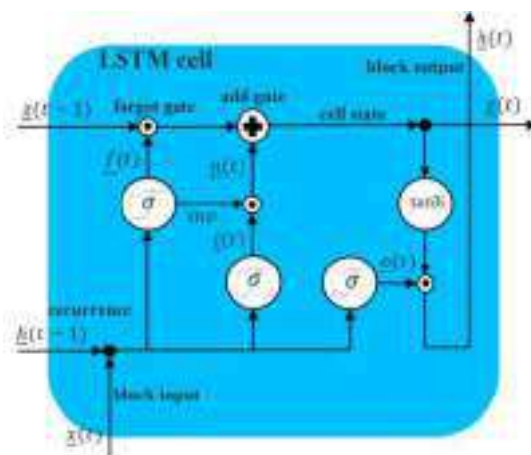


Рисунок 2. Архітектура комірки LSTM [10]

#### 4. ПОРІВНЯННЯ РЕЗУЛЬТАТІВ РОЗПІЗНАВАННЯ АНОМАЛІЙ РІЗНИХ ТИПІВ

Проведено наступні експерименти.

Експеримент 1. На вхід подавався багатовимірний часовий ряд без розмітки. Досліджувалися контекстні аномалії. В результаті ансамбль нейронних мереж LSTM [11] показав кращі результати за просту рекурентну нейронну мережу RNN за показниками точності (precision), повноти (recall) та F1-score.

Експеримент 2. На вхід подавався багатовимірний часовий ряд з розміткою на класи «нормальний» і «аномалія». Досліджувалися колективні та контекстні аномалії. Аналізувалася гібридна модель LSTM – SVM [12], яка складається з двох частин: 1) мережа LSTM – прогнозує динаміку системи, 2) модель опорних векторів (SVM) – використовується як класифікатор аномалій. Ця гібридна модель реалізує адаптивний і самонавчаючийся механізм виявлення. Модель LSTM – SVM показала кращі результати ніж традиційна SVM. Якість оцінювалася за показником AUC та ROC-кривою.

Експеримент 3. На вхід подавався одновимірний часовий ряд без розмітки. Досліджувалися колективні аномалії та викиди. В результаті дуальний LSTM [13] показав кращі результати ніж інші розглянуті методи за точністю (precision), повнотою (recall) та F1-score.

При наявності часового ряду з розміткою на класи, аналізувалися відхилення виходів, спрогнозованих мережею, від відомих реальних значень цільової змінної, і на основі цього приймалося рішення про присутність аномалії.

## 5. ВИСНОВКИ

Проведено порівняльний аналіз результатів розпізнавання різних видів аномалій різними методами інтелектуального аналізу даних. Розглянуто такі нові розробки як ансамбль нейронних мереж LSTM, гібридна модель LSTM – SVM, дуальний LSTM. Показано, що ці моделі на основі рекурентних нейронних мереж LSTM та гібридні з елементами LSTM більш точно виявили аномалії всіх розглянутих типів завдяки здатності моделей вивчати часові залежності та фіксувати їх у представленні стану низької розмірності, моделювати стаціонарні і нестаціонарні часові залежності.

### ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ahmed M., Naser Mahmood A., Hu J. A survey of network anomaly detection techniques. *J. Netw. Comput. Appl.* 2016. Vol. 60. P. 19–31, <http://dx.doi.org/10.1016/j.jnca.2015.11.016>.
2. Akoglu L., Tong H., Koutra D. Graph based anomaly detection and description: a survey. *Data Min. Knowl. Disc.* 2015. Vol. 29. P. 626–688, <http://dx.doi.org/10.1007/s10618-014-0365-y>.
3. Bontemps L., Cao V.L., McDermott J., Le-Khac N.A. Collective anomaly detection based on long short term memory recurrent neural networks. *International Conference on Future Data and Security Engineering*. 2016. P. 141–152, [http://dx.doi.org/10.1007/978-3-319-48057-2\\_9](http://dx.doi.org/10.1007/978-3-319-48057-2_9).
4. Canizo M., Triguero I., Conde A., Onieva E. Multi-head CNN–RNN for multi time series anomaly detection: an industrial case study. *Neurocomputing*. 2019. Vol. 363. P. 246–260, <http://dx.doi.org/10.1016/j.neucom.2019.07.034>.
5. Chalapathy R., Chawla S. Deep Learning for Medical Anomaly Detection – A Survey. 2019. P. 1–50, arXiv:1901.03407.
6. Chandola V., Banerjee A., Kumar V. Deep learning for anomaly detection: a survey. *ACM Comput. Surv.* 2009. Vol. 41 (3). P. 1–58, <http://dx.doi.org/10.1145/1541880.1541882>.
7. Cook A.A., Misirlı G., Fan Z. Anomaly detection for IoT time-series data: a survey. *IEEE Internet Things J.* 2019. Vol. 7, no. 7. P. 6481–6494, <http://dx.doi.org/10.1109/JIOT.2019.2958185>.
8. Ding S., Morozov A., Vock S., Weyrich M., Janschek K. Model-based error detection for industrial automation systems using LSTM networks. *IMBSA: Model-Based Saf. Assess.* 2020. P. 212–226, [http://dx.doi.org/10.1007/978-3-030-58920-2\\_14](http://dx.doi.org/10.1007/978-3-030-58920-2_14).
9. Kim T.Y., Cho S.B. Web traffic anomaly detection using C-LSTM neural networks. *Expert Systems With Application.* 2018. Vol.106. P. 66–76, <http://dx.doi.org/10.1016/j.eswa.2018.04.004>.
10. Hochreiter S., Schmidhuber J. Long short-term memory. *Neural Comput.* 1997. Vol. 9 (no. 8). P. 1735–1780, <http://dx.doi.org/10.1162/neco.1997.9.8.1735>.
11. Malhotra P., Vig L., Shrof G., Agarwal P. Long short term memory networks for anomaly detection in time series. In: *Proceedings of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*. 2015. P. 89–94.
12. Ergen T., Mirza A.H., Kozat S.S. Unsupervised and Semi-supervised Anomaly Detection with LSTM Networks. 2017. arXiv:1710.09207.
13. Lee M.-C., Lin J.-C., Gan E.G. ReRe: a lightweight real-time ready-to-go anomaly detection approach for time series. In: *2020 IEEE 4th Annual Computers, Software and Applications Conference (COMPSAC), Madrid, Spain.* 2020. P.322–327, <http://dx.doi.org/10.1109/COMPSAC48688.2020.0-226>.

# **КЛАСИФІКАЦІЯ ВАКАНСІЙ ВІДПОВІДНО ДО КЛАСИФІКАТОРА ПРОФЕСІЙ МЕТОДАМИ ШТУЧНОГО ІНТЕЛЕКТУ**

Цимбал Ю.О.<sup>1</sup>, Недашківська Н.І.<sup>2</sup>

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

<sup>1</sup>tsymbal.julia@lil.kpi.ua, <sup>2</sup>nedashkovskaya.nadezhda@lil.kpi.ua [0000-0002-8277-3095]

**Сучасні способи пошуку роботи значно змінили ринок праці України за останні роки. Інформація, що зазвичай збирається держстатом, є доцільною, але неповною. Значна частина інформації про потреби роботодавців, запит конкретних професій та сфер діяльності не збирається на даний момент в Україні. Мета роботи полягає у створенні комп'ютерної системи для збору, підготовки та класифікації зібраних вакансій згідно до класифікатора професій. Розроблено гібридний підхід до обробки неструктурованих текстових даних щодо ринку праці за допомогою методів машинного навчання.**

**Ключові слова: ринок праці, онлайн сегмент ринку праці, великі дані, інтелектуальний аналіз, обробка природної мови, класифікація.**

## **1. ВСТУП**

Основним напрямком цієї роботи є застосування різних методів штучного інтелекту та обробки природної мови як потужного інструменту інтелектуального аналізу даних у демографічній сфері населення, а саме ринку праці. Розвиток економічних відносин стрімко зростає, відповідно і робочі місця масштабуються, змінюються вимоги до працівників та шукачів роботи, створюються нові професії та трансформуються існуючі. Наразі в демографічних дослідженнях в Україні науковці тільки починають використовувати різноманітні інструменти машинного навчання та штучного інтелекту, тому дана тема є дуже актуальною. Такий автоматизований метод аналізу ринку праці дасть змогу більш детально та прицільно досліджувати професійну сферу. Очищена та належним чином систематизована інформація онлайн джерел щодо вакансій статистично значуще доповнює наявну інформацію про офіційні вакансії Державного центру зайнятості, їх вимог щодо знань і навичок робочої сили, типу трудових контрактів, досвіду, трудових обов'язків та особистих якостей. Головна мета та ціль даного продукту – це можливість оцінити та аналізувати ринок праці в Україні у реальному часі та базуючись на реальних даних. Такий аналіз надасть змогу ширше та глибше зрозуміти ситуацію працевлаштування.

## **2. СТАН РИНКУ ПРАЦІ УКРАЇНИ ТА ЙОГО ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ**

Інформаційне забезпечення дослідників, управлінців та учасників ринку праці в Україні надається кількома ресурсами, зокрема державними та недержавними інституціями та нерегулярно міжнародними дослідниками. Переважна більшість даних щодо ринку праці збирається та агрегується спеціальними державними інституціями, а саме Державною службою статистики України (далі ДССУ) та Державним центром зайнятості (ДЦЗ).

ДССУ проводить свою діяльність згідно державного законодавства за усіма нормами та правилами ведення статистичних оцінок. Тому дані, що подаються як результат опитування та агрегації є достовірними та релевантними, надійними та достатньо точними. Статистична методологія публікується у відкритий доступ, тому з цими методами та підходами можна вільно ознайомитися. У сфері ринку праці ДССУ збирає дані підприємств та організацій щодо середньооблікової кількості штатних працівників, обсягів прийнятих та звільнених, розмір використаного та невикористаного робочого часу, обсяги вимушеної неповної зайнятості, середню заробітну плату, фонд оплати праці та його структуру, розмір заборгованості працівникам заробітної плати. Всі ці показники можуть бути кореспондованими між собою та представлені за видами економічної діяльності, регіонами, організаційно-правовими формами господарювання, формами зайнятості. Опубліковані дані обов'язково класифікуються згідно різних видів державних класифікаторів відповідно до тематики даних. Використовують зокрема класифікатор видів економічної діяльності (КВЕД) ДК 009:2010, організаційно-правових форм господарювання (КОПФГ) ДК 002:2004, кодифікатор адміністративно-територіальних одиниць та територіальних громад (КАТОТТГ) [1].

Отже, ДССУ забезпечує велику кількість коректно зібраних та оброблених даних щодо ринку праці по підприємствах та організаціях. Проте залишається відкрите питання щодо професійної кваліфікації працівників, кількості відкритих вакантних робочих місць на підприємствах та на ринку праці загалом.

Цю прогалину частково заповнює Державний центр зайнятості – установа, що веде перелік непрацевлаштованих громадян, допомагає їм підвищити кваліфікацію та набути потрібні професійні навички. Також ДЦЗ має відкритий інформаційний портал для публікації та поширення даних про вакантні робочі місця по Україні [2]. Державний центр зайнятості – популярний ресурс для розміщення пропозицій робочих місць роботодавцями та пошуку роботи безробітними. Основна перевага вищезгаданого порталу в тому, що дані про вакансії уніфіковані, структуровані та найголовніше назви публікуються згідно з Державним класифікатором професій. З іншого боку на цьому ресурсі можна знайти переважно прості професії, що не потребують спеціальних навичок та підготовки та мало привабливі професії.

В той самий час безліч запитів та пропозицій різнопланових робочих місць можна знайти на недержавних онлайн ресурсах пошуку роботи. Станом на сьогодні це єдине найвагоміше джерело інформації про актуальний стан ринку праці. Відповідно до цього без використання таких даних інформаційна картина українського ринку праці є неповною та викривленою.

Отже, провівши дослідження якості та способів інформування ринку праці України очевидним стає те, що наявні на сьогодні дані та підходи їх збору не дають можливості повно та цілковито оцінити стан ринку праці. Наразі ніяка структура чи державна організація не збирає та не надає повноцінно дані, що стосуються вільних робочих місць, запиту роботодавців, кількості шукачів роботи та найцікавіше – розподіл актуальних сфер діяльності та професій.

Збір даних з порталів онлайн пошуку роботи та класифікації згідно державного класифікатора професій дозволить побудувати на об'єднаних даних показники українського ринку праці, які точніше показують ринкову ситуацію, перспективи його розвитку. Крім того, забезпечення єдиної класифікаційної платформи для державних і недержавних даних дозволить доповнити знання експертів та державного регулювання новою інформацією щодо нових навичок і компетенцій, що стають актуальними на ринку праці. Це дозволить більш обґрунтовано формувати систему професійної підготовки та вищої освіти, планувати державне замовлення на фахівців, вчасно заповнювати прогалини дефіцитних спеціалістів.

### **3. ТЕХНІЧНІ ІНСТРУМЕНТИ ДЛЯ РЕАЛІЗАЦІЇ АЛГОРИТМУ КЛАСИФІКАЦІЇ ПРИРОДНОЇ МОВИ**

Обробка природної мови (Natural Language Processing, NLP) – перетин машинного навчання та математичної лінгвістики, спрямований на вивчення методів аналізу та синтезу природної мови. NLP дозволяє краще розуміти запити користувачів та аналізувати корпоративну інформацію. Використання даної технології гарантує, що у кожного користувача є доступ до найбільш актуальних, корисних джерел інформації, які б інакше залишилися прихованими у величезних обсягах даних.

Робота була сконцентрована на використанні методів та підходів, що походять з розділу лінгвістики – дистрибутивної семантики. Дистрибутивна семантика (distributional semantics) займається обчисленням ступеня семантичної близькості між лінгвістичними одиницями на основі їх розподілу у великих масивах лінгвістичних даних або так званих текстових корпусах [3]. Одне з основних понять – розподільна семантична модель (distributional semantic model - DSM) – це масштабована та/або трансформована матриця спільного входження  $M$ , у якій кожен рядок  $x$  представляє розподіл цільового терміна в різних контекстах. Головний принцип використання методів дистрибутивної семантики полягає у тому, що кожному слову надається свій контекстний вектор. Множина векторів формує словесний векторний простір. Як спосіб представлення моделі використовуються векторні простори з лінійної алгебри. Інформація про дистрибуцію лінгвістичних одиниць подається у вигляді багаторозрядних векторів, які утворюють векторний словесний простір. Вектори відповідають лінгвістичним одиницям (словам чи словосполученням), а виміри – контекстам. Координати векторів – це числа, що показують скільки разів слово або словосполучення зустрілося в даному контексті. Семантична відстань між поняттями, вираженими словами природної мови, зазвичай обчислюється як косинусна відстань між векторами словесного простору [4]. Після проведення такого аналізу виявляють найближчі за змістом слова стосовно досліджуваного слова.

У обробці природної мови (NLP) word embedding – термін, який використовується для представлення слів для аналізу тексту, як правило, у формі дійсного вектора, який кодує значення слова таким чином, що слова, розташовані ближче у векторному просторі, є подібними за значенням. Тобто word embedding – це техніка отримання DSM-вектору.

Найпопулярніший алгоритм обробки природної мови – word2vec [5, 6]. Він використовує модель нейронної мережі для вивчення асоціацій слів із великого корпусу тексту. Після навчання така модель може виявляти слова-синоніми або пропонувати додаткові слова для окремого речення. Модель word2vec представляє кожне окреме слово певним вектором, який підбирається, щоб відображати семантичні та синтаксичні якості слів. Рівень семантичної подібності між словами, представленими цими векторами, зазвичай визначається на основі косинусної відстані. У word2vec представлено дві модельні архітектури для створення цих розподілених представлень слів: модель неперервного пакету слів (Continuous Bag Of Words, CBOW) та неперервного скіп-граму (Skip-gram) [5]. В обох цих архітектурах виконується ітераційне проходження по всьому корпусу тексту і розглядаються як окремі слова, так і ковзне вікно контекстних слів, що оточують обрані слова.

### **4. РОЗРОБКА АЛГОРИТМУ КЛАСИФІКАЦІЇ ТА РЕЗУЛЬТАТИ**

В роботі пропонується комплексний метод вирішення поставленої задачі, який складається з кількох етапів. Перший етап – збір даних. Наразі немає спільної бази даних

вакансій українського ринку праці. Кожен приватний інтернет-портал з пошуку роботи має свою базу, яка не узгоджена з іншими аналогічними посередниками пошуку роботи. Тому перший крок пропонованого методу – це визначити як можна зібрати дані з кількох онлайн джерел, а також встановити критерії вибору таких інтернет-порталів. Критерії необхідні для того, щоб зменшити розбіжність зібраних даних, оскільки кожен вебсайт має свою власну структуру та види, і формат опублікованої інформації відрізняється.

Другий крок – провести аналіз можливих методів збору інформації з вебресурсів та обрати підхід, який буде більш універсальним та не потребуватиме великих витрат: часу, знань, фінансів. Базуючись на цих критеріях та беручи до уваги особливості онлайн порталів було обрано методологію вебскрепінгу. Вебскрепінг використовується для отримання тексту з веб-сторінок. Програмне забезпечення для сканування веб-сторінок розробляється для розпізнавання різних типів вмісту на веб-сайті та для отримання та зберігання лише типів вмісту, вказаних користувачем, наприклад назви статей чи авторів із веб-сайту новин або ціни й описи продуктів із комерційного веб-сайту. Можна використовувати комерційне програмне забезпечення або мови програмування. Для цієї роботи був розроблений програмний продукт для збору даних, написаний на мові програмування Python та з використанням бібліотеки Selenium для імітування дії користувача у мережі. Особливість розробленої програми в тому, що алгоритм збору даних універсальний, але практична реалізація повинна коригуватися відповідно до вебресурсу. Це зумовлено тим, що кожен онлайн портал пошуку роботи зроблено за різними архітектурами та з використанням різних веб елементів. У результаті застосування розробленого інструменту вдалося зібрати дані щодо актуальних вакансій.

Третій етап – підготовка даних. Підготовка або препроцесинг – важлива та невід’ємна складова будь-якого циклу розробки програмного забезпечення. Від якості даних залежить результат класифікації чи прогнозу, відповідно до задачі. Для зібраного набору даних було використано усі основні елементи препроцесингу тестової інформації: вилучення спеціальних символів та стоп-слів української мови, вилучення пустих або незадовільно заповнених даних, та сортування даних згідно мови. Для публікації вакансій в Україні часто використовується не тільки українська мова, але й англійська та російська. Аналогічної підготовки потребує класифікатор професій, згідно з яким необхідно розділити зібрані вакансії.

Четвертий етап – розробка алгоритму класифікації вакансій. Для класифікації професій розроблено гібридний підхід з метою покращення ефективності класифікації та зменшення втручання людини. Використано різні методології автоматизації та штучного інтелекту, що включають методи обробки природної мови. Створено систему, яка використовує вказаний гібридний підхід для поєднання потужності онтологій і словників із гнучкістю алгоритмів машинного навчання (Рис. 1).



Таблиця 1. Відсоткове співвідношення співпадінь назви вакансії та назви професії в класифікаторі згідно різних методів перевірки

№	Метод визначення семантичної подібності	Відсоткове відношення співпадінь
1	Exact match	12%
2	Direct match	65%
3	Regular expression match	3%
4	FastText match (unsupervised)	16%

Отримані результати показують, що значну частину вакансій можна класифікувати різними комбінаціями порівнянь назв, але в будь-якому разі залишаються назви вакансій, які не узгоджені з класифікатором професій. Саме тому є сенс використовувати методи машинного навчання для досягнення повноцінного результату. Надалі та частина класифікованих вакансій може бути використана як навчальна вибірка для методів навчання з учителем. Це розширить спектр алгоритмів, які можна застосувати, та дасть змогу порівняти результати. Наприклад, навчання без учителя за алгоритмом FastText знаходить смислові схожості, використовуючи word embedding та обчислення косинусної відстані між закодованими словами. Таким чином є можливість встановити схожість між частинами тексту, навіть написані різним лексиконом.

## 5. ВИСНОВКИ

Аналіз світового досвіду засвідчує можливість використання великих даних для дослідження в реальному часі широкого кола проблем функціонування ринку праці: від оцінки загальної онлайн пропозиції робочих місць і їх класифікації за професіями та видами діяльності, до більш специфічних питань таксономії навичок, виявлення нових професій та розробки модельних траєкторій побудови кар'єри.

Загалом складається враження, що інформування зацікавлених осіб про український ринок праці є досить насиченим і різноманітним, торкається різних аспектів та відбувається досить регулярно. Однак, аналіз згаданих джерел щодо переліку та формату даних дає підстави для висновку, що дані цих джерел мало узгоджуються між собою, часто надають неактуальну інформацію, багато в чому дублюються, в той час як важливі моменти ринку праці залишаються без відповіді. В результаті, сьогодні неможливо сформулювати єдину і не суперечливу картину українського ринку праці, якою б могли одночасно користуватися державні структури, дослідники, населення та підприємства. Формування єдиної інформаційної системи ринку праці, де усі могли б знайти актуальну для себе інформацію, вимагає значних методичних та організаційних зусиль, подолання численних інституційних бар'єрів, узгодження методології збору та обробки даних, їх таксономії, а також застосування нових наукових інформаційних методів та технологій.

В цій роботі розроблено систему класифікації вакансій згідно класифікатора професій, яка базується на запропонованому гібридному підході. Така система наразі дає можливість регулярно, комплексно і в реальному часі отримувати актуальну інформацію щодо ринку праці в Україні.

Пропонований алгоритм класифікації включає етапи збору власних актуальних даних, що задовольняють умовам поставленого завдання, підготовки та попередньої обробки цих даних, моделі онтологій та методи машинного навчання для отримання фінального

результату – присвоєння кожній назві вакансії мітки класу згідно класифікатора. Складність цієї роботи полягає у тому, що, по-перше, немає готового набору даних, а отже необхідно попередньо збирати дані на різних онлайн-ресурсах та агрегувати. По-друге, немає розмічених даних, тому потрібно було створювати їх самостійно. По-третє, реальна оцінка результату класифікації може бути здійснена тільки експертом.

### **ПЕРЕЛІК ВИКРИСТАНИХ ДЖЕРЕЛ**

1. Державна служба статистики України. URL: <https://ukrstat.gov.ua/> (дата звернення 07.10.2022)
2. Державна служба зайнятості. URL: <https://www.dcz.gov.ua/> (дата звернення 07.10.2022)
3. Шарнін М. М., Сомін Н. В., Кузнецов І. П., Морозова Ю. І., Галина І. В., Козеренко Є. Б. Статистичні механізми формування асоціативних портретів предметних областей на основі природно-мовних текстів великих обсягів для систем здобуття знань. Інформатика та її застосування: журнал. - 2013. - Т. 7, вип. 2. - С. 92–99.
4. Schutze H. Dimensions of meaning Proceedings of Supercomputing'92. — 1992. — С. 787—796.
5. Mikolov, Tomas. Efficient Estimation of Word Representations in Vector Space, 2013.
6. Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep Learning. The MIT Press Cambridge, Massachusetts London, England, 2017.

# ІДЕНТИФІКАЦІЯ ПСИХІЧНИХ ЗАХВОРЮВАНЬ МЕТОДАМИ ОБРОБКИ ПРИРОДНОЇ МОВИ НА ОСНОВІ ДОПИСІВ КОРИСТУВАЧІВ У СОЦМЕРЕЖАХ

Цупрун І.Ю.<sup>1</sup>, Стусь О.В.<sup>2</sup>

<sup>1,2</sup> Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

<sup>1</sup>itsuprik@gmail.com, <sup>2</sup>o.stus@kpi.ua

**Методи обробки природної мови в останні роки отримали великий поштовх для розвитку та на сьогоднішній день, їх використання призвело до приголомшливих досягнень у різних сферах, від медицини до науки та розваг. Метою даної роботи є створення системи для ідентифікації психічних захворювань на основі дописів користувачів у соцмережах, тобто вирішити задачу класифікації текстів у вказаній предметній області. У роботі було описано та використано методи необхідні для розв'язання задачі класифікації тексту, а також оглянуто вже існуючі дослідження, що стосуються даної теми.**

**Ключові слова:** обробка природної мови, тексти, психічні захворювання, класифікація, ідентифікація, соцмережі.

## 1. ВСТУП

У сучасному світі, де розвиток технологій біжить нелюдськими темпами, все більше і більше людей страждає від психічних захворювань. Так, за даними Всесвітньої Організації Охорони Здоров'я (далі ВООЗ), у 2019 р. кожна восьма людина планети, тобто. загалом 970 мільйонів чоловік, страждають на психічні розлади, причому найбільш поширеними є тривожні та депресивні розлади. У 2020 р. на тлі пандемії COVID-19 кількість людей, які страждають на тривожні та депресивні розлади, значно зросла. За попередніми оцінками лише за рік поширеність тривожних і серйозних депресивних розладів збільшилася на 26% та 28% відповідно. Незважаючи на наявність ефективних методів профілактики та лікування, більшість людей із психічними розладами не мають доступу до ефективної медичної допомоги. Багато хто з них також стикається зі стигматизацією, дискримінацією та порушенням прав людини. Крім того, є люди які бояться звернутись по допомогу або просто не розуміють що потребують її. [1]

Метою даної роботи є дослідження методів машинного навчання, які допоможуть створити систему для ідентифікації психічних розладів на основі публікацій людини в соцмережах та повідомлень у месенджерах. Таким чином, після виявлення того чи іншого психічного захворювання система матиме можливість у не нав'язливій формі запропонувати людині допомогу, що дуже важливо з огляду на страх людей звертатись по допомогу та ділитись своїми проблемами.

Об'єктом дослідження стали дописи користувачів що мають психічні розлади у соцмережах. Предметом дослідження є підходи та методи вирішення задачі класифікації текстів.

## 2. МЕТОДИ ОБРОБКИ ПРИРОДНОЇ МОВИ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ КЛАСИФІКАЦІЇ ТЕКСТІВ

Класифікація тексту - це процес розуміння сенсу неструктурованого тексту та організації його за задалегідь визначеними категоріями (тегами). Одним з найпопулярніших завдань класифікації тексту є аналіз настроїв, який має на меті класифікувати неструктуровані дані за настроями. Інші завдання класифікації включають виявлення намірів, моделювання тем і виявлення мови. В даній роботі вирішується задача відношення тексту до мітки того чи іншого психічного розладу. Всього було розглянуто 8 класів, які відповідають восьми основним психічних розладам що виділяє ВООЗ: тривожні розлади, депресія, біполярні розлади, посттравматичні стресові розлади, шизофренія, розлади харчової поведінки, асоціальна поведінка та дисоційовані розлади, порушення розвитку центральної нервової системи (аутизм).

Процес вирішення будь-якої задачі класифікації текстів зазвичай складається з трьох основних етапів: попередня обробка тексту, векторизація слів, вирішення задачі класифікації.

Попередня обробка даних - це заключний етап збору та підготовки даних, який включає в себе трансформацію даних у доступний для розуміння формат. В обробці природної мови виділяють такі етапи попередньої обробки даних: чистка даних, зміна регістру, токенизація, видалення стоп-слів, стематизація, лематизація, N-грами. [2]

Багато алгоритмів машинного навчання і майже всі моделі глибокого навчання не здатні обробити простий текст в сирому вигляді. Для виконання будь-яких завдань, таких як класифікація, регресія, кластеризація і т.д., їм потрібні числові дані в якості вхідних даних. Крім того, з величезної кількості даних, які присутні в текстовому форматі, необхідно витягти якісь знання. Для перетворення текстових даних у числові були вигадані методи векторизації тексту.

Методи векторизації можна поділити на дві категорії: ті, що працюють на базі частоти та ті, що враховують контекст в якому вжито слово. Найкращим частотним методом вважається векторизація на базі метрики TF-IDF. В методі TF-IDF генерується матриця токенів документа, кожен стовпець якої представляє окреме унікальне слово. Кожна комірка не вказує частоту токена, а містить значення ваги, яке означає, наскільки важливим є слово для окремого текстового повідомлення або документа. Крім того, цей метод відрізняється від інших тим, що враховує не тільки входження слова в окремому документі, але і в усьому корпусі та надає більшої ваги подіям, що трапляються рідше, і меншої ваги очікуваним подіям. TF-IDF слова вказує на те, як часто лексема зустрічається в документі і наскільки унікальною є ця лексема для всього корпусу документів. [3]

Найбільш відомим способом контекстної векторизації став метод word2vec відкритий у 2013 році Томашом Міколовим. Ідея метода word2vec полягає у визначенні слова за його контекстом або контексту який відповідає слову за допомогою використання нейронних мереж. Ці два протилежні принципи отримали назви Skip Gram та Continuous Bag Of Words (CBOW) відповідно. Модель CBOW використовує контекст кожного слова як вхідні дані і намагається передбачити слово, що відповідає контексту. Розглянемо як приклад речення "Бажаю чудового дня". Нехай на вхід нейронної мережі подається слово "чудовий". У такому випадку, в якості цільового слова для нейронної мережі буде слово "день", використовуючи єдине контекстне вхідне слово "чудовий". Якщо бути точніше, на вхід в мережу подається вектор слова "чудовий" закодований за допомогою частотного методу векторизації. В процесі прогнозування цільового слова буде отримано векторне представлення цільового слова [4].

Алгоритми класифікації з використанням машинного навчання є останнім етапом у вирішенні задачі класифікації тексту, після попередньої обробки даних та векторизації, тобто приведення текстових даних до числових векторів. Після того, як було сформоване векторне представлення всіх розмічених текстових документів, їх можна використовувати для навчання класифікатора. Векторне представлення текстових документів передається класифікатору з правильними категоріями. Модель вивчає асоціації між різними лінгвістичними ознаками в тексті та категоріями.

Вибір моделі класифікації залежить від розміру корпусу та кількості слів у документах текстів. Зазвичай, вибір моделі повинен бути заснованим на відповідях на наступні питання: "Як представлені текстові дані алгоритму, який очікує на числовий вхід?", "Який тип моделі має бути використаним?", "Які параметри конфігурації мають бути використані для моделі?". На щастя, завдяки десятиліттям досліджень, на сьогоднішній день є доступ до великої кількості варіантів попередньої обробки даних і конфігурації моделі. Найкраще такі дані зведені у блок-схемі від компанії Google, яка допомагає звизити вибір моделей для експериментів базуючись на згаданих вище питаннях. Якщо відношення кількості документів до середньої кількості слів у документі менше за 1500, то рекомендується та є більш ефективним використання звичайних алгоритмів машинного навчання, таких як метод опорних векторів або градієнтний бустинг, та частотної векторизації текстів для підготовки даних, наприклад, TF-IDF. В іншому випадку більш ефективним буде використання моделі глибокого навчання, наприклад згорткової нейронної мережі та векторизації контекстним методом, наприклад за допомогою мережі Skim Gram.

### 3. ДОСЛІДЖЕННЯ ТА РЕЗУЛЬТАТИ

Для дослідження були зібрані публікації людей у спільнотах пов'язаних з психічним здоров'ям в соціальній мережі Reddit. Її особливістю, яка задовольняє потреби даного дослідження є те, що вона налічує безліч тематичних каналів (англ. subreddit) в яких люди діляться своїми думками, проблемами, почуттями згідно обраної теми. Головне припущення, яке було використано для збору даних можна сформулювати наступним чином: людина, яка хвора на певне психічне захворювання, пише пост у канал, в якому люди діляться як жити з цим психічним захворюванням. Якщо людина ділиться тим, що її також турбує ця проблема - всі її пости позначаються міткою цього захворювання. Всього було зібрано 740 345 публікації, всі англійською мовою.

Після збору даних кожен було виконано попередню обробку даних. Дописи поганих користувачів, тобто такі що є спамом, флудом або заблокованих користувачів - було видалено. Що стосується обробки тексту публікації - були видалені непотрібні знаки пунктуації та пробіли для кожного допису, символи приведені до нижнього регістру. Після цього був використаний набір інструментів природної мови (NLTK), реалізований на Python, щоб токенізувати дописи користувачів і фільтрувати часто вживані слова (стоп-слова). Для приведення токенізованих слів до їх кореневої форми було використано алгоритм Портера Стеммера, який застосовує послідовний ряд правил, що відсікає закінчення та суфікси та не містить бази слів, тому спрацьовує швидко, але не завжди безпомилково. Після цієї процедури для аналізу були використані дані 130 060 користувачів із 588 472 публікаціями. Фінальним етапом попередньої обробки стала розмітка даних для бінарної класифікації, тобто у відповідність кожному тексту була поставлена мітка 1 або 0 для кожного психічного розладу.

Набір даних було розбито на навчальну та тренувальну вибірку у відношенні 80% та 20% відповідно. Для вирішення задачі класифікації було порівняно два алгоритми:

класифікація за допомогою градієнтного бустингу та та класифікація за допомогою згорткових нейронних мереж (CNN).

Для класифікатору XGBoost було використано векторизатор на основі метрики TF-IDF для перетворення слів у n-вимірні вектори, тобто частотний метод. У випадку згорткової нейронної мережі було застосовано контекстний метод векторизації, а саме модель Skim Gram парадигми word2vec.

Рисунок 1 демонструє архітектуру нейронної мережі, що була розроблена під час дослідження. Архітектура моделі організована послідовністю шарів, що включає шар векторизації, декілька шарів згортки (Convolutional), шар максимального об'єднання (Max pooling), декілька щільних шарів (Dense), що представляють повнозв'язну нейронну мережу та вихідний шар з функцією активації сигмоїд для бінарної класифікації. Для навчання нейронної мережі було використано бінарну функцію втрат - перехресну кросенторпію та оптимізатор Адам. Гіперпараметр швидкості навчання було встановлено на рівні 0,001. Модель навчалася протягом 50 епох, а розмір батчу був встановлений на рівні 64 екземпляри.

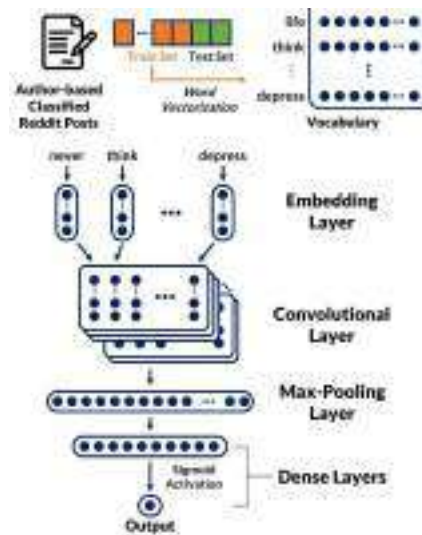


Рисунок 1. Архітектура згорткової нейронної мережі

Для перевірки ефективності моделей було використано чотири метрики оцінки: точність (accuracy), влучність (precision), повнота (recall) та міра F1 (F1-Score). TP, FN, TN та FP означають відповідно істинно-позитивний, хибно-негативний, істинно-негативний та хибно-позитивний результати.

Рисунки 2 та 3 підсумовують результати роботи восьми моделей бінарної класифікації. Серед восьми різних підкласів аутизм показав найвищу точність (96,96%) у згортковій нейронній мережі, але мав найнижчий показник F1 (XGBoost: 38,31%, CNN: 48,73%), що пов'язано з проблемою дисбалансу класів. Загалом, моделі CNN показали вищу точність, ніж моделі XGBoost, за всіма показниками. Для класу депресія були отримані найвищі показники точності (89,10%), відгуку (71,75%) та оцінки F1 (79,49%). Це пов'язано з тим, що він є найбільш збалансованим. Інші підкласи також показали високу точність з моделями CNN, а їхній показник F1 у визначенні психічних захворювань коливалися від сорока до п'ятдесяти (%), що є відносно нижчими, ніж у збалансованих за класами каналах. Таким чином, запропонована модель може точно виявляти потенційних користувачів, які можуть мати психічні розлади. Для покращення результатів необхідно провести додатковий збір даних та вирішити проблему незбалансованості даних. Це призведе до кращої продуктивності моделей.

Психічний розлад	Клас	XGBoost класифікатор	
		F1-Score	Точність (%)
депресія	1	58.02	71.69
	0	78.65	
тривожні розлади	1	55.92	70.41
	0	77.73	
біполярні розлади	1	53.59	85.53
	0	91.43	
посттравматичні розлади	1	46.43	85.14
	0	91.37	
шизофренія	1	40.97	86.72
	0	92.52	
аутизм	1	38.31	94.91
	0	97.35	
асоціальна поведінка	1	52.74	84.13
	0	90.34	
розлади харчової поведінки	1	34.25	92.44
	0	94.43	

Рисунок 2. Результати класифікації методом XgBoost

Психічний розлад	Клас	Згортова нейронна мережа CNN			
		Precision	Recall	F1-Score	Точність (%)
депресія	1	89.1	71.75	79.49	75.13
	0	58.66	82.04	68.41	
тривожні розлади	1	87.54	41.44	56.25	77.81
	0	75.92	96.91	85.14	
біполярні розлади	1	87.22	38.02	52.95	90.2
	0	90.4	99.05	94.53	
посттравматичні розлади	1	91.84	32.69	48.21	90.49
	0	90.42	99.54	94.76	
шизофренія	1	81.16	24.87	38.07	94.33
	0	94.62	99.56	97.03	
аутизм	1	48.08	49.39	48.73	96.96
	0	98.48	98.4	98.44	
асоціальна поведінка	1	86.17	36.52	53.25	90.1
	0	91.4	98.45	93.62	
розлади харчової поведінки	1	45.34	47.43	47.38	95.32
	0	97.32	96.65	96.23	

Рисунок 3. Результат класифікації за допомогою згорткових нейронних мереж.

## 4. ВИСНОВКИ

Методи обробки природної мови в останні роки отримали великий поштовх для розвитку та на сьогоднішній день, їх використання призвело до приголомшливих досягнень у різних сферах, від медицини до науки та розваг.

Однією з найбільш актуальних задач обробки природної мови є класифікації тексту. Основними етапами для її розв'язку є попередня обробка текстових даних, векторизація текстових даних та вирішення класичної задачі класифікації.

У ході дослідження було розроблено та порівняно вісім моделей машинного навчання для бінарної класифікації дописів у соцмережах з метою ідентифікувати психічні захворювання. Виконано порівняння ефективності звичайних алгоритмів класифікації та згорткових нейронних мереж. Дійшли до висновку, що згорткові нейронні мережі краще розв'язують поставлену задачу. Проведено тренування гіперпараметрів згорткової моделі. Найкращі результати для кожного з розладів було обгорнуто програмним інтерфейсом з метою надати користувачу можливість використовувати модель на практиці.

### ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Institute of Health Metrics and Evaluation. Global Health Data Exchange (GHDx). URL: <https://vizhub.healthdata.org/gbd-results/> (дата звернення: 10.08.2022).
2. Amal Menzli. Tokenization in NLP: Types, Challenges, Examples, Tools. URL: <https://neptune.ai/blog/tokenization-in-nlp/> (дата звернення: 14.10.2022).
3. Chirag Goyal. Step by Step Guide to Master NLP – Word Embedding and Text Vectorization. URL: <https://www.analyticsvidhya.com/blog/2021/06/part-5-step-by-step-guide-to-master-nlp-text-vectorization-approaches/> (дата звернення: 10.09.2022).
4. Tomas Mikolov. Distributed Representations of Words and Phrases and their Compositionality. URL: <https://arxiv.org/pdf/1310.4546.pdf> (дата звернення: 20.10.2022).

# ГІБРИДНА НЕЙРОННА МЕРЕЖА ДЛЯ ВИЯВЛЕННЯ І ВІДСТЕЖЕННЯ ЛЮДИНИ У ВІДЕОПОТОЦІ

Шаповал Н.В.<sup>1</sup>, Русакова Л.О.<sup>1</sup>

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

<sup>1</sup>shovgun@gmail.com, <sup>2</sup>rusakova.larisa@iit.kpi.ua

**Аналіз відеоконтенту (Video content analysis, VCA) або відеоаналітика (VA) на сьогоднішній день використовується у охоронних системах, роздрібній торгівлі, автомобільній промисловості, технологіях розумного будинку тощо. До задач відеоаналітики відносяться виявлення і відстеження об'єктів. Метою дослідження є підвищення ефективності виявлення і відстеження людини у відеопослідовностях. Було використано емпіричні та теоретичні методи досліджень. В результаті розроблено гібридну нейронну мережу для виявлення і відстеження людини: удосконалено згорткову нейромережу типу YOLO.**

**Ключові слова: виявлення об'єктів, відстеження об'єктів, згорткова нейронна мережа, YOLO, HOG.**

## 1. ВСТУП

На сьогоднішній день системи відеоспостереження використовуються майже у всіх сферах людської діяльності. Системи відеоаналітики надають можливість автоматичного аналізу відео для виявлення та визначення часових і просторових подій [1]. До них відносяться системи розпізнавання і відстеження осіб, що застосовуються у автоматичному керуванні для виявлення пішоходів на дорогах, у магазинах для підрахунку кількості та аналізу поведінки клієнтів, на зупинках громадського транспорту і в аеропортах для аналізу пасажиропотоку, у офісних приміщеннях і системах розумного будинку тощо.

Програмні підходи до виявлення і відстеження людини досить різноманітні, створюються спеціальні застосунки і веб-додатки чи окремі програмні модулі. Останнім часом широкого поширення набули нейронні мережі. Але при такому підході можуть виникнути проблеми з виявленням дрібних чи неповних об'єктів, стійкістю до зашумлених чи погано освітлених зображень. Багато часу займає навчання мереж як і сам процес виявлення. Деякі мережі обробляють відео зі швидкістю менше 10 кадрів за секунду що не прийнятно для застосування у режимі реального часу. Тому виникає необхідність створення ефективного алгоритму виявлення і відстеження людини, який дозволить значно точніше і швидше обробляти відеокадри.

## 2. МЕТОДИ ВИЯВЛЕННЯ І ВІСТЕЖЕННЯ ЛЮДИНИ

Існує багато різних методів виявлення об'єктів на зображеннях. Відео є складнішим за зображення, оскільки воно має ще один вимір – час. Його можна представити у вигляді серії (набору) зображень. І для його обробки достатньо прокрутити всі кадри у відеофайлі, застосувати відповідні методи розпізнавання об'єктів для кожного із кадрів. Але оскільки застосування одного і того самого алгоритму кожного разу не є ефективним з точки зору обчислень і не гарантує, що певний виділений об'єкт буде знайдено на кожному з кадрів (не

всі детектори є стійкими до різних поз та ракурсів), додатково використовуються алгоритми відстеження об'єктів.

Значного успіху в задачі виявлення об'єктів досягнув метод Віоли-Джонса [2], але він навчається повільніше у порівнянні з іншими існуючими методами, оскільки використовує значну кількість ознак. Гістограма орієнтованих градієнтів (Histogram of Oriented Gradients, HOG) працює з локальними комірками [3]. Тому він інваріантний до геометричних і фотометричних перетворень об'єкта. Але все одно традиційні методи виявлення об'єктів краще підходять для задач, де набір тренувальних прикладів невеликий. А для роботи з відеопослідовностями краще застосовувати згорткові нейронні мережі.

Найважливішою особливістю нейронних мереж, яка дозволяє успішно використовувати їх у різноманітних задачах (кластеризація, класифікація, розпізнавання образів, прогнозування і т. д.), полягає у паралельній обробці інформації усіма ланками, що дозволяє значно прискорити процес обробки інформації. Інша, не менш важлива властивість, – здатність до навчання. Згорткові нейронні мережі ефективно застосовуються у задачі розпізнавання об'єктів у режимі реального часу завдяки можливості до узагальнення накопичених знань і врахуванню інформації про співвідношення частин зображення між собою.

Згорткові мережі, які застосовуються для виявлення об'єктів базуються на двох підходах – одно і двоступеневому виявленні об'єктів. Двоступеневі детектори спочатку визначають регіони (області), у яких можуть знаходитись об'єкти, а потім шукають їх саме у цих регіонах. Такі мережі мають високу точність розпізнавання. Двоступеневими детекторами є мережа R-CNN (Region-based Convolutional Neural Networks) та її похідні (Fast R-CNN, Faster R-CNN, Mask R-CNN, Mesh R-CNN). А одноступеневі детектори, що одночасно виконують локалізацію та класифікацію об'єктів у всіх частинах зображення за один прохід, демонструють високу швидкість передбачення (висновку). Серед прикладів можна навести YOLO, SSD і RetinaNet, а також CenterNet, CornerNet та FCOS, що не використовують прив'язки.

В рамках даного дослідження було проведено порівняльний аналіз мереж Faster R-CNN, YOLO, SSD, RetinaNet з метою оцінки точності й швидкості виявлення об'єктів. Для цього використано датасет Human-Aligned Bounding Boxes from Overhead Fisheye cameras (НАВВОВ) [4]. Розроблений в лабораторії обробки візуальної інформації (Visual Information Processing, VIP) Бостонського університету та опублікований у вересні 2019 року. Набір даних містить 4 відеозаписи, записані камерами типу «риб'яче око» (fisheye cameras) у двох різних кімнатах (комп'ютерна лабораторія та невеликий конференц-зал) та відповідні анотації загальною кількістю 5837 кадрів.

Таблиця 1. Порівняння мереж на датасеті НАВВОВ

Мережа	Основа	mAP, %	recall, %	fps
Faster R-CNN	ResNet50	51,3	31,7	9
YOLOv3	DarkNet-53	37,5	26,4	39
SSD300	VGG16	39,7	27,6	36
RetinaNet	ResNet50	41,2	33,1	28

В цілому, мережа Faster R CNN досить точно виявляє об'єкти (51,3% на НАВВОВ), але її не варто застосовувати для роботи з відео (лише 9 кадрів за секунду). SSD і RetinaNet можуть бути застосовані тоді, коли важливою є точність виявлення об'єктів. Хоча вони поступаються швидкістю YOLO. SSD при цьому виявляє об'єкти у різних масштабах, а RetinaNet володіє високим значенням повноти (33,1%), що вказує на правильність класифікації об'єктів. YOLO працює значно швидше (39 кадрів за секунду) і підходить для

роботи в режимі реального часу, але показує відносно невелику точність (37,5%). Виявляє більше помилок локалізації, ніж Faster R CNN, та має труднощі з виявленням дрібних та об'єднаних у групи об'єктів. Тому її було обрано для подальшого удосконалення.

### 3. РОЗРОБЛЕННЯ ГІБРИДНОЇ МЕРЕЖІ

Запропонована у ході дослідження мережа складається з мережі YOLOv3 як базової для вирішення підзадачі виявлення об'єктів. Оскільки побудова такої моделі «з нуля» вимагає багато обчислювальних ресурсів, було вирішено завантажити модель, попередньо натреновану на датасеті COCO [5].

Структуру мережі подано на рисунку 1.



Рисунок 1. Структура гібридної мережі

Дескриптор HOG витягує ознаки з вхідного зображення і подає їх на вхід мережі YOLO для виявлення об'єктів. Далі за інформацією про місцезположення центральної точки об'єкта на кожному з кадрів реалізується відстеження цього об'єкта. Для цього у програмному коді створено окремий клас PntTracking, який знаходить послідовні пари найближчих сусідів, використовуючи двостороннє зіставлення дескрипторів точок.

Навчання мережі тривало 1 годину, що зумовлено комбінованою архітектурою мережі. Готова мережа досягла точності 39,2 % при 43 кадрах за секунду.

Таким чином було встановлено, що використання дескриптора HOG для даного набору даних дало вираш у точності на 1,7% і на чотири кадри за секунду швидшу роботу. Проте навчання зайняло на 20 хвилин більше часу.

### 4. ОЦІНКА ЕФЕКТИВНОСТІ СТВОРЕНОЇ МЕРЕЖІ

Надалі проведено кілька експериментів із запропонованою нейронною мережею з метою оцінки ефективності її роботи в різних умовах. Спершу на вхід мережі було подано відео із конференц залу в офісному приміщенні. Кадр з результатами виявлення і відстеження представлено на рисунку 2.



Рисунок 2. Результати роботи мережі

Як і було очікувано, мережа досить точно виявила і відстежила присутніх у відеофрагменті людей.

Другим випробуванням стала робота мережі в умовах поганого освітлення. На вхід подано відеофрагмент із пішоходами, знятий у вечірній час. Кадр результуючого відео наведено на рисунку 3.



Рисунок 3. Результати роботи мережі в умовах поганого освітлення

Мережа досить точно впоралася із завданням. Змогла навіть виявити осіб, що знаходилися подалі від камери і майже зливалися із заднім фоном.

Насамкінець на вхід мережі подано відео з камер спостереження магазину. Наявна велика кількість дрібних об'єктів. Частина людей перекривається. Приклад кадру результату мережі представлено на рисунку 4.



Рисунок 4. Результати роботи мережі

Мережа впоралася відносно добре навіть при перекритті об'єктів. Проте проблема виявлення дрібних об'єктів все ж залишається не вирішеною до кінця.

## 5. ВИСНОВОК

Таким чином, у даній роботі було досліджено використання згорткових нейронних мереж у задачі виявлення та відстеження людини в режимі реального часу.

Отримані результати наступні. Були розглянуті методи виявлення і відстеження об'єктів на зображеннях, зокрема: алгоритм Віоли-Джонса, гістограми орієнтованих градієнтів, нейронні мережі, відстеження на основі виявлення. Проведено порівняльний аналіз швидкості й точності нейронних мереж для виявлення об'єктів Faster R-CNN, YOLO, SSD та RetinaNet.

Найбільшу збалансованість між швидкістю й точністю продемонструвала мережа YOLO. Тому у ході дослідження реалізовано гібридну нейронну мережу на її основі. Поєднання YOLO з дескриптором HOG дозволило підвищити точність й швидкість виявлення і відстеження людини у відеопотоці. До того ж мережа виявила гарні результати і при слабкому освітленні. Відкритим залишається питання виявлення дрібних об'єктів.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Introduction to Video Analytics. [Електронний ресурс]. URL: <https://www.eetimes.com/Introduction-to-video-analytics/> (дата звернення: 06.06.22).
2. Viola P., Jones M. Rapid object detection using a boosted cascade of simple features. In: Computer Vision and Pattern Recognition Conference (CVPR). 2001.
3. Dalal N., Triggs B. Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition Conference (CVPR). 2005.
4. Li S., Tezcan M. O., Ishwar P., Konrad J. Supervised people counting using an overhead fisheye camera. In: International Conference on Advanced Visual and Signal-Based Surveillance (AVSS). 2019.
5. Redmon J., Farhadi A. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767. 2018.

# ЗГОРТКОВА НЕЙРОННА МЕРЕЖА ДЛЯ ЗАДАЧІ КЛАСИФІКАЦІЇ ЗАХВОРЮВАННЯ ЛИСТЯ РОСЛИН

Шевченко В.В.<sup>1</sup>, Шаповал Н.В.<sup>2</sup>

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

<sup>1</sup>nvo63911@gmail.com, <sup>2</sup>shovgun@gmail.com

**Актуальність теми обумовлена розвитком сільськогосподарства. Кількість нових видів захворювання рослин збільшується, що негативно впливає на врожайність і як наслідок на економіку країни. Метою цієї роботи є класифікації зображень вірусних захворювань листків рослин за допомогою згорткової нейронної мережі. Результат дослідження можна використовувати для подальшого аналізу під час створення автоматизованих теплиць.**

**Ключові слова:** класифікація зображень, згорткова нейронна мережа

## 1. ВСТУП

Згорткові нейронні мережі (CNN, ConvNet) – це клас глибинних штучних нейронних мереж прямого поширення, який застосовується до аналізу зображень. Використання згортки дозволяє зменшити кількість інформації, що дозволяє краще обробляти зображення більш високої роздільної здатності та виділяти опорні ознаки зображення, а саме: ребра, контури, грані. На наступному рівні обробки з цих ознак можна розпізнати повторювані фрагменти текстур, які далі можуть скластися в фрагменти зображення.

В цій роботі реалізовано декілька моделей згорткових нейронних мереж для задачі класифікації захворювань листків рослин. Дана модель допоможе фахівцям точніше аналізувати ситуацію із захворювання та приймати більш підходящий сценарій лікування.

## 2. ЗГОРТКОВА НЕЙРОННА МЕРЕЖА

Набір даних було зібрано організацією AIR Lab Makerere University. Дана вибірка[0] містить зображення двох типів захворювання листків та здорові листки рослин. Кількість зображень – 1034.

Для створення моделі була проведена попередня обробка зображень, стандартизація.

Першим дослідженням було створення власної шестишарової згорткової мережі. Точність розпізнавання зображень – 70%.

Тому для покращення результатів було використано мережу VGG16[0], який вважається однією з найкращих моделей комп'ютерного зору на сьогоднішній день. Дана модель використовує архітектуру з дуже малими (3 на 3) згортковими фільтрами. Глибина становить до 16-19 вагових шарів, що значно покращує результат. Щоб отримати оптимальний набір параметрів, необхідно мати багато даних та велику обчислювальну потужність.

Fine-tuning [0] дозволяє уникнути цих обмежень. Завдання тонкого налаштування мережі полягає в налаштуванні параметрів вже навченої мережі, щоб вона адаптувалась до нового завдання. Початкові рівні вивчають дуже загальні особливості, кожен наступний рівень має тенденцію вивчати моделі, більш специфічні для завдання, якому вони



# РОЗРОБКА МОДЕЛЕЙ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ АНАЛІЗУ ТА ФОРМУВАННЯ ІНВЕСТИЦІЙНОГО ПОРТФЕЛЮ

Шевчук О.С.<sup>1</sup>, Кузнєцова Н.В.<sup>2</sup>

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

<sup>1</sup>oleksii.shevchuk@ukr.net, <sup>2</sup>natalia-kpi@ukr.net [0000-0002-1662-1974]

У сучасному світі важливим є навички правильного фінансового управління, ціллю якого є приведення активів у рух з метою їх збереження та примноження. Ключовим елементом фінансового менеджменту є інвестування, зокрема у формі інвестиційного портфелю. Для ефективного формування та управління інвестиційного портфелю було розроблено велика кількість математичних методів та моделей, зокрема модель Марковіца. Останнім часом все більшої популярності набувають моделі штучного інтелекту. Їх застосування розповсюджується на різноманітні сфери, фінансова не стала виключенням. У даній роботі будуть розглянуті класичні методи формування інвестиційного портфелю, розроблено нову модель з використанням штучного інтелекту та проведено їх порівняння на конкретних моделях.

**Ключові слова:** інвестиційний аналіз, формування інвестиційного портфелю, методи штучного інтелекту.

## 1. ВСТУП

Поява поняття інвестування бере свій початок ще з часів стародавнього Вавилону. Саме тими часами датуються перші писемні згадки, про інвестування та грамотне управління своїми фінансами. З плином часу розроблялися нові способи інвестування, які підвищували прибутки та зменшували ризики. Одним із таких інструментів ефективного інвестування є інвестиційний портфель.

Інвестиційний портфель – це цілеспрямовано сформована сукупність вкладень в інвестиційні об'єкти, яка відповідає певній інвестиційній стратегії інвестора. Це означає, що основною ціллю формування інвестиційного портфелю є забезпечення реалізації розробленої інвестиційної політики шляхом підбору найбільш ефективних та надійних інвестицій [1].

Зважаючи на це, ключовим аспектом у формування інвестиційного портфелю є вибір оптимальних інвестиційних об'єктів, що обумовлює доцільність вивчення способів їх аналізу та моделювання їх поведінки. При цьому, важливим є врахування усіх особливостей фінансової системи, у якій знаходиться інвестиційний об'єкт, та чинників, що впливають на неї. Для підвищення ефективності аналізу, використовують математичні моделі, що описують фінансові процеси, а також стає все більш популярним використання моделей зі штучним інтелектом для вирішення інвестування.

## 2. ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

Метою даної роботи є розробка методу формування інвестиційного портфелю з використанням штучного інтелекту. Крім цього будуть розглянуті й існуючі методи, та

проведене їх порівняння між собою, та із методом штучного інтелекту. Результатом роботи буде розробка методу штучного інтелекту для формування інвестиційного портфелю, а також конкретних моделей, на основі математичних методів, та методів зі штучним інтелектом. Таким чином, об'єктом дослідження даної роботи є інвестиційні портфелі, а також способи їх аналізу та формування. Предметом дослідження є математичні методи та методи штучного інтелекту, що допомагають у формуванні та аналізі інвестиційних портфелів.

### **3. ОГЛЯД КЛАСИЧНИХ МЕТОДІВ ФОРМУВАННЯ ІНВЕСТИЦІЙНОГО ПОРТФЕЛЮ**

Формування інвестиційного портфелю можна розглядати як задачу вибору декількох оптимальних альтернатив серед множини усіх можливих інвестиційних об'єктів. Саме у таких випадках може допомогти метод прийняття рішень при нечіткому відношенні переваги на множині альтернатив. Даний метод базується на теорії Люфті Заде про нечіткі множини та їх властивості, яка була опублікована ще у 1965 році. Дана теорія допомагає формально описувати нечіткі поняття, якими користується людина, описуючи свої бажання, цілі та уявлення про систему [2].

Суть методу прийняття рішень полягає у порівнянні альтернатив за їх критеріями, і вибору оптимального або оптимальних із усієї множини. Зокрема цінність кожного з критеріїв встановлюється відповідно до цілей та вподобань. У випадку формування інвестиційного портфелю, альтернативами є об'єкти інвестування, а критеріями можуть бути різноманітні метрики, наприклад, для акцій компанії це може бути капіталізація компанії, її борги, вид діяльності компанії, ціна самої акції та інше. Таким чином, метод прийняття рішень допомагає інвестору обгрунтовано обрати об'єкти інвестування, зважаючи на свої цілі та вподобання.

Найбільш розповсюдженим і, певною мірою, класичним способом формування інвестиційного портфелю є спосіб із використанням моделі Марковіца. Теорія формування інвестиційного портфелю за Г. Марковіцем засновується на поведінковій специфіці інвестора, який хоче вкласти свої ресурси в компанії із найменшим ризиком, і за цей ризик отримати певні дивіденди. Підхід Марковіца передбачає, що інвестор враховує лише два параметри: ризикованість та величину доходів [3]. Модель оптимального портфелю за Г. Марковіцем ґрунтується на наступних принципах:

- інвестор хоче максимізувати дохід за певного заданого рівня ризику;
- інвестори завжди намагаються уникати ризику. Із двох активів із однаковою прибутковістю обирається той, ризикованість якого нижча;
- ризик є невизначеністю майбутнього результату;
- портфель інвестора складається з всіх його активів та зобов'язань;
- інвестори приймають рішення про інвестиції, засновуючись на очікуваних доходах та ризикованості інвестицій.

Таким чином, основна ідея моделі Марковіца полягає у тому, щоб статистично визначити майбутній дохід та ризик кожного інвестиційного об'єкта. Вибір об'єктів портфелю та їх частини може варіюватися від стратегії інвестора. Можна мінімізувати ризики портфеля або максимізувати його прибутковість. З розвитком теорії Марковіца з'явився інший спосіб вибору об'єктів інвестування – максимізація коефіцієнта Шарпа. Даний коефіцієнт допомагає вирішити проблему вибору між двома об'єктами інвестування, шляхом порівняння їх дохідності відносно безризикового об'єкту інвестування. Коефіцієнт Шарпа показує відносну прибутковість об'єкта на одиницю ризику [4]. Таким чином,

максимізація цього значення допомагає сформувати портфель із максимальною «прибутковістю» ризику.

Ще один метод, який допоможе визначити ймовірність певної події за наявними вхідними даними є мережі Байєса. Мережі Байєса допомагають встановити причинно-наслідкові зв'язки між певними ознаками та висновком, що отримується при таких умовах. У випадку задачі формування інвестиційного портфелю, на основі різноманітних вхідних параметрів, таких як зміна ціни акції, зміна капіталу, виду діяльності компанії, можна встановити, що чекає на ціну акції в майбутньому – падіння чи зростання.

Мережу Байєса можна розглядати як модель представлення взаємозв'язків між вершинами ациклічного графа, що представлені у вигляді ймовірнісних залежностей, тобто зв'язки між вершинами графа є причинними. Зв'язок буде називатися причинним лише тоді, коли подія є причиною виникнення події, тобто подія певною мірою впливає на виникнення події [5]. Теоретичним підґрунтям для мереж Байєса є теорія ймовірностей і, зокрема, теорема Байєса яку можна описати формулою \_\_\_\_\_.

Ця формула дає змогу визначити ймовірність залежної події, базуючись на наявній статистичній інформації.

Таким чином, метод мереж Байєса, дає можливість побудувати модель, яка буде визначати ймовірність кожного результату подій, із множини всіх результатів, базуючись на отриманих знаннях.

#### **4. ОГЛЯД МЕТОДУ ШТУЧНОГО ІНТЕЛЕКТУ ФОРМУВАННЯ ІНВЕСТИЦІЙНОГО ПОРТФЕЛЮ**

Метод з використанням штучного інтелекту є досить універсальним щодо даних. На вхід йому подаються різноманітні дані про акції компанії та саму компанію, як місячні, так і поквартальні. З цих даних збираються статистичні метрики для кожного параметру, такі як максимальне та мінімальне значення, середнє значення на проміжку, стандартне відхилення, а також ці самі ж метрики для першої різниці та інші параметри. Також використовується базова інформація про компанії, така як сектор та індустрія в якій працює компанія.

Сам метод базується на моделях градієнтних бустингів для визначення «справжньої» ціни компанії. Метод вивчає всі компанії, що відносяться до певних сектору та індустрії, їх метрики та ціни на акції, і встановлює де ціна завищена через певні чинники, а де навпаки – акції недооцінені. Сама суть градієнтних бустингів полягає у пошуку функції, яка описує набір вхідних та відповідних цільових даних. Навчання такої моделі – процес ітеративний, ідеєю якого є знайти таке рівняння, для якого деяка функція похибки на всьому наборі тренувальних даних буде оптимальним [6].

Бустингові алгоритми мають можливість формувати ансамблі – коли декілька моделей об'єднуються в єдине ціле. Такий підхід часто використовується для того, щоб підсилити позитивні якості окремих алгоритмів, які самі по собі можуть проявлятися слабо, а в групі показують чудовий результат. При використанні ансамблевих методів, алгоритми навчаються одночасно і можуть виправляти помилки один одного. Ансамблі моделей зазвичай будуються на основі моделі дерева рішень. Дерева додаються по одному в ансамбль і навчаються для взаємного виправлення помилок прогнозування, що роблять попередні моделі. Таким чином кінцевий результат покращується.

#### **5. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ**

Для дослідження використовувалися різноформатні дані про різноманітні компанії їх ціни на акції, зокрема базова інформація про компанію, їх поквартальні звіти, а також денні

дані про компанію та акції. Цільовим значенням, було значення курсу закриття ціни на акції, а прогноз робився на наступний квартал. Портфель формувався не з усіх можливих акцій компаній, а лише із компаній, що відносяться до п'яти обраних індустрій. Таким чином, портфель формувався із акцій наступних компаній: NKLA, TSLA, RACE, STLA, F, TM, MSFT, ADBE, ORCL, PAYO, BB, DBX, AAPL, NVDA, AMD, SONY, KOSS, BOX, NFLX, DIS, IMAX, WWE, CNK, WMG. Першою побудованою моделлю була модель для методу прийняття рішень при нечіткому відношенні переваги на множині альтернатив. Портфель, сформований цією моделлю, складається із семи об'єктів, таке значення було обране інвестором. Результати (Табл. 1) даного портфелю незадовільні, зокрема через наявність акцій TSLA, які дуже сильно впали за останній час, а саме їх в портфелі найбільший відсоток. Сумарний прибуток портфелю складає -1291,63.

Таблиця 1. Інвестиційний портфель моделлю прийняття рішень

Компанія	Різниця ціни закриття	Відсоток	Прибуток
TSLA	-64.24	21.74	-1396.52
NKLA	0.29	16.30	4.73
WWE	12.68	15.22	192.96
BOX	-1	13.04	-13.04
STLA	-4.77	11.96	-57.03
BB	-0.76	10.87	-8.26
AMD	-1.33	10.87	-14.46

Наступною побудованою моделлю була модель, що базується на теорії Марковіца. Вона показала значно кращі результати (Табл. 2, Табл. 3) ніж попередня і її можна назвати ефективною. Особливо це стосується портфеля, що збирався зо максимальним коефіцієнтом Шарпа, адже його прибутковість становить 96.75 у порівнянні із -105.67 для портфелю мінімального ризику.

Таблиця 2. Інвестиційний портфель моделлю Марковіца з мінімальним ризиком (найбільш значущі об'єкти)

Компанія	Різниця ціни закриття	Відсоток	Прибуток
TSLA	-64.24	1.4	-89.94
STLA	-4.77	2.3	-10.97
TM	-8.15	33.3	-271.40
MSFT	6.19	2.8	17.33
ADBE	-87.1	1.4	-121.94
ORCL	8.36	16.9	141.28
NVDA	-37.17	0.9	-33.45
DIS	7.22	2.6	18.77
WWE	12.68	16.4	207.95

Таблиця 3. Інвестиційний портфель моделлю Марковіца з максимальним коефіцієнтом Шарпа (найбільш значущі об'єкти)

Компанія	Різниця ціни закриття	Відсоток	Прибуток
TSLA	-64.24	3.9	-250.54
TM	-8.15	2.7	-22.01
ADBE	-87.1	2.1	-182.91
ORCL	8.36	1.3	10.87
NVDA	-37.17	2.7	-100.36
WWE	12.68	48.9	620.05

Модель мереж Байєса показала невдалі результати (Табл. 4), як і модель прийняття рішень. Прибуток портфелю, створеного цією моделлю, складає -740.65.

Таблиця 4. Інвестиційний портфель моделлю прийняття рішень

Компанія	Різниця ціни закриття	Відсоток	Прибуток
MSFT	6.19	11.41	70.63
ADBE	-87.1	11.41	-993.42
ORCL	8.36	11.41	95.38
PAYO	2.08	11.41	23.72
BB	-0.76	11.41	-8.67
DBX	2.41	11.41	27.49
SONY	2.76	20.15	55.62
BOX	-1	11.41	-11.41

Найбільш ефективною моделлю стала модель побудована із використанням штучного інтелекту. Сумарний дохід створеного портфелю складає 195.93. Модель виключила акціях, які впали найбільше за останній час, а саме ADBE та TSLA, і додала до портфелю найбільш прибуткові об'єкти з непоганим відсотком. Складові портфелю та їх відсотки в портфелі представлені у таблиці 5.

## 5. ВИСНОВКИ

Задача фінансового менеджменту завжди була актуальною. Особливо це важливо в кризові часи, коли ситуація в світовій економіці дуже нестабільні і більшість фінансових активів зазнають падіння в ціні.

В результаті даного дослідження було розглянуто класичні методи формування інвестиційного портфелю та побудовано відповідні моделі. Також, було розроблено новий метод формування, що використовує штучний інтелект. Було проведено порівняння створених моделей та визначено їх ефективність.

Таблиця 5. Інвестиційний портфель моделлю зі штучним інтелектом

Компанія	Різниця ціни закриття	Відсоток	Прибуток
NKLA	0.29	8.96	2.60
RACE	-0.71	6.08	-4.31
STLA	-4.77	10.72	-51.13
ORCL	8.36	9.50	79.43
PAYO	2.08	13.80	28.71
AMD	-1.33	9.81	-13.05
SONY	2.76	9.61	26.53
BOX	-1.00	8.62	-8.62
WWE	12.68	7.20	91.35
CNK	16.75	6.49	10.19
WMG	3.72	9.20	34.23

Найменш ефективною була модель прийняття рішень. Дійсно, даний метод допомагає лише обрати найбільш оптимальні альтернативи серед всіх можливих, але він ніяк не аналізує фінансові процеси. Також погано себе показав метод мереж Байєса. Йому не вдалося не те що отримати прибуто, але хоча б не зазнати збитків. Щодо методу Марковіца, то він показав відносно непогані результати. При виборі стратегії мінімального ризику, збитки будуть значно меншими, ніж для попередніх двох методів. Для стратегії максимізації коефіцієнта Шарпа був отриманий позитивний результат із прибутковим портфелем. Найбільш ефективним виявився метод із штучним інтелектом. Портфель, сформований даною моделлю, був найбільш прибутковим і він виключи акції з найбільшим падінням курсу.

### ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Замковий О.І. Портфельні теорії інвестування. метод.-наук. посіб. Дніпро: НТУ «ДП», 2020. – 70 с.
2. Зайченко Ю.П. Портфельні теорії інвестування. метод.-наук. посіб. Дніпро: НТУ «ДП», 2020. – 70 с.
3. Скрипниченко М.В. Портфельные инвестиции: Учебное пособие. - СПб: Университет ИТМО, 2016 - 40с.
4. Шарп У., Инвестиции. / У. Шарп, Г. Александер, Дж. Бейли. – М.: ИНФРА-М, 1997. – 172 с.
5. Бідюк П.І. Коршевніюк Л.О. Проектування комп'ютерних інформаційних систем підтримки прийняття рішень: Навчальний посібник. — Київ: ННК „ІПСА” НТУУ „КПІ”, 2010. — 340 с.
6. Открытый курс машинного обучения. Тема 10. Градиентный бустинг. URL: <https://habr.com/ru/company/ods/blog/327250/> (дата звернення: 29.10.2022)
7. Galkina S. Optimal Portfolio Selection Using Machine Learning Techniques / International Journal of Open Information Technologies ISSN: 2307-8162 vol. 2, no. 6, 2014.

# MAKING DECISIONS IN THE SERVICE BUSINESS SPHERE USING ARTIFICIAL INTELLIGENCE

I.A. Shubenkova.<sup>1</sup>, M.S. Marenych<sup>2</sup>

National Technical University of Ukraine  
“Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv, Ukraine

<sup>2</sup> mariiamarenych@gmail.com

**Current business conditions, featuring an unstable economic situation and growing fierce competition, put higher requirements on the efficiency and quality of decision-making at all levels of enterprise management. The research aims to analyze the business processes of the service sector businesses and develop software to improve the efficiency of increasing sales. Based on the researched material, I created the decision support system (DSS) to predict sales of service sector businesses (coffee, restaurant). The main idea of product development is free of charge and mass use. In work were used both theoretical and empirical research methods.**

**Keywords: decision-making, service sector business, business process analysis, sales prediction.**

## 1. INTRODUCTION

Today, making weighted and high-quality decisions is one of the main challenges for managers. Decision support provides up-to-date global information about the current situation and business development tendencies through data mining methods and tools. At the same time, the amount of information is constantly growing. It leads to a situation when it becomes impossible to effectively manage the company without using business analysis automation methods and tools [1].

The implementation of data mining in the practical work of enterprises to resolve specific issues of improving management effectiveness in the majority of cases is relatively expensive and time-consuming. The key challenges are the need for a formal statement of the problem and strategy of searching for information, the heuristic approach of most intelligent models, and high dimension and low-quality data. Therefore, developing new systems and implementing data mining methods to solve specific problems of increasing management efficiency in social and economic sectors is an urgent research and technical task.

## 2. PROBLEM STATEMENT

Based on the review of available scientific and technical sources, we can define the research goal: to develop a system for predicting demand for the assortment of products of the service sector business using artificial neural networks. For this, it is necessary to complete the following tasks:

- To analyze and propose a list of service sector business parameters those affect the value of demand and direct them to the neural network inputs;
- To justify the type and structure of a neural network for predicting the demand for products;
- To implement a system for predicting demand for products of the restaurant and coffee business in the service sector;
- To find relevant information in open sources and create a pool of training information based on which to train the neural network;

- To test the system and conclude the effectiveness of the developed product.

As inputs for predicting demand will be used:

- A list of names and economic efficiency values of restaurants and coffee shops that may affect the demand for the range of their products;
- A list of values of the selected values and the relevant value of the upcoming period demand, which makes up the dataset for training the neural network;
- Parameters of available system software development tools that enable to choice of the best option for the implementation of a software based on neural networks.

The output information will be the value of demand for specific products for a certain time period.

### 3. PREDICTION METHODS FOR DEMAND FOR PRODUCT SALES

Ideally, demand prediction should include both qualitative and quantity prediction. Both of them require collecting information from different data sources. Qualitative data can be extracted from external sources such as news reports, cultural and social media trends, and competitor and market research. Information from internal sources, such as customer feedback and preferences, also significantly increases the accuracy of the prediction. Quantitative information is usually internal and can be collected from sales figures, periods of peak purchases, and web and search analytics. Current technologies use advanced analytics, powerful databases, artificial intelligence, and machine learning to analyze and process highly complex data sets. When modern technology is applied to qualitative and quantitative prediction and predictive analytics, supply chain managers can provide ever-increasing levels of accuracy and resilience.

According to the industry, customer database, and product volatility, demand planners use the following prediction methods [2]:

1. Predicting demand at the macro level;
2. Predicting demand at the micro level;
3. Short-term demand prediction;
4. Long-term demand prediction.

To understand it better, we provide a relative analysis of prediction methods among themselves (Table 1).

Table 1. Comparative analysis of prediction methods

Comparison parameters	Timeline analysis method	Casual method	Expert assessment methods
Term	Short-term, mid-term	medium-term, long-term	long-term
Requirements for implementing the method	Quantitative parameters	Quantitative parameters	Experts and qualified specialists
Speed	+	+	+
Complexity of method	Medium	High	High
Credibility and reliability	Depends on the quality and quantity of data	Depends on the prediction	Depends on the expert's competence
Advantages	Simplicity, clarity of results, easy implementation		Systematic approach
Disadvantages	Only quantitative predictions, availability of data for a long-term perspective	Practical implementation of the method for some areas may be incorrect	The credibility and reliability of the research results depend on the expert's competence

After a detailed analysis of each method and considering the specifics of the service sector business, this method is used when predicting for the short and medium term period, allowing to obtain sufficient accuracy of predicted values. Also, this method is not only the most popular method of all listed but also rather simple and effective. It is easy enough to implement using a computer. Using this method will allow us to consider the effect of seasonality and tendency factors to identify the months of the highest and lowest activity in sales and, therefore, to plan stocks correctly. Based on the previous analysis of time series reflecting the dynamics of sales of products of the service sector business, linear and polynomial functions were chosen as the trend model. These are the most simple and commonly used functions in sales prediction. The terms of these functions can be easily calculated using the least squares method. Let's create a mathematical model:

Let  $x = \{x_1, x_2, x_3 \dots x_n\}$  – a set of parameters applied to the inputs of the neuron, where  $n$  is the number of input parameters.

$y_{true} = \{y_1, y_2, y_3 \dots y_m\}$  – a training set of valid values of demand used in the neural network, where  $m$  is the number of output parameters;

$\omega_1, \omega_2, \omega_3 \dots \omega_n$  – a set of synaptic weights

$$S = \sum_{i=1}^n \omega_i \cdot X_i,$$

Where  $S$  is an adder designed to produce a weighted sum.

$$y_{pred} = f(S) = F\left(\sum_{i=1}^n \omega_i \cdot X_i\right),$$

Where  $y_{pred}$  is the predicted demand value for the product,  $f$  is the sigmoid activation function. The target function will be the following formula:

$$L = \frac{1}{2} \sum_{i=1}^m (y_{true} - y_{pred})^2 \rightarrow \min,$$

Where  $L$  is the loss function.

By using neural networks prediction demand for a range of things in a supermarket solve, it is necessary to train the neural network so that the prediction accuracy is as high as possible. To implement the algorithm of the training process, you must initially import the training data from the database. The file holds the parameters that affect the demand and the demand values for the selected products for a given time term using simulated sales for the selected products. Figure 1 illustrates the neural network training process for the service sector business. To start training, the user needs to select the necessary product for which the sales simulation for a specific time was performed and begin the neural network training process. After the training process completing and the weights are corrected, the program will write all the training data of the neural network to the database file. By using the training, the user will be able to predict the sales of products for a selected time. Neural network training is the loss function minimization problem. To solve this problem, we use the stochastic gradient descent algorithm. Using this algorithm, you can adjust the synapse weights of the network. The following steps carry out the implementation of the stochastic gradient descent algorithm:

1. Select one parameter from the data set.
2. Find all partial derivatives of the weight or displacement loss function.
3. Update all weights and displacements.
4. Repeat the cycle from step 1.

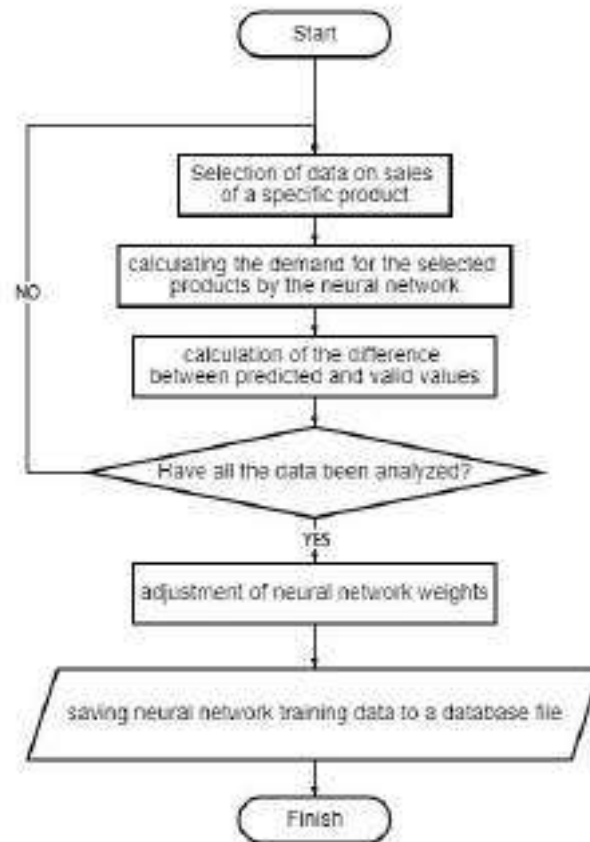


Figure 1. The algorithm of the neural network training process for service sector business

#### 4. SOLUTION

The developed product must comply with quality features such as sustainability, usefulness, accessibility, scalability, openness, flexibility, and testability. It requires additional constraints/rules from the development process, such as:

- Compliance with templates and styles;
- Documentation of development at various levels;
- Testing of components, specific modules, and subsystems;
- Project and process management.

By considering the requirements to ensure the sustainability and flexibility of the system, we chose the Layers pattern for its development, which divides the system into two parts: client and server. The system's design will rely on the subject area (DDD approach) and SOLID principles [3]. When developing the customer, we chose the Windows Forms user interface. The server, in turn, will consist of the following modules:

1. Business Logic Layer– logic and all necessary calculations of the application in the language of business;
2. Data Access Layer – the level of access to data.
3. Data Base – database for data storage.

By taking into account all the above templates, the project will look as follows (Figure 2):

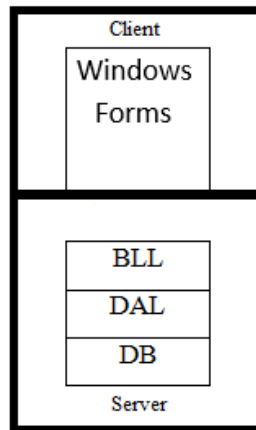


Figure 2. Project structure

Each component will implement a contract (interface) that provides flexibility to the element. Since the business is constantly changing relatively quickly, each part must soon adapt to these changes. For this, we will use the Dependency Injection pattern.

## 5. CONCLUSIONS

We have reviewed the main methods of prediction of economic processes and conducted their comparative analysis. We chose the development tools to implement the decision support system for the service sector business and justified the prediction method.

An algorithm for building a prediction and estimating we developed its error based on the chosen method. We implemented the developed algorithm in software and the information system of the service business. The program showed good forecasting accuracy results and fully met the requirements. In further research, it is advisable to introduce such elements that take into account at the probability level rare events that sometimes occur in the world (for example, pre-holidays, quarantine, etc.). It can radically change the real numbers compared to the forecasts of the best economic mathematical models.

## REFERENCES

1. "Business intelligence (BI): the definitive guide". MicroStrategy. <https://www.microstrategy.com/en/resources/introductory-guides/business-intelligence-the-definitive-guide>.
2. Project Analysis. Kyiv: Libra, 1999.
3. Karpychev, V. Yu, та Yu P. Shal'nova. "Big Data Conceptual Model". Proceedings of R.E. Alekseev NSTU, № 2 (2021): 16–23. [https://doi.org/10.46960/1816-210x\\_2021\\_2\\_16](https://doi.org/10.46960/1816-210x_2021_2_16).

# ВИКОРИСТАННЯ МОДЕЛІ BI-LSTM ДЛЯ ПІДВИЩЕННЯ ТОЧНОСТІ ПРОГНОЗУВАННЯ КУРСУ КРИПТОВАЛЮТ

Яцько Я.В.<sup>1</sup>, Терентьєв О.М.<sup>2</sup>

<sup>1</sup> Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

<sup>2</sup> Інститут телекомунікацій і глобального інформаційного простору  
Національної академії наук України, Київ, Україна

<sup>1</sup>yayatskaa@gmail.com, <sup>2</sup>o.terentiev@gmail.com

**Криптовалюта – цифрова валюта, яка використовується у фінансових системах і використовує технологію блокчейн і криптографічні функції для досягнення прозорості та децентралізації. Оскільки ціни на криптовалюту сильно коливаються, необхідні інструменти для їх моніторингу та прогнозування. Довга короткочасна пам'ять (LSTM) – це модель глибокого навчання, яка здатна точно прогнозувати часові ряди даних. У цьому дослідженні використовується двонаправлена LSTM. (Bi-LSTM) для підвищення точності і нормалізації оцінки середньоквадратичної помилки (RMSE) з використанням чотирьох криптовалют: Bitcoin, Ethereum, Ripple і Binance (BNB).**

**Ключові слова:** криптовалюта, Bi-LSTM, блокчейн, Bitcoin.

## 1. ВСТУП

На ринок криптовалют впливають фактори невизначеності, такі як політичні та економічні проблеми на глобальному рівні. Тому точна інтерпретація прогнозів є складним завданням. Іншою проблемою, яка є предметом цього дослідження, є щоденні коливання курсу криптовалют, які необхідно вирішити за допомогою прикладного інструменту, який може відстежувати та запобігати невизначеності в транзакціях [3]. На ринку криптовалют є багато популярних монет, таких як Bitcoin (BTC), Ethereum (ETH), Binance Coin (BNB) і Ripple (XRP).

Попереднє дослідження включало в себе використання алгоритму нейронної мережі з довгою короткочасною пам'яттю (LSTM). При цьому, значення RMSE мало значний розбіг (дисперсію) від 2 000 до 50 000 доларів. Тому ціллю експерименту було використання моделі Bi-LSTM для підвищення точності прогнозування.

## 2. ПОСТАНОВКА ЗАДАЧІ

Прогнозування курсу криптовалют має незначні відмінності від методу прогнозування акцій. Кращі результати можна отримати, поєднуючи різні методи, такі як аналіз даних часових рядів, технічний аналіз фондового ринку та історичні дані з ціни, з декількома алгоритмами [4]–[6]. Це дослідження робить певний внесок у область прогнозування криптовалют. Оцінкою якості моделі обрано RMSE та MAPE. У попередніх дослідженнях для прогнозування використовувалися лише алгоритми нейронної мережі з довгою короткочасною пам'яттю (LSTM). Для вдосконалення попередньої моделі запропоновано використати гібридну нейронну мережу, що складається з рекурентних блоків (GRU) і двонаправленої LSTM (Bi-LSTM), оскільки остання обробляє набори даних за допомогою багаторазового циклу, що збільшує передбачувану здатність моделі. В цьому дослідженні

було використано дані з Binance за останні 3 роки та використано модель Bi-LSTM, структура мережі якої зображена на рис.1:

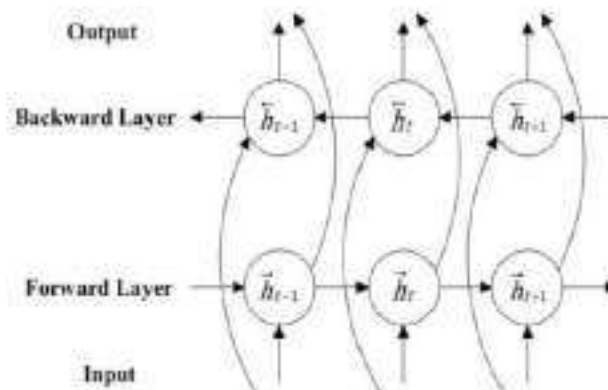


Рисунок 1. Структура мережі Bi-LSTM [7]

### 3. ОПИС ЗНАЙДЕНОГО РІШЕННЯ

Слід зазначити, що результати є кращими, ніж в попередньому експерименті з використанням простої моделі LSTM (RMSE менше на 20 % пунктів). Після нормалізації результати стають ближчими до 0, а оцінка MAPE становить менше 10% за RMSE [4], [6]. На рис. 2 зображено порівняльну тенденцію реальних даних та прогнозу за допомогою моделі експерименту.

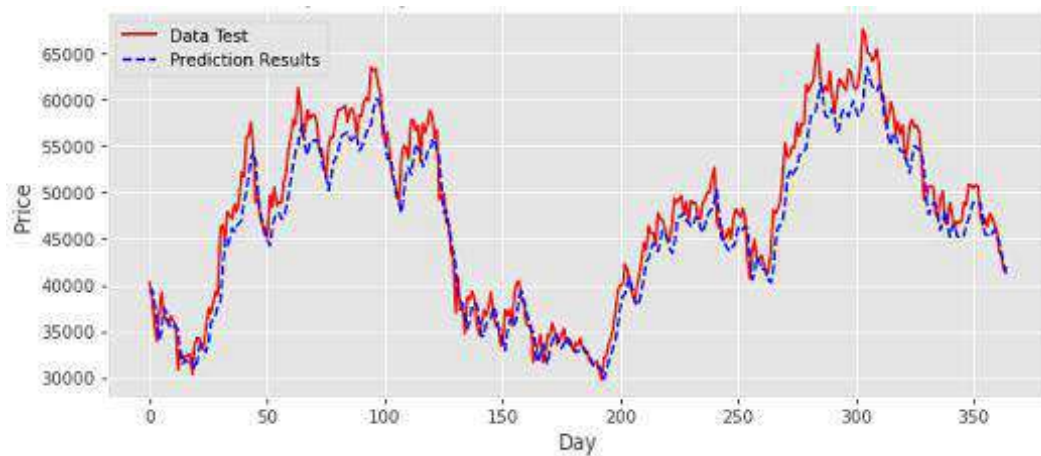


Рисунок 2. Порівняння фактичних даних і прогнозу курсу Bitcoin

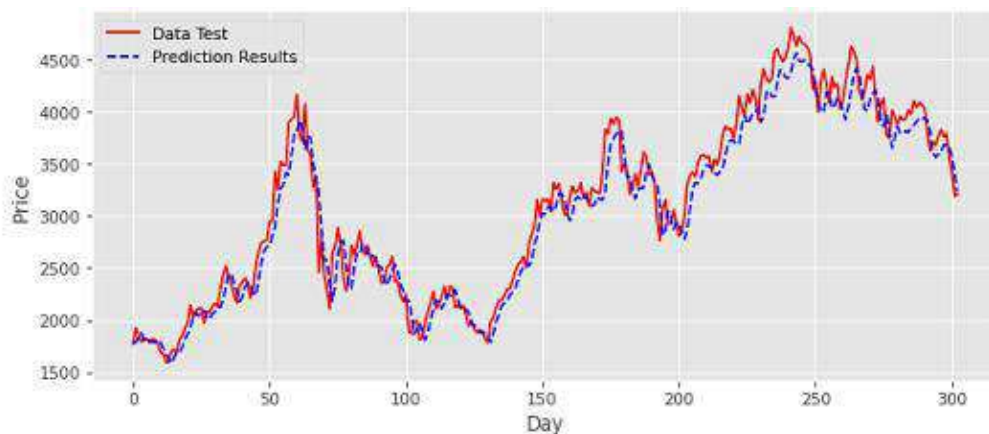


Рисунок 3. Порівняння фактичних даних і прогнозу курсу Ethereum

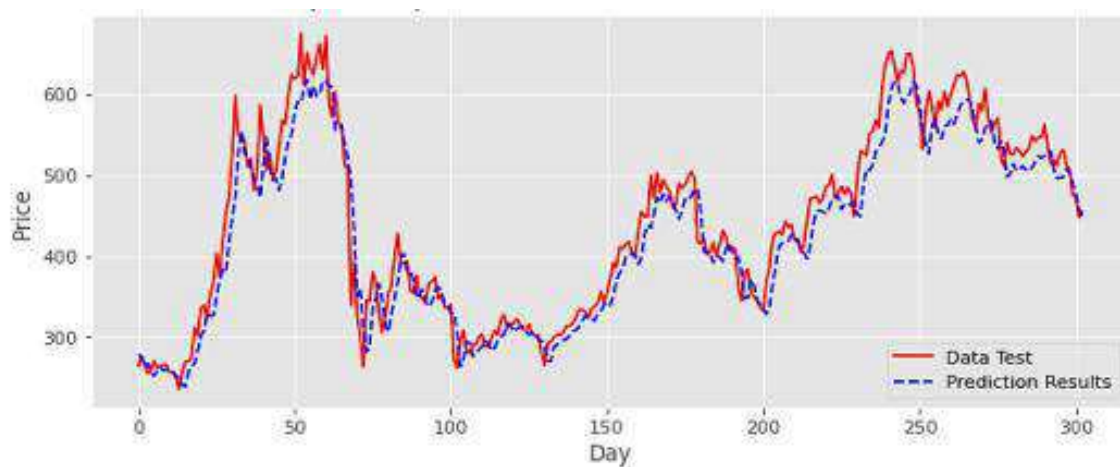


Рисунок 4. Порівняння фактичних даних і прогнозу курсу Binance

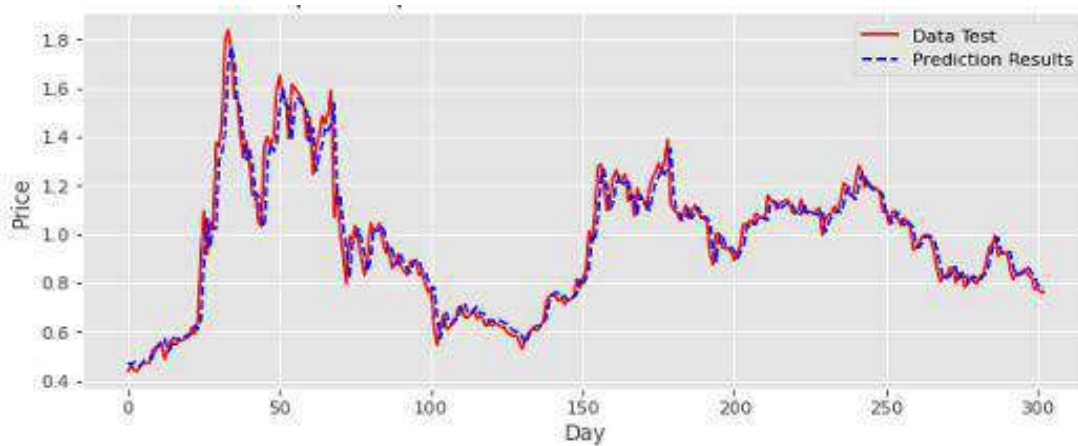


Рисунок 5. Порівняння фактичних даних і прогнозу курсу Ripple

В табл. 1 розміщено результати оцінки моделі на основі значень MAPE і RMSE.

Таблиця 1. Метрики якості моделі

Монета	RMSE	Normalize RMSE	MAPE (%)
Bitcoin	2343.2200	0.062	4.0
Ethereum	203.8900	0.063	5.31
Binance	404.1800	0.073	5.64
Ripple	0.0933	0.066	5.82

#### 4. ВИСНОВКИ

В ході дослідження було розглянуто гібридну модель GRU Bi-LSTM для покращення прогнозування курсу криптовалют. Запропонована модель дала значно кращі результати, у порівнянні зі звичайним LSTM. У дослідженні було застосовано кілька підходів для прогнозування курсу криптовалют з використанням історичних даних, отриманих за допомогою Binance API за 3 роки. Чотири монети (Bitcoin, Ripple, Binance та Ethereum) були використані для перевірки точності запропонованої моделі щодо забезпечення оптимальних результатів. Завданням для наступних досліджень є покращення точності прогнозування за рахунок додання в модель нових даних, таких як політичні настрої, природні умови і т.д. з зовнішніх джерел інформації.

#### ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. U. Mukhopadhyay, A. Skjellum, O. Hambolu, J. Oakley, L. Yu, and R. Brooks, "A brief survey of cryptocurrency systems," 2016 14th Annual Conference on Privacy, Security and Trust, PST 2016, pp. 745–752, 2016, doi: 10.1109/PST.2016.7906988.
2. E. Pintelas, I. E. Livieris, S. Stavroyiannis, T. Kotsilieris, and P. Pintelas, "Investigating the problem of cryptocurrency price prediction: a deep learning approach," IFIP Advances in Information and Communication Technology, vol. 584 IFIP, pp. 99–110, 2020, doi: 10.1007/978-3-030-49186-4\_9.
3. J. Eapen, D. Bein, and A. Verma, "Novel deep learning model with CNN and bi-directional LSTM for improved stock market index prediction," in 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC 2019, 2019, pp. 264–270, doi: 10.1109/CCWC.2019.8666592.
4. V. B. Kamble and S. N. Deshmukh, "Comparision between accuracy and MSE, RMSE by using proposed method with imputation technique," Oriental journal of computer science and technology, vol. 10, no. 04, pp. 773–779, 2017, doi: 10.13005/ojst/10.04.11.
5. M. De Caux, F. Bernardini, and J. Viterbo, "Short-term forecasting in Bitcoin time series using LSTM and GRU RNNs," in Anais do VIII Symposium on Knowledge Discovery, Mining and Learning, SBC, 2020, pp. 97–104.
6. T. O. Hodson, "Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not," Geoscientific Model Development, vol. 15, no. 14, pp. 5481–5487, Jul. 2022, doi: 10.5194/gmd-15-5481-2022.
7. Tavakoli, N. (2019). Modeling Genome Data Using Bidirectional LSTM. 2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC), 2, 183.