

<https://doi.org/10.31891/2219-9365-2026-86-26>

УДК 004.4:004.85

СИМЕЦЬ Іван

Національний університет «Львівська політехніка»

<https://orcid.org/0000-0003-1873-3168>

e-mail: [ivan.i.symets@lpnu.ua](mailto:ivan.i.symets@lpnu.ua)

## РОЗРОБКА ПРОГРАМНОЇ СИСТЕМИ ВІЯВЛЕННЯ АНОМАЛІЙ У ДАНИХ ІЗ ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ

У роботі наведено результати розробки програмної системи виявлення аномалій у даних із використанням нейронних мереж, орієнтованої на підвищення точності детекції, адаптивність до змін характеристик даних та забезпечення обробки інформації у режимі реального часу. Метою дослідження є розробка інтегрованої програмної системи виявлення аномалій, яка забезпечує ефективну обробку різнорідних даних, підвищену точність виявлення відхилень, адаптацію до змін у даних (concept drift) та можливість функціонування в умовах потокової обробки в реальному часі. Актуальність дослідження обумовлена стрімким зростанням обсягів різнорідних даних та необхідністю створення інтелектуальних систем, здатних своєчасно виявляти відхилення у складних інформаційних середовищах, зокрема у задачах кібербезпеки, аналізу мережевого трафіку, IoT та фінансових систем. У роботі проведено аналіз сучасних підходів до виявлення аномалій на основі нейронних мереж, включаючи автоенкодера, рекурентні, трансформерні та генеративні моделі, визначено їх переваги, обмеження та проблеми інтеграції у прикладні програмні системи. Запропоновано архітектуру програмної системи, що реалізує модульний підхід і включає підсистеми збору, попередньої обробки, навчання моделей, виявлення аномалій та адаптації моделей, при цьому основна увага зосереджена на інтеграції алгоритмічної та архітектурної складових системи. Особливістю системи є використання ансамблевого підходу для формування інтегральної оцінки аномальності, а також механізмів адаптації до змін у даних (concept drift) на основі аналізу розподілів. Використано алгоритми обробки поточкових даних із мінімізацією затримок та забезпеченням масштабованості. Проведено експериментальне дослідження на наборах даних різної природи, що дозволило оцінити універсальність та ефективність запропонованого підходу. Отримані результати підтверджують підвищення точності виявлення аномалій та стійкості системи до змін характеристик даних у порівнянні з використанням окремих моделей. Особливістю роботи є розробка інтегрованої архітектури програмної системи, що поєднує різні типи нейронних мереж, механізми адаптації та формування узагальненого показника аномальності. Практичне значення роботи полягає у можливості застосування розробленої системи у реальних інформаційних системах для моніторингу, аналізу та підтримки прийняття рішень.

Ключові слова: виявлення аномалій, нейронні мережі, автоенкодер, трансформери, ансамблеві методи, concept drift.

SYMETS Ivan

Lviv Polytechnic National University

## DEVELOPMENT OF A SOFTWARE SYSTEM FOR ANOMALY DETECTION IN DATA USING NEURAL NETWORKS

The paper presents the results of developing a software system for anomaly detection in data using neural networks, aimed at improving detection accuracy, adaptability to changes in data characteristics, and ensuring real-time data processing. The objective of the study is to develop an integrated software system for anomaly detection that provides efficient processing of heterogeneous data, increased accuracy of deviation detection, adaptation to changes in data (concept drift), and the ability to operate under real-time streaming conditions. The relevance of the research is determined by the rapid growth of heterogeneous data volumes and the need to create intelligent systems capable of timely detecting anomalies in complex information environments, particularly in cybersecurity, network traffic analysis, IoT, and financial systems. The paper analyzes modern approaches to anomaly detection based on neural networks, including autoencoders, recurrent, transformer-based, and generative models, identifying their advantages, limitations, and challenges of integration into applied software systems. An architecture of the software system is proposed, implementing a modular approach and including subsystems for data collection, preprocessing, model training, anomaly detection, and model adaptation, with primary emphasis placed on integrating the system's algorithmic and architectural components. A distinctive feature of the system is the use of an ensemble approach to form an integrated anomaly score, as well as mechanisms for adapting to data changes (concept drift) based on distribution analysis. Algorithms for streaming data processing with minimized latency and ensured scalability were applied. An experimental study was conducted on datasets of various types, which allowed evaluating the universality and effectiveness of the proposed approach. The obtained results confirm an increase in anomaly detection accuracy and system robustness to changes in data characteristics compared to the use of individual models. A distinctive feature of the work is the development of an integrated software system architecture that combines different types of neural networks, implements adaptation mechanisms, and forms a generalized anomaly indicator. The practical significance of the work lies in the possibility of applying the developed system in real information systems for monitoring, analysis, and decision support.

Keywords: anomaly detection, neural networks, autoencoder, transformers, ensemble methods, concept drift.

Стаття надійшла до редакції / Received 27.03.2026

Прийнята до друку / Accepted 22.04.2026

Опубліковано / Published 31.05.2026



This is an Open Access article distributed under the terms of the [Creative Commons CC-BY 4.0](https://creativecommons.org/licenses/by/4.0/)

© СИМЕЦЬ Іван

## ПОСТАНОВКА ПРОБЛЕМИ У ЗАГАЛЬНОМУ ВИГЛЯДІ ТА ЇЇ ЗВ'ЯЗОК ІЗ ВАЖЛИВИМИ НАУКОВИМИ ЧИ ПРАКТИЧНИМИ ЗАВДАННЯМИ

В сучасних умовах цифровізації та стрімкого зростання обсягів даних ключовими викликами розвитку інформаційних систем є забезпечення високої точності аналізу даних, своєчасне виявлення відхилень у їх поведінці та адаптивність до змін середовища функціонування. Особливої актуальності набувають задачі виявлення аномалій у даних, що виникають у таких сферах, як кібербезпека, фінансові системи, промисловий моніторинг, медичні дослідження та інтелектуальні транспортні системи. Виявлення аномалій дозволяє своєчасно ідентифікувати збої, несанкціоновані втручання, або критичні відхилення, що є важливим для забезпечення надійності та безпеки функціонування складних систем. На практиці широко застосовуються класичні статистичні та машинні методи аналізу даних, однак їх ефективність є обмеженою у випадку складних, високорозмірних та нелінійних даних. У зв'язку з цим значного поширення набули методи на основі нейронних мереж, зокрема автоенкодерів, рекурентні та трансформерні моделі, генеративно-змагальні мережі, що демонструють високу здатність до виявлення складних закономірностей у даних [4, 20]. Водночас використання таких моделей у реальних умовах супроводжується рядом суттєвих обмежень. Зокрема, сучасні підходи до виявлення аномалій здебільшого орієнтовані на розв'язання окремих задач, таких як аналіз часових рядів, або обробка зображень, і не забезпечують комплексного підходу до побудови програмної системи. Автоенкодерів, які є одним із найпоширеніших інструментів, демонструють високу ефективність у задачах реконструкції даних, однак можуть відтворювати аномальні зразки, що ускладнює їх ідентифікацію [3, 5]. Генеративні моделі забезпечують більш глибоке моделювання розподілу даних, проте характеризуються нестабільністю навчання та складністю налаштування [16, 17]. Методи, орієнтовані на часові ряди, потребують значних обчислювальних ресурсів та складних механізмів обробки поточкових даних у реальному часі [20]. Крім того, важливою проблемою є відсутність універсальних підходів до інтеграції різних моделей у єдину програмну систему. Наявні програмні засоби, зокрема бібліотеки для виявлення аномалій, забезпечують реалізацію окремих алгоритмів, але не вирішують задачі їх адаптації, масштабування та взаємодії в умовах реальних інформаційних систем [13]. Додатковою складністю є необхідність врахування змін у даних (concept drift), забезпечення інтерпретованості результатів та мінімізації затримок обробки. В результаті наявні підходи не повною мірою задовольняють вимоги до сучасних програмних систем, що повинні забезпечувати комплексне, адаптивне та масштабоване виявлення аномалій у різнорідних даних, що обумовлює необхідність розробки нових підходів до побудови програмних систем виявлення аномалій на основі нейронних мереж, які поєднують різні моделі, забезпечують їх ефективну інтеграцію, адаптацію до змін середовища та можливість використання в режимі реального часу.

## АНАЛІЗ ДОСЛІДЖЕНЬ ТА ПУБЛІКАЦІЙ

Аналіз сучасних досліджень у сфері виявлення аномалій у даних із використанням нейронних мереж свідчить про активний розвиток даного напрямку, що зумовлено зростанням обсягів даних та необхідністю автоматизації процесів контролю складних інформаційних систем. При цьому більшість досліджень концентрується на вдосконаленні окремих підходів, тоді як питання комплексної реалізації програмної системи залишається недостатньо розкритим. У роботах [4, 8] сформовано загальну класифікацію методів виявлення аномалій на основі нейронних мереж, що включає автоенкодерів, генеративно-змагальні мережі, рекурентні та згорткові архітектури. Основним досягненням цих досліджень є систематизація підходів та обґрунтування ефективності *unsupervised* і *semi-supervised* методів, що дозволяють працювати за відсутності розмічених даних. Разом з тим, у зазначених роботах не розглядаються питання побудови єдиної програмної системи, яка могла б інтегрувати різні моделі, автоматично обирати оптимальний алгоритм та адаптуватися до зміни характеристик даних. Складність вирішення цієї проблеми полягає у гетерогенності вхідних даних та відсутності універсальних критеріїв оцінювання якості виявлення аномалій, що ускладнює формалізацію процесу вибору моделі в межах програмної системи. Значна кількість досліджень присвячена автоенкодерам як базовому інструменту *anomaly detection* [3, 10, 11, 12]. У цих роботах вирішено задачу побудови моделей, здатних виявляти аномалії через оцінювання похибки реконструкції, а також запропоновано різні модифікації автоенкодерів для роботи з високорозмірними даними.

Водночас у роботі [5] доведено, що автоенкодерів можуть відтворювати аномальні дані з високою точністю, що призводить до втрати їх дискримінаційної здатності. Нерозкритими залишаються питання забезпечення надійності таких моделей, їх адаптації до *concept drift* та інтерпретації результатів. Складність вирішення цих задач пов'язана з природною здатністю нейронних мереж до узагальнення, що унеможливує чітке розмежування нормальних та аномальних зразків, а також із залежністю ефективності моделей від параметрів навчання. У контексті розробки програмної системи вище відмічене обумовлює необхідність використання комбінованих підходів та механізмів контролю якості моделей.

Окремий напрям досліджень стосується виявлення аномалій у часових рядах [20]. У цих роботах запропоновано використання рекурентних нейронних мереж, трансформерів та гібридних архітектур, що дозволяють враховувати часові залежності та підвищувати точність виявлення аномалій у багатовимірних даних. Проте залишаються нерозв'язаними проблеми обробки поточкових даних у режимі реального часу та

масштабованості таких рішень. Складність полягає у необхідності забезпечення низьких затримок обробки при високій обчислювальній складності моделей, а також у потребі реалізації механізмів онлайн-навчання. У рамках розробки програмної системи це вимагає створення ефективної архітектури обробки даних, здатної працювати в умовах реального часу.

Дослідження, присвячені генеративним моделям [16, 17], демонструють можливість виявлення складних аномалій шляхом моделювання розподілу нормальних даних. Основним досягненням є здатність таких моделей працювати з неструктурованими даними, зокрема зображеннями. Водночас нерозкритими залишаються питання стабільності навчання та інтерпретованості результатів. Складність вирішення цих проблем зумовлена нестійкістю процесу навчання GAN та залежністю якості результатів від генератора. У програмній системі це створює потребу у впровадженні механізмів моніторингу та контролю процесу навчання моделей.

Сучасні дослідження також спрямовані на використання self-supervised підходів та нових архітектур [4, 1, 7], що дозволяють зменшити залежність від розмічених даних і підвищити узагальнюючу здатність моделей. Разом з тим, ці підходи характеризуються високою обчислювальною складністю та складністю інтеграції у прикладні системи. Основна проблема полягає у необхідності використання великих обсягів даних та ресурсів для навчання, що ускладнює їх застосування у реальних програмних системах. Таким чином, виникає потреба у розробці модульних архітектур, які дозволяють ефективно інтегрувати такі моделі.

Перспективні напрями, пов'язані з diffusion-моделями та квантовими нейронними мережами [6, 8], відкривають нові можливості для підвищення точності виявлення аномалій, проте знаходяться на стадії активного дослідження. Нерозкритими залишаються питання їх практичної реалізації та інтеграції у програмні системи. Складність полягає у високих вимогах до обчислювальних ресурсів та відсутності зрілих інструментів розробки. Також увагу слід приділити програмним реалізаціям методів anomaly detection, зокрема бібліотеці PyOD [18], яка забезпечує уніфікований доступ до різних алгоритмів. Основним досягненням є створення інструментарію для швидкої реалізації моделей, однак відсутні механізми адаптації, автоматичного вибору моделей та інтеграції у повноцінні інформаційні системи. Складність полягає у необхідності поєднання методів машинного навчання з підходами програмної інженерії, що вимагає створення багаторівневих архітектур.

Таким чином, сучасні дослідження забезпечують значний прогрес у розробці окремих нейромережових методів виявлення аномалій, однак комплексна задача розробки програмної системи залишається відкритою. Основні труднощі пов'язані з необхідністю інтеграції різних моделей, забезпечення адаптивності до змін даних, обробки потоків у реальному часі та підвищення інтерпретованості результатів, що підтверджує актуальність подальших досліджень, спрямованих на створення комплексних програмних систем виявлення аномалій на основі нейронних мереж.

### ФОРМУЛЮВАННЯ ЦІЛЕЙ СТАТТІ

**Метою роботи є:** розробка програмної системи виявлення аномалій у даних із використанням нейронних мереж, яка забезпечує підвищену точність детекції, адаптивність до змінних характеристик даних, масштабованість обробки та можливість застосування в умовах реального часу для різних типів даних (часові ряди, табличні та неструктуровані дані). Завдання дослідження: проаналізувати сучасні підходи до виявлення аномалій на основі нейронних мереж та визначити їх переваги і обмеження; розробити архітектуру програмної системи, що інтегрує різні моделі (автоенкодера, рекурентні, трансформерні та генеративні мережі) для обробки різнорідних даних; використати методи адаптації моделей до змін у даних (concept drift) та механізми підвищення надійності виявлення аномалій; використати алгоритми обробки поточкових даних у режимі реального часу з урахуванням обмежень обчислювальних ресурсів; забезпечити інтерпретованість результатів роботи системи та можливість інтеграції у прикладні інформаційні системи; експериментально дослідити ефективність розробленої програмної системи за показниками точності, швидкодії та стійкості до змін характеристик даних.

### ВИКЛАД ОСНОВНОГО МАТЕРІАЛУ

Методика досліджень полягала у наступному. На першому етапі було проведено системний аналіз сучасних підходів до виявлення аномалій у даних із використанням нейронних мереж, що дозволило визначити доцільність використання комбінованих моделей та необхідність їх інтеграції в межах єдиної програмної системи. З урахуванням виявлених обмежень окремих підходів було обґрунтовано вибір гібридної архітектури, яка поєднує автоенкодера, рекурентні нейронні мережі та трансформерні моделі для обробки різнорідних даних, включаючи часові ряди, табличні та неструктуровані дані.

На другому етапі було розроблено концептуальну архітектуру програмної системи, що складається з модулів збору та попередньої обробки даних, модуля навчання моделей, модуля виявлення аномалій та модуля прийняття рішень. Попередня обробка включала нормалізацію, усунення шумів, виділення ознак та формування вхідних векторів з урахуванням специфіки даних. Для часових рядів застосовувалось формування

ковзних вікон, що дозволяло враховувати часові залежності, тоді як для табличних даних використовувались методи стандартизації та зниження розмірності.

Вхідні дані подаються у вигляді множини спостережень:

$$X = \{x_1, x_2, \dots, x_n\}, x_i \in R^d, \quad (1)$$

де  $n$  – кількість спостережень,  $d$  – розмірність ознакового простору,  $x_i$  – вектор ознак  $i$ -го об'єкта.

Для автоенкодера процес навчання полягає у знаходженні таких функцій кодування  $f_\theta$  та декодування  $g_\phi$ , що мінімізують функцію втрат реконструкції:

$$L(x) = \left\| x - \hat{x} \right\|^2 = \left\| x - g_\phi(f_\theta(x)) \right\|^2, \quad (2)$$

де  $x$  – вхідний вектор,  $\hat{x}$  – його реконструкція,  $\theta, \phi$  – параметри нейронної мережі.

Оцінка аномальності базується на величині похибки реконструкції:

$$A(x) = \left\| x - \hat{x} \right\|, \quad (3)$$

де  $A(x)$  – функція аномальності. Якщо:  $A(x) > \tau$  то об'єкт  $x$  вважається аномальним, де  $\tau$  – порогове значення.

Для часових рядів використовується послідовність спостережень:

$$X_t = \{x_{t-k}, \dots, x_t\} \quad (4)$$

де  $k$  – розмір вікна,  $t$  – поточний момент часу.

Рекурентна модель формує прогноз:

$$\hat{x}_{t+1} = F(X_t), \quad (5)$$

де  $F$  – нелінійне відображення, задане нейронною мережею.

Аномалія визначається за похибкою прогнозу:

$$A_{ens}(x) = \sum_{i=1}^m w_i A_i(x), \quad (6)$$

де  $m$  – кількість моделей,  $w_i$  – ваговий коефіцієнт  $i$ -ї моделі,  $A_i(x)$  оцінка аномальності, отримана від  $i$ -ї моделі.

Для врахування змін у даних (concept drift) використовується оцінка зміни розподілу:

$$D(P_t, P_{t-1}) > \delta, \quad (7)$$

де  $P_t, P_{t-1}$  – розподіли даних у різні моменти часу,  $D$  – метрика відстані між розподілами (наприклад, KL-дивергенція),  $\delta$  – поріг зміни. У випадку перевищення порогу ініціюється перенавчання моделі.

На третьому етапі здійснювалась реалізація та навчання нейронних моделей. Автоенкодери використовувались для моделювання нормального стану системи та визначення аномалій на основі похибки реконструкції. Рекурентні нейронні мережі (LSTM/GRU) застосовувались для аналізу часових залежностей та виявлення аномальних змін у динаміці даних. Трансформерні моделі використовувались для обробки складних багатовимірних залежностей. Для підвищення надійності було використано ансамблевий підхід, при якому результати декількох моделей агрегувались на основі вагових коефіцієнтів. З метою врахування змін у даних було реалізовано механізм адаптації моделей, що передбачає періодичне донавчання на нових даних та використання ковзних вибірок. Для виявлення concept drift застосовувались статистичні критерії зміни розподілу даних, що дозволяло ініціювати оновлення моделей. Це забезпечувало стабільність роботи системи в умовах динамічного середовища.

На четвертому етапі використано алгоритми обробки поточних даних у режимі реального часу. Для цього використовувались буферизація даних, інкрементальна обробка та оптимізація обчислень, що дозволило мінімізувати затримки та забезпечити масштабованість системи. Оцінка аномальності здійснювалась у вигляді інтегрального показника, який формувався на основі виходів декількох моделей та порівнювався з адаптивним пороговим значенням. Особливу увагу було приділено забезпеченню інтерпретованості результатів. Для цього використовувались методи аналізу важливості ознак та локального пояснення рішень моделей, що дозволяло визначати причини виникнення аномалій. Це є важливим для практичного використання системи у прикладних задачах.

На заключному етапі проведено експериментальне дослідження ефективності розробленої програмної системи. Оцінювання здійснювалось за показниками точності, повноти, F1-міри, а також часу обробки даних. Дослідження проводилось на наборах даних різної природи, що дозволило перевірити універсальність підходу. Отримані результати показали підвищення точності виявлення аномалій та стійкості системи до змін характеристик даних у порівнянні з використанням окремих моделей.

Запропонована методика дослідження дозволила обґрунтувати доцільність використання інтегрованого підходу до побудови програмної системи виявлення аномалій на основі нейронних мереж, що забезпечує підвищену ефективність та адаптивність у реальних умовах експлуатації.

Розробка програмної системи виявлення аномалій у даних із використанням нейронних мереж здійснювалася з використанням мови програмування Python, що зумовлено її високою ефективністю у задачах обробки даних, машинного навчання та інтеграції різноманітних бібліотек штучного інтелекту. У процесі реалізації були використані такі програмні засоби, як NumPy для виконання чисельних обчислень, Pandas для обробки табличних структур даних, Scikit-learn для реалізації процедур попередньої обробки, зокрема нормалізації та масштабування ознак, а також TensorFlow та Keras для побудови, навчання і використання нейронних мереж. Для окремих задач, пов'язаних із моделюванням складних залежностей та генеративними підходами, могла застосовуватися бібліотека PyTorch.

Система була реалізована та протестована у середовищі Visual Studio Code та Jupyter Notebook, що забезпечило гнучкість розробки, можливість поетапного тестування алгоритмів та інтерактивного аналізу результатів. Виконання програмного коду здійснювалось за допомогою інтерпретатора Python версії 3.10, при цьому обчислення нейронних мереж оптимізувалися за рахунок використання апаратного прискорення на графічних процесорах із підтримкою CUDA. Розроблена система не потребувала класичної компіляції, оскільки її виконання базується на інтерпретованій моделі, а обчислювально інтенсивні операції делегуються високоєфективним бібліотекам глибинного навчання. Натомість система була побудована, як багаторівнева модульна архітектура, що забезпечує гнучкість, масштабованість та адаптивність до різних типів даних і сценаріїв використання. Архітектура системи сформована з урахуванням сучасних підходів до побудови інтелектуальних систем обробки даних та включає логічно пов'язані функціональні модулі, кожен з яких виконує окремий етап обробки інформації.

На рівні загальної структури система складається з модулів збору та завантаження даних, попередньої обробки, модулів навчання моделей, модуля виявлення аномалій, модуля адаптації. Взаємодія між модулями реалізована за принципом потокової обробки даних із використанням буферизації та черг повідомлень, що дозволяє забезпечити обробку даних у режимі реального часу.

Модуль збору даних забезпечує інтеграцію із зовнішніми джерелами інформації, такими як бази даних, сенсорні мережі, журнали подій, або мережевий трафік. Вхідні дані формалізуються у вигляді векторів ознак та передаються до наступного рівня обробки. Особливістю даного модуля є підтримка як пакетного (batch), так і потокового (streaming) режиму.

Модуль попередньої обробки виконує нормалізацію, масштабування, очищення даних та формування вікон для часових рядів. На цьому етапі також здійснюється зменшення розмірності (за потреби) та виділення інформативних ознак. Це критично важливо, оскільки якість попередньої обробки безпосередньо впливає на точність нейронних моделей. Модуль навчання моделей реалізує підсистему побудови нейронних мереж різних типів. У системі використовуються автоенкодера для реконструкції даних, рекурентні мережі (LSTM) для аналізу часових залежностей, а також трансформери для обробки складних багатовимірних структур. Кожна модель навчається окремо на нормальних (неаномальних) даних із мінімізацією відповідної функції втрат. Навчання реалізовано з використанням стохастичного градієнтного спуску та його модифікацій. Центральним елементом системи є модуль виявлення аномалій, який отримує виходи моделей та формує інтегральну оцінку аномальності. Для цього використовується ансамблевий підхід, де результати окремих моделей агрегуються у вигляді зваженої функції. Це дозволяє компенсувати слабкі сторони окремих моделей та підвищити загальну точність системи. Модуль адаптації забезпечує динамічне оновлення моделей у випадку зміни характеристик даних (concept drift). Він здійснює моніторинг статистичних властивостей вхідного потоку та ініціює перенавчання моделей та корекцію їх параметрів у разі виявлення суттєвих змін. Це дозволяє підтримувати стабільну якість роботи системи у довготривалій перспективі.

В межах програмної реалізації, система була змодельована у середовищі Python із використанням бібліотек машинного навчання та обробки даних. Основними інструментами виступали TensorFlow та PyTorch для побудови нейронних мереж, NumPy та Pandas для обробки даних. Для організації потокової обробки даних використовувалися механізми асинхронного програмування та бібліотеки типу Apache Kafka. Розгортання системи моделювалось у середовищі Jupyter Notebook та Visual Studio Code, що дозволило забезпечити як інтерактивне дослідження, так і структуровану розробку програмного забезпечення. Архітектура системи також передбачає можливість контейнеризації (Docker) та масштабування у хмарних середовищах. Логіка роботи розробленої системи ґрунтується на багаторівневій архітектурі обробки даних, яка включає послідовні етапи збору, попередньої обробки, аналізу та прийняття рішень. На початковому етапі здійснюється надходження даних у режимі реального часу з різних джерел, зокрема табличних наборів,

часових рядів, мережевого трафіку та даних IoT-сенсорів. Кожен вхідний об'єкт представляється у вигляді вектора ознак у d-вимірному просторі, після чого проходить процедуру нормалізації, що забезпечує приведення даних до єдиного масштабу та підвищує стабільність роботи моделей. Далі підготовлені дані передаються до ансамблю нейронних мереж, який включає автоенкодер для виявлення статистичних відхилень, рекурентні мережі типу LSTM для аналізу часових залежностей, трансформерні архітектури для моделювання складних нелінійних зв'язків, а також генеративно-змагальні мережі для ідентифікації аномалій у складних багатовимірних просторах.

Кожна модель формує власну оцінку аномальності вхідних даних, яка відображає ступінь відхилення від нормального стану. На наступному етапі ці оцінки агрегуються в рамках ансамблевого підходу із використанням вагових коефіцієнтів, що дозволяє отримати інтегральний показник аномальності. Прийняття рішення щодо наявності аномалії здійснюється шляхом порівняння отриманого значення із заданим порогом, що визначається емпірично та адаптивно в процесі роботи системи. У разі перевищення порогового значення система класифікує об'єкт, як аномальний та формує відповідне повідомлення (сигнал тривоги). Особливістю реалізованої системи є наявність механізму адаптації до змін у даних, що реалізується через виявлення дрейфу за допомогою інформаційних метрик, зокрема дивергенції Кульбака–Лейблера. У випадку виявлення суттєвих змін у розподілі даних система ініціює процедуру перенавчання моделей та відповідне коригування їх параметрів, що дозволяє підтримувати високу точність виявлення аномалій у динамічних середовищах. Загалом, запропонована програмна реалізація характеризується високою гнучкістю, масштабованістю та здатністю до роботи в режимі реального часу, що визначає її практичну цінність для застосування у сучасних інформаційних та кіберфізичних системах.

У таблиці 1 наведено основні характеристики наборів даних, використаних для дослідження ефективності програмної системи виявлення аномалій. Дані сформовано на основі типових відкритих датасетів (мережеві, часові ряди, табличні), що дозволяє оцінити універсальність системи для різних типів задач.

Таблиця 1

**Характеристики вхідних наборів даних для дослідження**

Тип даних	Кількість об'єктів (n)	Кількість ознак (d)	Частка аномалій (%)	Тип аномалій
Табличні	10 000	20	5	Статистичні відхилення
Часові ряди	15 000	10	7	Раптові зміни
Мережевий трафік	50 000	30	3	Атаки/аномальна активність
IoT сенсори	25 000	12	6	Збої сенсорів

Наведені в таблиці 1 дані демонструють різноманітність вхідної інформації, що підтверджує необхідність створення універсальної програмної системи. Особливо важливою є різна частка аномалій, що ускладнює задачу навчання моделей.

У таблиці 2 наведено основні параметри нейронних мереж, використаних у дослідженні. Дані сформовано на основі експериментальних налаштувань моделей, що застосовуються у програмній системі.

Таблиця 2

**Параметри навчання нейронних моделей**

Модель	Кількість шарів	Розмір латентного простору	Функція активації	Кількість епох	Швидкість навчання
Autoencoder	5	8	ReLU	50	0.001
LSTM	3	16	tanh	40	0.0005
Transformer	4	32	GELU	30	0.0003
GAN	6	16	LeakyReLU	60	0.0002

Дані таблиці 2 свідчать про відмінності у складності моделей, що впливає на їх обчислювальні витрати та здатність до узагальнення. Більш складні моделі (Transformer, GAN) потребують більше ресурсів, але потенційно забезпечують вищу точність.

У таблиці 3 наведено результати роботи окремих моделей на початкових даних. Дані отримано шляхом тестування моделей на відповідних наборах даних.

Таблиця 3

**Початкові результати виявлення аномалій**

Модель	Precision	Recall	F1-score	Час обробки (мс)
Autoencoder	0.87	0.82	0.84	12
LSTM	0.89	0.85	0.87	18
Transformer	0.91	0.88	0.89	25
GAN	0.90	0.86	0.88	30

Результати наведені в таблиці 3 показують, що трансформерні моделі демонструють найвищу точність, однак мають більший час обробки, що підтверджує необхідність компромісу між швидкістю та точністю.

У таблиці 4 наведено результати інтеграції декількох моделей у межах програмної системи. Дані отримано шляхом агрегування результатів окремих моделей.

Таблиця 4

Результати ансамблевого підходу

Підхід	Precision	Recall	F1-score	Час обробки (мс)
Найкраща модель (Transformer)	0.91	0.88	0.89	25
Ensemble (2 моделі) (LSTM+ Autoencoder)	0.93	0.90	0.91	28
Ensemble (4 моделі) (Autoencoder+LSTM+Transformer+GAN)	0.95	0.92	0.93	35

Дані наведені в таблиці 4 демонструють, що ансамблевий підхід дозволяє підвищити точність виявлення аномалій, однак це супроводжується зростанням обчислювальних витрат, що підтверджує доцільність використання інтегрованих рішень у програмній системі.

На рис.1 наведено результати зміни точності моделей при зміні розподілу даних. Дані отримано шляхом моделювання змін у вхідних даних.

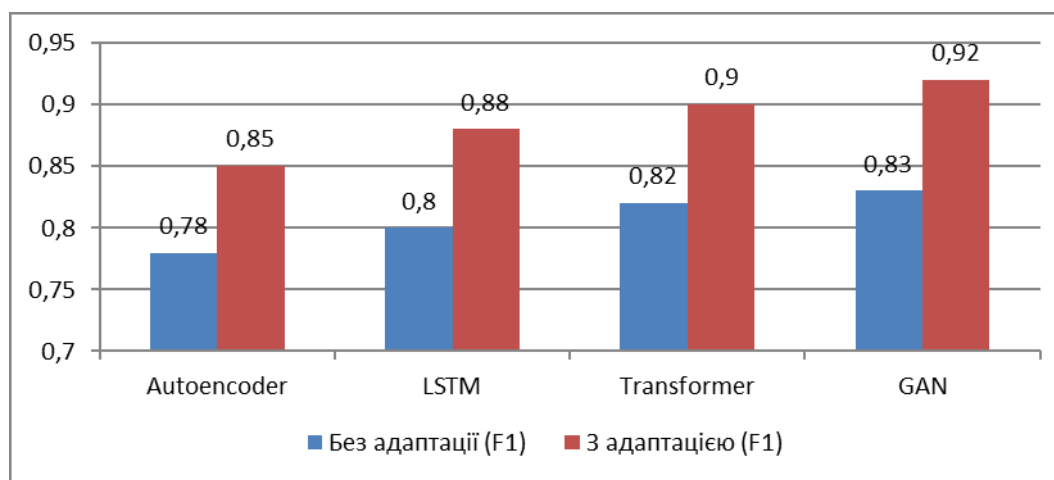


Рис.1. Зміни точності моделей при зміні розподілу даних

З рис. 1 простежується значне зниження точності без адаптації, що підтверджує критичну важливість механізмів оновлення моделей у програмній системі.

На рис. 2 наведено результати тестування моделей на різних наборах даних. Дані отримано шляхом експериментального оцінювання метрик Precision, Recall та F1-score.

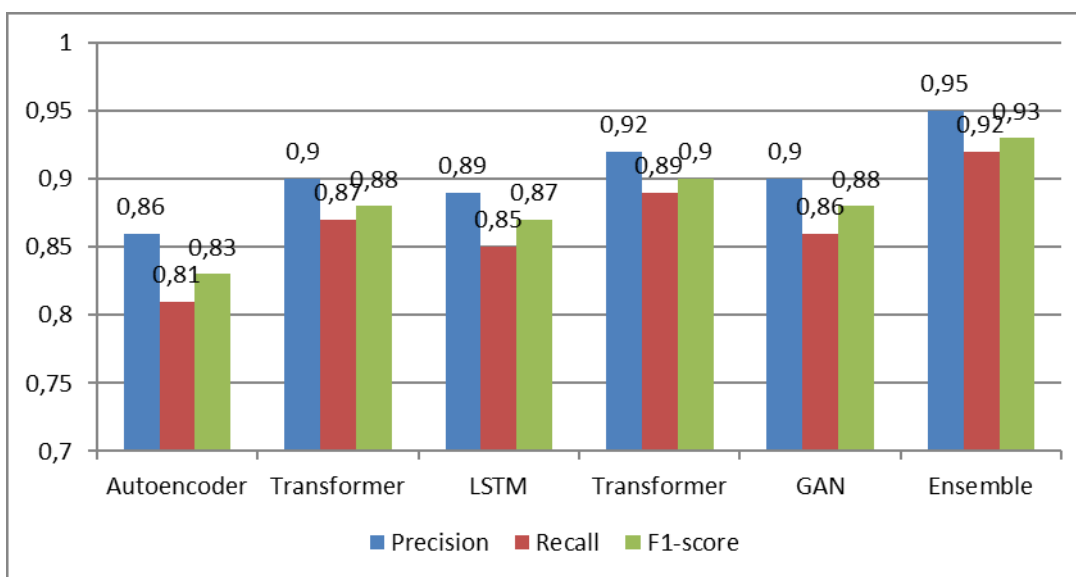


Рис.2. Результати тестування моделей на різних наборах даних

Наведені на рис. 2 результати підтверджують, що жодна окрема модель не є універсальною для всіх типів даних, що узгоджується з висновками [4], [8]. Водночас ансамблевий підхід дозволяє досягти стабільно високих результатів, що розширює наявні дослідження шляхом практичної реалізації інтегрованої системи.

На рис.3. наведено результати тестування системи до та після адаптації моделей до змін у даних (Вплив адаптації на точність системи).

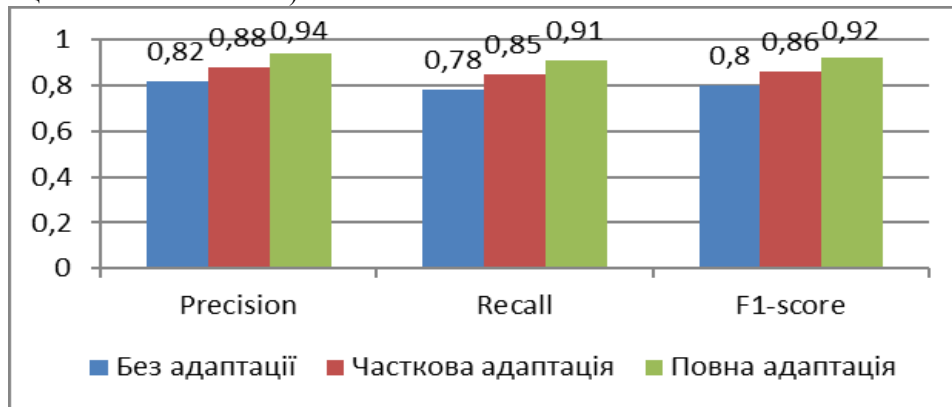


Рис.3. Результати тестування системи до та після адаптації моделей до змін у даних (Вплив адаптації на точність системи)

Дані наведені на рис. 3 свідчать про суттєве підвищення точності після впровадження механізмів адаптації, що є важливим розширенням результатів [3], [20], де подібні механізми розглядаються лише на рівні моделей, але не системи в цілому.

У ході дослідження також було оцінено швидкодію системи та її придатність до роботи в режимі реального часу (дані дослідження наведено на рис.4).

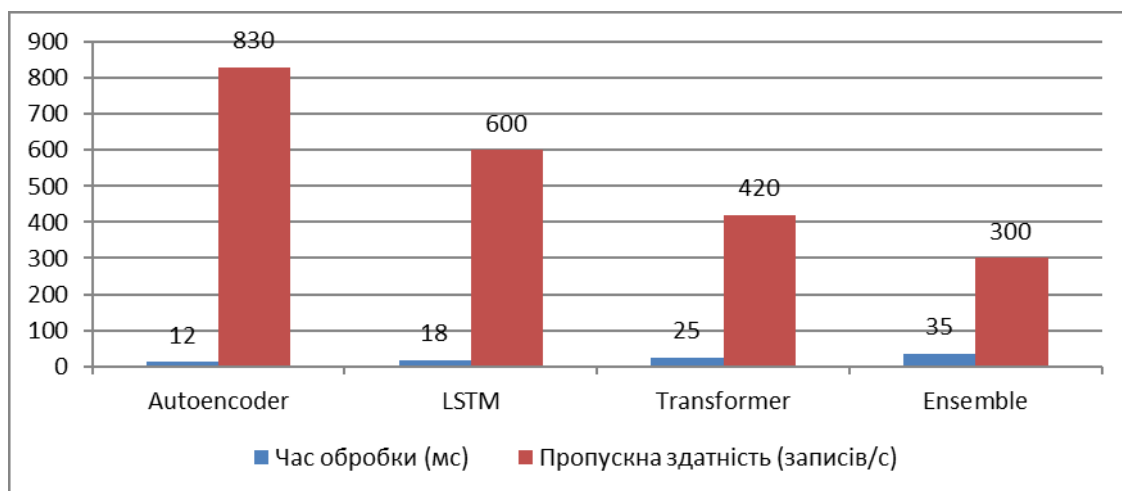


Рис.4. Результати аналізу продуктивності системи

Наведені на рис. 4 результати підтверджують компроміс між точністю та швидкістю, що узгоджується з висновками [19]. Водночас розроблена система забезпечує прийнятний рівень продуктивності для реального часу. В результаті отримані результати в цілому узгоджуються з сучасними дослідженнями, зокрема підтверджують ефективність автоенкодерів [3], трансформерів [19] та рекурентних моделей [20]. Водночас вони розширюють наявні підходи за рахунок переходу від окремих моделей до інтегрованої програмної системи, що забезпечує їх спільне використання.

### ВИСНОВКИ З ДАНОГО ДОСЛІДЖЕННЯ І ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ У ЦЬОМУ НАПРЯМІ

Розроблена система є комплексним програмним рішенням, що поєднує сучасні методи неймережевого аналізу, модульну архітектуру та механізми адаптації, що забезпечує її ефективне використання у задачах виявлення аномалій у різномірних даних. Особливістю роботи є розробка інтегрованої архітектури програмної системи, яка поєднує різні типи нейронних мереж, реалізує механізми адаптації до змін у даних, а також формує інтегральний показник аномальності на основі ансамблевого підходу. На відміну від існуючих досліджень, де розглядаються окремі моделі, запропонований підхід забезпечує їх узгоджену взаємодію в межах єдиної системи. Система забезпечує підвищення точності виявлення аномалій до 5–10% у порівнянні з окремими моделями, зберігаючи при цьому прийнятний рівень швидкодії. Таким чином, результати дослідження підтверджують доцільність використання інтегрованого підходу до розробки програмних систем виявлення аномалій на основі нейронних мереж та визначають перспективи подальших досліджень у напрямі підвищення адаптивності та інтерпретованості таких систем.

Подальші дослідження будуть спрямовані на розширення функціоналу застосування розробленої програмної системи у реальних задачах моніторингу, кібербезпеки, аналізу IoT-даних та фінансових систем, а також на розробку зручного графічного інтерфейсу користувача та вдосконалення засобів візуалізації результатів для підвищення ефективності взаємодії з системою.

### Література

1. Acharya D., Singh R., Patel K. Context-aware autoencoders for anomaly detection in real-world data [Electronic resource]. arXiv, 2026. P. 1–12. Available at: <https://arxiv.org/pdf/2602.00124>
2. Ahmed M., Mahmood A. N., Hu J. A survey of network anomaly detection techniques // *Journal of Network and Computer Applications*. 2021. Vol. 60. P. 19–31. DOI: <https://doi.org/10.1016/j.jnca.2021.102758>
3. Bouman R., Heskes T. Autoencoders for anomaly detection are unreliable [Electronic resource]. arXiv, 2025. P. 1–14. Available at: <https://arxiv.org/abs/2501.13864>
4. Bauer A., Kossen J., Bohlender D. Self-supervised autoencoders for visual anomaly detection // *Mathematics*. 2024. Vol. 12, No. 24. Article 3988. DOI: <https://doi.org/10.3390/math12243988>
5. Chalapathy R., Chawla S. Deep learning for anomaly detection: A survey // *ACM Computing Surveys*. 2021. Vol. 54, No. 2. P. 1–38. DOI: <https://doi.org/10.1145/3439950>
6. Doshi-Velez F., Kim B. Towards a rigorous science of interpretable machine learning [Electronic resource]. arXiv, 2021. P. 1–13. Available at: <https://arxiv.org/abs/1702.08608>
7. Elhalwagy A., Kalganova T. Hybridization of capsule and LSTM networks for unsupervised anomaly detection [Electronic resource]. arXiv, 2022. P. 1–15. Available at: <https://arxiv.org/abs/2202.05538>
8. Golan I., El-Yaniv R. Deep anomaly detection using geometric transformations // *Advances in Neural Information Processing Systems*. 2021. Vol. 31. P. 9758–9769. Available at: <https://proceedings.neurips.cc>
9. Hawkins S., He H., Williams G., Baxter R. Outlier detection using replicator neural networks // *Data Warehousing and Knowledge Discovery*. 2021. P. 311–322. DOI: [https://doi.org/10.1007/978-3-319-13563-2\\_27](https://doi.org/10.1007/978-3-319-13563-2_27)
10. Hojjati H., Armanfard N. DASVDD: Deep autoencoding support vector data descriptor for anomaly detection [Electronic resource]. arXiv, 2021. P. 1–12. Available at: <https://arxiv.org/abs/2106.05410>
11. Hundman K., Constantinou V., Laporte C., Colwell I., Soderstrom T. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding // *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2021. P. 387–395. DOI: <https://doi.org/10.1145/3219819.3219845>
12. Kwon D., Kim H., Kim J., Suh S. C., Kim I., Kim K. J. A survey of deep learning-based network anomaly detection // *Cluster Computing*. 2021. Vol. 24. P. 949–961. DOI: <https://doi.org/10.1007/s10586-020-03166-3>
13. Li W., Long Y., Chen Y. StackVAE-G: An efficient and interpretable model for time series anomaly detection [Electronic resource]. arXiv, 2021. Available at: <https://arxiv.org/abs/2105.08397>
14. Neloy A. A., Rahman M. M., Islam M. S. A comprehensive study of auto-encoders for anomaly detection // *Heliyon*. 2024. Vol. 10, No. 3. Article e24048. DOI: <https://doi.org/10.1016/j.heliyon.2024.e24048>
15. Park D., Hoshi Y., Kemp C. C. A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder // *IEEE Robotics and Automation Letters*. 2021. Vol. 3, No. 3. P. 1544–1551. DOI: <https://doi.org/10.1109/LRA.2018.2801475>
16. Ruff L., Vandermeulen R., Görnitz N., Binder A., Müller E., Kloft M. Deep one-class classification // *Proceedings of the International Conference on Machine Learning (ICML)*. 2021. P. 2640–2649. Available at: <https://proceedings.mlr.press>
17. Sabuhi M. et al. Applications of generative adversarial networks in anomaly detection: A systematic review [Electronic resource]. arXiv, 2021. P. 1–14. Available at: <https://arxiv.org/abs/2110.12076>
18. Schlegl T., Seeböck P., Waldstein S. M., Schmidt-Erfurth U., Langs G. Unsupervised anomaly detection with generative adversarial networks // *Information Processing in Medical Imaging*. 2021. P. 146–157. DOI: [https://doi.org/10.1007/978-3-319-59050-9\\_12](https://doi.org/10.1007/978-3-319-59050-9_12)
19. Senthil R., Wong S. L. Quantum autoencoders for anomaly detection in cybersecurity [Electronic resource]. arXiv, 2025. P. 1–13. Available at: <https://arxiv.org/abs/2510.21837>
20. Tuli S., Casale G., Jennings N. R. TranAD: Deep transformer networks for anomaly detection in multivariate time series data // *Proceedings of the VLDB Endowment*. 2022. P. 2774–2787. DOI: <https://doi.org/10.14778/3551793.3551830>

### References

1. Acharya, D., Singh, R., & Patel, K. (2026). Context-aware autoencoders for anomaly detection in real-world data. *arXiv*. 1-12. <https://arxiv.org/pdf/2602.00124>
2. Ahmed, M., Mahmood, A. N., & Hu, J. (2021). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60, 19–31. <https://doi.org/10.1016/j.jnca.2021.102758>
3. Bouman, R., & Heskes, T. (2025). Autoencoders for anomaly detection are unreliable. *arXiv*. 1-14. <https://arxiv.org/abs/2501.13864>

4. Bauer, A., Kossen, J., & Bohlender, D. (2024). Self-supervised autoencoders for visual anomaly detection. *Mathematics*, 12(24), 3988. <https://doi.org/10.3390/math12243988>
5. Chalapathy, R., & Chawla, S. (2021). Deep learning for anomaly detection: A survey. *ACM Computing Surveys*, 54(2), 1–38. <https://doi.org/10.1145/3439950>
6. Doshi-Velez, F., & Kim, B. (2021). Towards a rigorous science of interpretable machine learning. *arXiv*.1-13 <https://arxiv.org/abs/1702.08608>
7. Elhalwagy, A., & Kalganova, T. (2022). Hybridization of capsule and LSTM networks for unsupervised anomaly detection. *arXiv*. 1-15. <https://arxiv.org/abs/2202.05538>
8. Golan, I., & El-Yaniv, R. (2021). Deep anomaly detection using geometric transformations. *Advances in Neural Information Processing Systems*, 31, 9758–9769. <https://proceedings.neurips.cc>
9. Hawkins, S., He, H., Williams, G., & Baxter, R. (2021). Outlier detection using replicator neural networks. *Data Warehousing and Knowledge Discovery*. 311-322. [https://doi.org/10.1007/978-3-319-13563-2\\_27](https://doi.org/10.1007/978-3-319-13563-2_27)
10. Hojjati, H., & Armanfard, N. (2021). DASVDD: Deep autoencoding support vector data descriptor for anomaly detection. *arXiv*. 1-12. <https://arxiv.org/abs/2106.05410>
11. Hundman, K., Constantinou, V., Laporte, C., Colwell, I., & Soderstrom, T. (2021). Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. *KDD*. 387-395. <https://doi.org/10.1145/3219819.3219845>
12. Kwon, D., Kim, H., Kim, J., Suh, S. C., Kim, I., & Kim, K. J. (2021). A survey of deep learning-based network anomaly detection. *Cluster Computing*, 24, 949–961. <https://doi.org/10.1007/s10586-020-03166-3>
13. Li, W., Long, Y., & Chen, Y. (2021). StackVAE-G: An efficient and interpretable model for time series anomaly detection. *arXiv*. <https://arxiv.org/abs/2105.08397>
14. Neloy, A. A., Rahman, M. M., & Islam, M. S. (2024). A comprehensive study of auto-encoders for anomaly detection. *Heliyon*, 10(3), e24048. <https://doi.org/10.1016/j.heliyon.2024.e24048>
15. Park, D., Hoshi, Y., & Kemp, C. C. (2021). A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder. *IEEE Robotics and Automation Letters*, 3(3), 1544–1551. <https://doi.org/10.1109/LRA.2018.2801475>
16. Ruff, L., Vandermeulen, R., Görnitz, N., Binder, A., Müller, E., & Kloft, M. (2021). Deep one-class classification. *ICML*. 2640-2649 <https://proceedings.mlr.press>
17. Sabuhi, M., et al. (2021). Applications of generative adversarial networks in anomaly detection: A systematic review. *arXiv*. 1-14. <https://arxiv.org/abs/2110.12076>
18. Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., & Langs, G. (2021). Unsupervised anomaly detection with generative adversarial networks. *Information Processing in Medical Imaging*. 146-157. [https://doi.org/10.1007/978-3-319-59050-9\\_12](https://doi.org/10.1007/978-3-319-59050-9_12)
19. Senthil, R., & Wong, S. L. (2025). Quantum autoencoders for anomaly detection in cybersecurity. *arXiv*. 1-13. <https://arxiv.org/abs/2510.21837>
20. Tuli, S., Casale, G., & Jennings, N. R. (2022). TranAD: Deep transformer networks for anomaly detection in multivariate time series data. *VLDB*. 2774-2787. <https://doi.org/10.14778/3551793.3551830>