https://doi.org/10.31891/2307-5732-2025-351-14
УДК 004.75

ХВОРОСТЯНИЙ РОДІОН
Державний університет інформаційно-комунікаційних технологій
https://orcid.org/0009-0004-4591-7100
e-mail: rodionhvorostyanoy@gmail.com
КОРЕЦЬКИЙ ОЛЕКСАНДР
Державний університет інформаційно-комунікаційних технологій
https://orcid.org/0009-0001-4809-7556
e-mail: okoretsky@gmail.com

МЕТОДИКА ПОБУДОВИ ПЛАТФОРМИ ШТУЧНОГО ІНТЕЛЕКТУ МОНІТОРИНГУ КОНФІДЕНЦІЙНОГО ТА АНОМАЛЬНОГО ТРАФІКУ ПРИ УМОВІ МІНІМІЗАЦІЇ ЧАСУ РОЗГОРТАННЯ ТА НАДІЙНОСТІ ФУНКЦІОНУВАННЯ

В роботі подані результати розробки модель побудови платформи штучного інтелекту при умові мінімізації часу розгортання та забезпечення надійності роботи в ході моніторингу корпоративного та аномального трафіку інформаційно-комунікаційної мережі передачі даних. При створені моделі застосовано технологію MLOps, чим показано, що її використання дозволяє створити ефективний спосіб розгортання платформ штучного інтелекту.

Подана модель побудови платформа має можливість до автоматизації та моніторингу етапів збору даних, перетворення, навчання, впровадження та представлення результатів кінцевому користувачеві. Етап розгортання платформи забезпечує швидку інтеграції, мінімізацію часу розгортання інтеграцію та надійність функціонування моделі з постійним її моніторингом. Висвітлені в роботі результати оцінки ефективності застосування розробленої моделі платформи штучного інтелекту що при розвитку топології платформи, від 3 до 63 вузлів час розгортання збільшується лише на 434 секунди, що свідчить про здатність платформи до адаптації та розвитку. Тестування навантаженням показала, що подана модель платформи здатна функціонувати без відмов, за умови навантаження черги запитів від 25 потоків одночасно, при цьому застосування додаткових вузлів забезпечує мінімізацію середнього часу відповіді системи.

Ключові слова: штучний інтелект, інформаційно-комунікаційні мережі, модель побудови платформи штучного інтелекту, час розгортання платформи штучного інтелекту, надійність моделі штучного інтелекту

KHVOROSTIANYI RODION
KORETSKYI OLEKSANDR
State University of Information and Communication Technologies

METHODOLOGY FOR CONSTRUCTING AN ARTIFICIAL INTELLIGENCE PLATFORM FOR MONITORING CONFIDENTIAL AND ANOMALOUS TRAFFIC WITH MINIMIZATION OF DEPLOYMENT TIME AND RELIABILITY OF OPERATION

The paper presents the results of developing a model for building an artificial intelligence platform under the condition of minimizing deployment time and ensuring reliability of operation during monitoring of corporate and anomalous traffic of the information and communication network. When creating the model, MLOps technology was used, which shows that its use allows you to create an effective way to deploy artificial intelligence platforms. A feature of the presented binary tree is its relatively low reliability, when in case of a single failure, the topology can actually be divided into two parts. To minimize this problem, it is advisable to use the De Bruyne technology to form additional ties.

The presented model for building the platform has the ability to automate and monitor the stages of data collection, transformation, training, implementation and presentation of results to the end user. The platform deployment stage provides quick integration, minimizing deployment time, integration and reliability of the model with its constant monitoring. The paper highlights the results of assessing the effectiveness of the application of the developed artificial intelligence platform model, which shows that when developing the platform topology from 3 to 63 nodes, the deployment time increases by only 434 seconds, which indicates the platform's ability to adapt and develop. Load testing showed that the presented platform model is able to function without failures, provided that the request queue is loaded with 25 threads simultaneously, while the use of additional nodes ensures the minimization of the average system response time.

Keywords: artificial intelligence, information and communication networks, artificial intelligence platform construction model, artificial intelligence platform deployment time, artificial intelligence model reliability

Стаття надійшла до редакції / Received 11.04.2025
Прийнята до друку / Accepted 26.04.2025

Постановка проблеми

Низка важливих завдань підвищення ефективності передачі даних в транспортних та корпоративних інформаційно-комунікаційних мережах різного призначення вимагає постійного аналізу, моніторингу та розподілу великих масивів даних на необхідному рівні якості обслуговування в короткі терміни. Частину таких завдань визначимо як моніторинг та управління вхідним конфіденційним трафіком визначеної мережі та забезпечення кібербезпеки мережі передачі даних методом моніторингу аномального рафіку.

Є очевидним, що на сучасному етапі розвитку ІТ-технологій більшість завдань аналізу, моніторингу та обробки великих масивів даних, включно з конфіденційними даними та аномальними запитами в ході кібератак на мережу з успіхом вирішуються механізмами та технологіями штучного інтелекту [1]. Залучення таких технологій, в свою чергу передбачає створення відповідних програмно-апаратних комплексів та платформ штучного інтелекту (AI-систем), підготовку їх до використання та оцінку ефективності застосування AI-систем при вирішенні визначеного кола задач відповідно функціонального призначення [1,2].

Одним з важливих етапів розробки платформи штучного інтелекту є її підготовка до застосування, а саме навчання платформи обробки відповідних масивів даних, тобто застосування відповідних процесів та технологій машинного навчання [2].

Побудови конвеєра машинного навчання для конкретної моделі штучного інтелекту передбачає наступні етапи: підготовка даних; навчання моделі; розгортання моделі (платформи чи програмно-апаратного комплексу її функціонування); моніторинг застосування моделі. [1,2]

Однією з технологій, яка може ефективно вирішити задання машинного навчання є технологія конвеєра машинного навчання Machine Learning Operations та Machine Learning Pipeline (MLOps) [1,2].

Технологія MLOps дозволяє значно пришвидшити інтеграцію моделей у хмарні інфраструктури та підвищити надійність функціонування завдяки постійному контролю стану системи й можливості її динамічного налаштування. Програмно-апаратні комплекси, побудовані за цим принципом, можуть ефективно застосовуватись у таких сферах, як охорона здоров'я, безпека чи логістика. Використання таких практик також сприяє оперативному оновленню або поглибленому багатоетапному додатковому навчанню моделей, підвищуючи точність і ефективність прийняття рішень.

Щоб оцінити загальний рівень ефективності використання моделі штучного інтелекту з визначеною точністю та надійністю, необхідно провести її попереднє тестування й аналітичну оцінку процесу застосування, зокрема в частині розгортання та забезпечення стабільної роботи. Такий підхід дає змогу не лише своєчасно виявляти можливі недоліки, а й перевіряти ефективність функціонування системи в режимі реального часу. У перспективі впровадження подібних рішень на рівні підприємств сприятиме глибшій автоматизації процесів і підвищенню точності не лише в рамках AI-модулів, але й у ширшому контексті бізнес-операцій. Це, своєю чергою, забезпечить більш надійне та ефективне використання технологій штучного інтелекту в практичній діяльності.

Аналіз сучасних досліджень і публікацій

На сьогодні дедалі більше наукових праць присвячено впровадженню концепції конвеєра машинного навчання як інноваційного підходу до розгортання систем на основі штучного інтелекту. Цей підхід значною мірою ґрунтується на наборах наборах методик, інструментів и філософії побудови, які дозволяють автоматизувати і інтегрувати між собою процеси та процедури розробки програмного забезпечення та ИТ технологій (технологія DevOps), які вже продемонстрували ефективність у галузі розробки програмного забезпечення. У ряді публікацій [3,4] описуються різноманітні інструменти та конфігурації побудови конвеєрів, які забезпечують масштабоване виконання ШІ-задач і сприяють підвищенню стійкості систем до збоїв. Застосування таких технологій дозволяє автоматизувати критичні процеси, що зазвичай потребують значних людських ресурсів — таких як моніторинг, документування подій та пере конфігурація системи. Проте більшість існуючих рішень лише частково охоплюють аспекти продуктивності, а отже, не дають повної картини ефективності запропонованих підходів.

Наукові дослідження [3,4,5] також засвідчують потенціал інтеграції вбудованих пристроїв у процеси MLOps, однак вказують на потребу у подальшому вивченні можливості по розробці платформ штучного інтелекту з урахування визначених умов функціонування. Зокрема, недостатньо дослідженими залишаються питання надійності при пікових навантаженнях, а також ефективність виконання задач штучного інтелекту у середовищах розподілених обчислень.

Одним із ключових напрямків, важливих при врахування в процесі побудови платформи є аналіз часових характеристик розгортання таких систем і порівняння з класичними архітектурними рішеннями моделей штучного інтелекту.

Мета та завдання дослідження.

Метою даної роботи є розробка методики побудови платформи штучного інтелекту моніторингу конфіденційного та аномального трафіку при умові мінімізації часу розгортання та надійності функціонування.

Для досягнення поставленої мети визначено такі завдання:

— провести аналіз можливостей до розробки платформи штучного інтелекту, на основі технології MLOps, і сформувати основні етапи її реалізації;

— розробити модель побудови платформи штучного інтелекту при умові мінімізації часу розгортання та забезпечення надійності роботи;

— здійснити оцінку ефективності застосування розробленої моделі платформи штучного інтелекту при умові мінімізації часу розгортання та надійної роботи в умовах штатного навантаження корпоративним та аномальним трафіком мережі.

Матеріал і результати дослідження

Технологія Machine Learning Operations (MLOps) стала результатом розвитку підходів та методів DevOps, в частині впровадження спеціальних процедур, які дозволяють об'єднати інструменти, методи розгортання та організацію робочих механізмів ефективної побудови моделей штучного інтелекту на етапі реалізації машинного навчання [3]. Технологія MLOps дозволяє провести автоматизацію всіх ключових етапів життєвого циклу моделей — від створення до впровадження — із забезпеченням прозорості, моніторингу та фіксації внутрішніх операцій [3,4]. Такий підхід вимагає чіткої координації між програмними компонентами, що відповідають за основні функції: збір та обробку даних, безперервне навчання моделі, її оновлення, впровадження та взаємодію з користувачем.

Завдяки впровадженню технології MLOps, розробники отримують змогу підтримувати кінцевий продукт у стабільному та ефективному стані впродовж усього періоду його використання. У протилежному випадку, коли відсутня система підтримки моделей, їхня продуктивність може знижуватися через зміну вхідних даних, які раніше не використовувалися під час навчання [4,5]. Отже, важливою складовою довготривалої ефективності платформ ШІ є постійний моніторинг їх роботи, що є однією із ключових переваг у рамках застосування технології MLOps.

У межах запропонованого підходу з'являється можливість до побудови узагальненого конвеєра машинного навчання, орієнтованого на потреби систем штучного інтелекту. Такий конвеєр зазвичай стартує з інтеграційного етапу, під час якого актуальні зміни з системи керування версіями моделей об'єднуються в основну топологію моделі. Після цього автоматично запускається набір стандартних тестів, який включає як модульну перевірку окремих компонентів, так і інтеграційне тестування моделей машинного навчання з використанням визначених метрик ефективності.

Оскільки платформа (програмно-апаратний комплекс) для виконання завдань штучного інтелекту поєднує програмні та апаратні ресурси, вона потребує розробки окремої моделі машинного навчання, яка, як правило, є експериментальною. Типовий процес створення моделі включає кілька обов'язкових кроків: збирання додаткових відомостей щодо вхідних даних і формулювання цілей моделювання, попередній аналіз даних (Exploratory Data Analysis, EDA), очищення й підготовку (включно з інженерією ознак і виділенням релевантних параметрів). На основі цих дій формуються навчальні, тестові й валідаційні набори даних, необхідні для повноцінного навчання і перевірки моделі [4,5].

Реалізація MLOps-процесу передбачає багаторівневу архітектуру, яка охоплює валідацію, тестування та інтеграцію компонентів системи штучного інтелекту. Ці послідовні етапи формують цілісний конвеєр машинного навчання, що забезпечує автоматизоване впровадження та оновлення моделі штучного інтелекту.

Спосіб реалізація MLOps включає наступні етапи:

1. Безперервна інтеграція (CI) є ключовою практикою, що полягає у регулярному об'єднанні змін у кодовій базі. Це дозволяє оперативно виявляти і вирішувати конфлікти, які виникають між паралельними змінами, внесеними різними учасниками команди [2].

2. Безперервна доставка (CD) - передбачає впровадження змін у коді та підготовку додатка до негайного розгортання у продуктивному середовищі. Від подальшого етапу — безперервного розгортання — його відрізняє те, що саме розгортання може виконуватись вручну у разі недоступності нової моделі, тоді як усі попередні процеси є повністю автоматизованими. До безперервної доставки відносять кроки: компіляцію дистрибутиву, розгортання, моніторинг та аналітика.

Фінальним етапом реалізації є постійний контроль за станом платформи — збір метрик продуктивності, системних показників і виявлення відхилень від нормального режиму роботи. До таких девіацій відносяться відмови як на рівні програмного забезпечення, так і апаратної частини. Реакція системи на критичні події реалізується шляхом автоматизованих механізмів відновлення, які можуть передбачати повне або часткове повторне виконання CI/CD-процесу для відновлення функціональності.

Після здійснення етапу тестування на точність, модель разом із відповідним звітом про результати передається до системи управління моделями. Для ефективної реалізації програмної інфраструктури, яка використовує штучний інтелект, необхідно враховувати цілу низку факторів, серед яких важливу роль відіграє вибір оптимальних технологічних засобів і інструментів. Від їхнього правильного добору залежить здатність побудувати гнучку, масштабовану і стійку до навантажень інфраструктуру для запуску AI-рішень.

Завершальними стадіями реалізації конвеєра машинного навчання є розгортання та моніторинг. У цей період відбувається інтеграція перевіреної моделі у відповідний застосунок, з акцентом на забезпечення стабільності та безперервності функціонування всієї програмної системи. Оскільки саме на цьому етапі формується взаємодія між усіма складовими рішення, інженери MLOps здійснюють постійне спостереження за станом моделі, працездатністю додатку та якістю вхідних даних [3,4]. Водночас у процес розгортання залучається фахівець, відповідальний за налаштування, тестування та запуск готової системи в робочому середовищі.

Після завершення навчання модель може бути впроваджена для подальшого використання кінцевими користувачами за допомогою різних стратегій. Найбільш розповсюдженими є два підходи: «модель як сервіс» (Model-as-a-Service) та «вбудована модель» (Embedded Model) [5,6].

Вибір стратегії розгортання AI-моделі безпосередньо залежить від характеру взаємодії користувача з платформою. У рамках визначеної в даній роботі пролеми щодо моніторингу корпоративного та аномального рафіку пропонується використати підхід Model-as-a-Service, що дозволяє забезпечити масштабовану взаємодію з моделлю через API.

Надійність і стабільність усієї платформи штучного інтелекту значною мірою залежить від впровадження безперервного моніторингу. Це охоплює контроль ключових метрик продуктивності моделі, стану інфраструктури та якості даних, що надходять до системи. Особливу роль у підтримці працездатності платформи відіграє логування, яке забезпечує відстежування взаємозв'язку між обробленими даними та моделями, які їх використовують. Постійний зворотній зв'язок між компонентами системи створює динамічне середовище, в якому оновлення даних та перенавчання моделей можуть здійснюватися автоматично, що є однією з ключових переваг сучасних рішень у рамках концепції MLOps [7].

Апаратна реалізація архітектури платформи. Апаратна реалізація платформи базується на поєднанні хмарних обчислень (Cloud) та периферійних (Edge) пристроїв. Центральним елементом такої системи виступає обрана топологія, яка визначає спосіб взаємодії між обчислювальними вузлами. Найбільш доцільним для гібридної архітектури вважається використання ієрархічної структури типу дерева, що дозволяє ефективно організувати обмін даними між центральними хмарними вузлами та локальними вбудованими пристроями.

Зокрема, топологія бінарного дерева є однією з базових у комп'ютерних науках і активно застосовується в алгоритміці, статистиці та теорії графів, Рис.1. У такій структурі кожен вузол може мати максимум два дочірні вузли — лівий та правий. Кожен із них, у свою чергу, може бути або листовим (кінцевим), або внутрішнім вузлом, що продовжує ієрархію. Ця структура є зручною для організації обчислень, які вимагають розподіленої обробки даних між центральною платформою та периферійними пристроями, зберігаючи при цьому логічну впорядкованість і керованість системи [7,8].

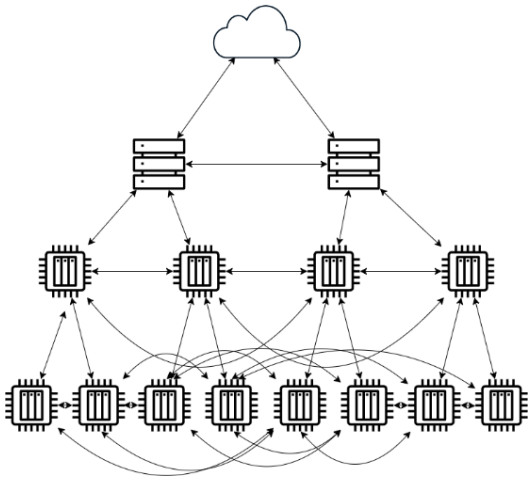

Рис. 1. Топологія побудови платформи штучного інтелекту на основі дерева з Де Брайнівськими зв'язками

Особливістю поданого бінарного дерева є відносно низька надійність, коли при одній відмові, можливий фактичний розподіл топології на дві частини. Для мінімізації вказаної проблеми є доцільним застосування технології Де Брайна по формуванню додаткових зав'язків [8].

Модель формування таких зав'язків подамо в наступному вигляді:

$$D_{(n,k)} = kD_{(n-1,k)} + (-1)^k * (k-1) * S_{n-1,k-1} \quad (1)$$

Де *D(n,k)* – значення Де Брайна, *S(n,k)* – значення функції Стерлінга другого роду, n - кількість елементів, k - кількість циклів, *(k/j)* - біноміальний коефіцієнт "k по j" (кількість способів вибрати *j* елементів з *k* без урахування порядку), а $(-1)^m$ – сума альтернативно збігу.

Перший етап тесту проводиться з метою визначення швидкості розгортання системи при довільних значень кількості вузлів. Одержані результати, табл.1, Рис.2, є основою оцінки можливостей запропонованої платформи до обробки запитів за визначені часові терміни при збільшенні числа залучених пристроїв, та про можливості масштабування платформи на основі запропованого методу [8,9].

Другий етап тесту проводиться з метою оцінки можливостей програмно-апаратного комплексу до обробки великих масивів запитів та підтвердження придатності платформи до ефективної роботи у реальних умовах, Тестовими умовами приймемо один запит від одного вузла при

забезпеченні встановленого рівня масштабованості та надійності. В процесі тестування прийнято, що кожен потік робить 100 запитів на платформу. Параметрами, значення яких будуть аналізуватися, прийнято середній (Tavg), максимальний (Tmax) та мінімальний (Tmin) час на реагування, значення 99-го персентиля та стандартне відхилення для різних типів мереж передачі даних.

Таблиця 1

Часові показники формування програмно- апаратного комплексу відносно кількості залучених вузлів поданої топології

Кількість пристроїв	Час розгортання, секунд
2	915.25
6	967.06
15	1102.26
29	1212.27
62	1349.04

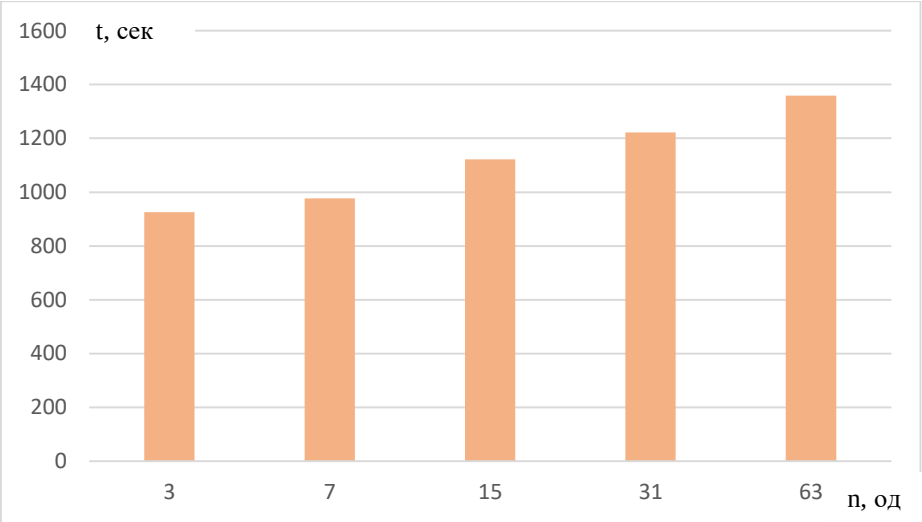

Рис. 2. Залежність часу формування програмно- апаратного комплексу від кількості залучених вузлів

В Таб. 2 представлені результати другого етапу оцінки можливостей програмно-апаратного комплексу до функціонування в умовах навантаження різної інтенсивності. Під час оцінки можливостей комплекс працював без відмов і не спостерігалося перенасичення черги задачі. Також було показано, що наявність в топології побудови додаткових вузлів приводить до значної мінімізації середнього часу відповіді.

Формування топології платформи з включенням контейнерованих моделей штучного інтелекту та вузлів, що забезпечують баланс навантаження між різними контейнерованими моделями штучного інтелекту забезпечує зменшення кількості залучених пристрої, зменшення часу обробки запитів та в цілому забезпечить високу продуктивність системи.

Даний висновок ґрунтується на даних моделювання, які подані на Рис.4.

Подані на них залежності дають змогу оцінити показники динаміки швидкості реагування платформи, а також відображають взаємодію між вузлами перерозподілу навантаження (воркерами) та контейнерами моделей, що підтверджує можливість даної топології до застосування в умовах реального навантаження [9-11].

Висновки

1. В роботі подані результати розробки модель побудови платформи штучного інтелекту при умові мінімізації часу розгортання та забезпечення надійності роботи в ході моніторингу корпоративного та аномального трафіку інформаційно-комунікаційної мережі передачі даних. При створені моделі застосовано технологію MLOps, чим показано, що її використання дозволяє створити ефективний спосіб розгортання платформ штучного інтелекту.

2. Подана модель побудови платформа має можливість до автоматизації та моніторингу етапів збору даних, перетворення, навчання, впровадження та представлення результатів кінцевому користувачеві. Етап розгортання платформи забезпечує швидку інтеграції, мінімізацію часу розгортання інтеграцію та надійність функціонування моделі з постійним її моніторингом.

3. Висвітлені в роботі результати оцінки ефективності застосування розробленої моделі платформи штучного інтелекту що при розвитку топології платформи, від 3 до 63 вузлів час розгортання збільшується лише на 434 секунди, що свідчить про здатність платформи до адаптації та розвитку.

Тестування навантаженням показала, що подана модель платформи здатна функціонувати без відмов, за умови навантаження черги запитів від 25 потоків одночасно, при цьому застосування додаткових вузлів забезпечує мінімізацію середнього часу відповіді системи.

Таблиця 2

Параметри реагування тестового програмно-апаратного комплексу при зміні навантаження

Tavg	Tmax	T_{99}	Tmin	SD	Потоки
MobileNet					
929.49	12472.76	827.44	41.24	1836.23	1
954.92	12976.17	938.56	42.05	1845.24	2
1021.02	11299.37	1006.42	42.65	2058.29	5
1298.33	15127.89	1188.00	44.77	2637.24	10
1527.38	29245.45	1655.83	46.78	3409.68	25
MobileNet Raspberry PI Tree-Debrujin (2 ступінь)					
2518	47033	1647.9	62.511	4007.8	1
2028.1	18511	2452.49	58.977	4340.7	2
2512.8	27766	3245.07	61.99	5601.6	5
3811.6	46171	3862.83	63.911	8075.5	10
4301.8	73503	4851.08	65.674	9665.7	25
MobileNet Raspberry PI Tree-Debrujin (2 ступінь, 2 воркери)					
507.67	1366.38	397.85	41.85	1035.63	1
527.38	13767.49	554.74	41.72	1054.81	2
597.83	11143.9	487.76	42.92	1052.85	5
849.44	10847.87	890.73	46.84	1453.46	10
871.87	15597.56	909.74	44.49	1684.74	25

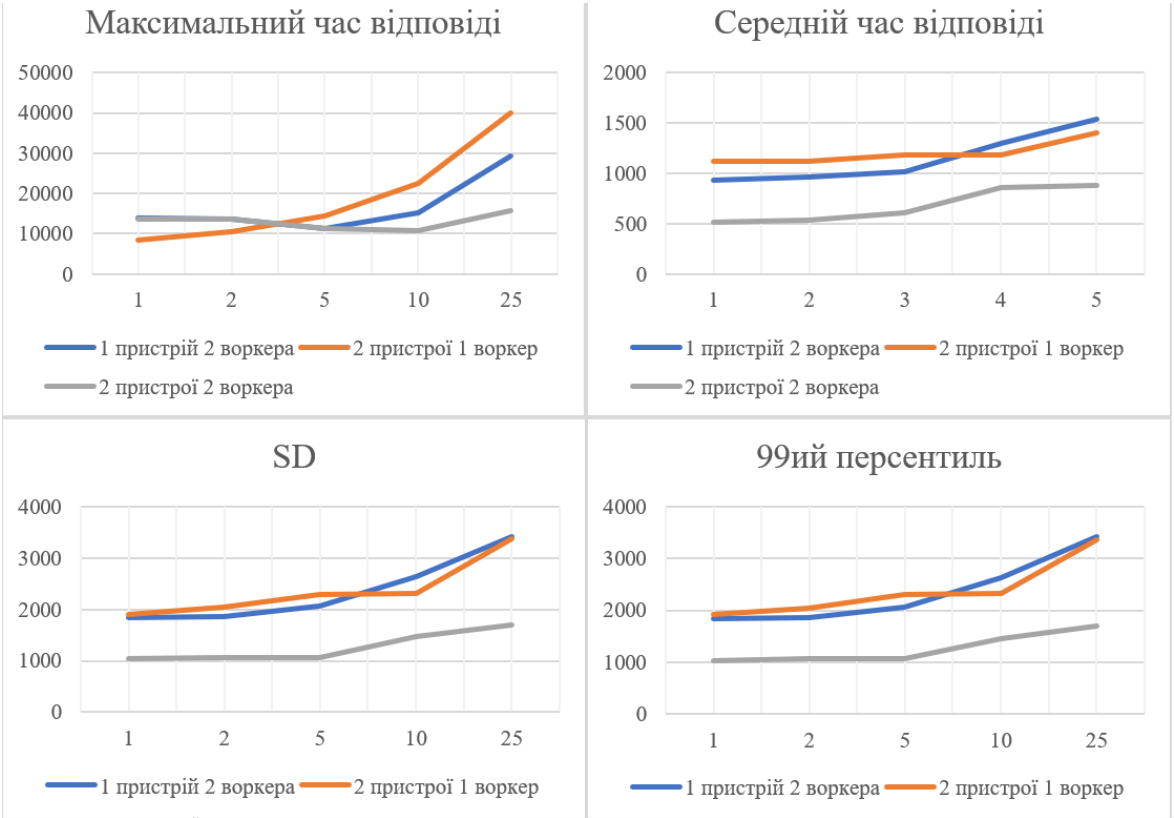

Рис. 4. Порівняльні графіки характеристик тестування системи на навантаження

Література

1. Кондакова, А., & Корецький, О. (2024). Моделі побудови мереж інтернету речей для управління інфраструктурою міста. *Smart Technologies*, 2(15), 124–132. DOI: https://doi.org/10.32347/st.2024.2.1901

2. Плечистий, Д. & Сітайло, М. (2024). Дослідження застосування автоматизованого машинного навчання для порівняльного аналізу методів прогнозування курсу криптовалют. Технічна інженерія. 218-224. 10.26642/ten-2024-1(93)-218-224.

3. Machine Learning Pipeline. Data science. Available online: https://itwiki.dev/data-science/ml-reference/ml-glossary/machine-learning-pipeline.

4. M. Safdar, P. P. Paul, G. Lamouche, G. Wood, *at. ets*. (2024) Fundamental requirements of a machine learning operations platform for industrial metal additive manufacturing, Computers in Industry, (154), 104037. https://doi.org/10.1016/j.compind.2023.104037.

5. Chintan, N. Ashwini. (2025). An analysis of the challenges in the adoption of MLOps. Journal of Innovation & Knowledge. (10) 100637. https://doi.org/10.1016/j.jik.2024.100637.

6. S. Karthik, H. & Ghadi, M. Antonio. (2025). Scalability and Maintainability Challenges and Solutions in Machine Learning: Systematic Literature Review. https://doi.org/10.48550/arXiv.2504.11079.

7. D. Alsagheer, L. Xu & W. Shi, "Decentralized Machine Learning Governance: Overview, Opportunities, and Challenges," in *IEEE Access*, vol. 11, pp. 96718-96732, 2023, doi: 10.1109/ACCESS.2023.3311713.

8. S. J. Warnett & U. Zdun, "Bridging the Gap Between MLOps and RLOps: An Industry 4.0 Case Study on Architectural Design Decisions in Practice," *2025 IEEE 22nd International Conference on Software Architecture (ICSA)*, Odense, Denmark, 2025, pp. 232-242, doi: 10.1109/ICSA65012.2025.00031.

9. L. Su & T. Storer, "A Case Study of DevOps Adoption within a Large Financial Organisation," in 2023 IEEE International Conference on Software Maintenance and Evolution (ICSME), Bogotá, Colombia, 2023, pp. 403-413, doi: 10.1109/ICSME58846.2023.00053.

10. P. Woźniak, M. Milczarek & J. Woźniak, "MLOps Components, Tools, Process, and Metrics: A Systematic Literature Review," in *IEEE Access*, vol. 13, pp. 22166-22175, 2025, doi: 10.1109/ACCESS.2025.3534990.

11. Семенко, А. І., Бойко, Ю. М., *et al*. (2024). Сучасні технології інфокомунікаційних та комп'ютерних мереж: Монографія (А. І. Семенко, Ред.). Європейський університет, ФО-П Білецький Р. Г., 557 с. https://elar.khmnu.edu.ua/handle/123456789/17381.

References

1. Kondakova, A., & Koretsky, O. (2024). Models for building Internet of Things networks for city infrastructure management. Smart Technologies, 2(15), 124–132. DOI: https://doi.org/10.32347/st.2024.2.1901.

2. Plechysty, Dmytro & Sitailo, Maksym. (2024). Research on the application of automated machine learning for comparative analysis of cryptocurrency exchange rate forecasting methods. Technical Engineering. 218-224. 10.26642/ten-2024-1(93)-218-224.

3. Machine Learning Pipeline. Data science. Available online: https://itwiki.dev/data-science/ml-reference/ml-glossary/machine-learning-pipeline.

4. M. Safdar, P. P. Paul, G. Lamouche, G. Wood, *at. ets*. (2024) Fundamental requirements of a machine learning operations platform for industrial metal additive manufacturing, Computers in Industry, (154), 104037. https://doi.org/10.1016/j.compind.2023.104037.

5. Chintan, N. Ashwini. (2025). An analysis of the challenges in the adoption of MLOps. Journal of Innovation & Knowledge. (10) 100637. https://doi.org/10.1016/j.jik.2024.100637.

6. S. Karthik, H. & Ghadi, M. Antonio. (2025). Scalability and Maintainability Challenges and Solutions in Machine Learning: Systematic Literature Review. https://doi.org/10.48550/arXiv.2504.11079.

7. D. Alsagheer, L. Xu & W. Shi, "Decentralized Machine Learning Governance: Overview, Opportunities, and Challenges," in IEEE Access, vol. 11, pp. 96718-96732, 2023, doi: 10.1109/ACCESS.2023.3311713.

8. S. J. Warnett & U. Zdun, "Bridging the Gap Between MLOps and RLOps: An Industry 4.0 Case Study on Architectural Design Decisions in Practice," 2025 IEEE 22nd International Conference on Software Architecture (ICSA), Odense, Denmark, 2025, pp. 232-242, doi: 10.1109/ICSA65012.2025.00031.

9. L. Su & T. Storer, "A Case Study of DevOps Adoption within a Large Financial Organization," in 2023 IEEE International Conference on Software Maintenance and Evolution (ICSME), Bogotá, Colombia, 2023, pp. 403-413, doi: 10.1109/ICSME58846.2023.00053.

10. P. Woźniak, M. Milczarek & J. Woźniak, "MLOps Components, Tools, Process, and Metrics: A Systematic Literature Review," in IEEE Access, vol. 13, pp. 22166-22175, 2025, doi: 10.1109/ACCESS.2025.3534990.

11. Semenko, A. I., Boiko, J. M., *et al*. (2024). Suchasni tekhnolohii infokomunikatsiinykh ta kompiuternykh merezh: Monohrafiia (A. I. Semenko, Red.). European University, FO-P Biletskyi R. H., 557 s. https://elar.khmnu.edu.ua/handle/123456789/17381.