https://orcid.org/0000-0002-9639-3544
https://orcid.org/0000-0001-9600-2518
https://orcid.org/0000-0002-1268-7837

УДК 004.4`2

О.Є. КОВАЛЕНКО[*], О.С. КОПЧА[*], В.М. СМОЛІЙ[*]

ОСОБЛИВОСТІ ЗАСТОСУВАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В СИСТЕМАХ ІНТЕРНЕТУ РЕЧЕЙ

[*]Національний університет біоресурсів і природокористування України, м. Київ, Україна

Анотація. *У статті проведено аналіз особливостей застосування штучного інтелекту (ШІ) для обробки BigData (BD) у системах IoT (BD-IoT). Досліджено побудову сучасних багаторівневих систем IoT, проведено порівняння обробки BD в традиційних хмарних сервісах та в граничній області IoT. Показано можливості використання новітніх мережевих стандартів стільникового зв'язку для побудови систем IoT. Досліджено сучасні методи для систем виявлення вторгнень у мережу (NIDS), що дозволяють оптимізувати аналіз трафіку та підвищити рівень безпеки в системах IoT на основі обробки BD. Також було розглянуто підхід до TinyML для розподілення систем обробки між пристроями задля дослідження доцільності використання у системах BD-IoT. У результаті проведеного аналізу встановлено, що перехід від традиційної хмарної архітектури до розподілених архітектур з використанням ШІ сприяє горизонтальній масштабованості мереж BD-IoT та мінімізує навантаження на хмару в умовах обмежених обчислювальних та мережевих ресурсів. Обґрунтовано, що використання стандартів 5G та 6G є необхідною вимогою для масштабування систем BD-IoT великою кількістю сенсорів і забезпечення стабільної роботи в умовах географічного поширення на основі збільшення пропускної здатності мереж із використанням ШІ. Визначено доцільність розподілу функцій NIDS між різними рівнями BD-IoT-системи відповідно до обсягу даних та необхідної швидкості реакції при застосуванні ШІ. Використання вибору ознак на рівні граничних обчислень забезпечує високу швидкість обробки критичних даних та інтерпретацію результатів, тоді як вилучення ознак у хмарному середовищі дозволяє ідентифікувати складні приховані атаки на великих масивах даних. Показано, що використання підходів TinyML є перспективним напрямом розподілу обчислювального навантаження у BD-IoT-системах. Впровадження моделей машинного навчання безпосередньо на рівні кінцевих пристроїв дозволяє суттєво мінімізувати обсяги переданих даних до хмарної інфраструктури, підвищуючи при цьому енергоефективність системи та забезпечуючи можливість прийняття критичних рішень у режимі реального часу.*

Ключові слова: *інтернет речей, великі дані, штучний інтелект, машинне навчання, TinyML.*

Abstract. *The article analyzes the features of the application of artificial intelligence (AI) for Big Data (BD) processing in IoT (Internet of Things) systems (BD-IoT). The construction of modern multi-tier IoT systems has been studied, and a comparison of BD processing in traditional cloud services and at the IoT edge has been carried out. The possibilities of using the latest cellular network standards for building IoT systems are shown. Modern network intrusion detection methods (NIDS) are studied, enabling optimization of traffic analysis and increasing the level of security in IoT systems based on BD network processing. The TinyML approach for distributing system processing between devices has also been considered to evaluate its applicability in BD-IoT-systems. The analysis has shown that the transition from traditional cloud architecture to distributed ones using AI increases the horizontal scalability of the BD-IoT network and minimizes the load on the cloud in conditions of limited computing and network resources. It is substantiated that the use of 5G and 6G standards is a necessary requirement for scaling a BD-IoT-system with a large number of sensors and ensuring stable operation in conditions of geographical expansion*

ISSN 1028-9763. Математичні машини і системи. 2026. № 2

based on increasing network bandwidth using AI. The feasibility of distributing NIDS functions between higher levels of BD-IoT-systems is determined, in accordance with the volume of data and the required speed of response when using AI. The use of feature selection at the edge computing levels provides high speed processing of critical data and interpretation results, while feature extraction in cloud environments allows the identification of complex concealed attacks on large data sets. It is shown that the use of TinyML approaches is a promising direction for distributing computational load in BD-IoT-systems. The implementation of machine learning models at the end-device level significantly reduces the amount of data transmitted to the cloud infrastructure, increases the energy efficiency of the system, and enables making critical decisions in real time.
Keywords: *Internet of Things, big data, artificial intelligence, machine learning, TinyM.*

DOI: 10.34121/1028-9763-2026-2-3-13

1. Вступ

З кожним роком зростає кількість систем IoT та число пристроїв, підключених до них. Очікується, що до 2030 року буде підключено 29 мільярдів пристроїв [1]. Традиційна хмарна архітектура часто перестає бути ефективною при великих масивах даних, що зумовлює перехід до нових, більш розподілених парадигм, які переносять частину обчислень на сенсори та проміжні вузли системи, тим самим забезпечуючи оптимальне горизонтальне масштабування. Також зі збільшенням обсягу даних зростає латентність на події при класичній хмарній архітектурі. Така кількість пристроїв генерує великі об'єми даних, через що багато систем IoT переходять до категорії BD-IoT. Пропускна здатність мереж обмежена, а використання апаратного ресурсу для обробки такої кількості даних потребує значних фінансових затрат. Різниця між потенційними потребами для функціонування систем великих даних інтернету речей (BD-IoT) та фактичними апаратними й мережевими ресурсами зумовлює потребу в застосуванні методів штучного інтелекту (ШІ) для оптимізації обробки великих даних у системах інтернету речей як критично важливу задачу. Це зумовлює актуальність у дослідженні застосування штучного інтелекту для обробки великих масивів даних.

Пристрої інтернету речей безперервно генерують величезну кількість даних. Зі збільшенням кількості пристроїв, підключених до екосистеми інтернету речей, обсяг даних, які вони генерують, зростає експоненціально. Це обумовлює потребу у застосуванні методів обробки великих даних для покращення аналізу даних, але також створює проблеми для їх зберігання та управління.

Пристрої інтернету речей збирають різноманітні типи даних, від числових показників у промислових машинах до аудіо- та відеоданих в інтелектуальних системах безпеки. Таке різноманіття збагачує великі дані, даючи змогу отримувати ціннішу аналітику. Воно також додає складності обробці та аналізу даних, вимагаючи більш просунутих інструментів та методів для ефективної інтеграції та обробки даних [2].

Пристрої інтернету речей, особливо пристрої інтернету речей з технологією 5G, часто передають дані в режимі реального часу або майже в режимі реального часу. Така висока швидкість генерації даних означає, що обробка великих даних повинна забезпечувати отримання своєчасної аналітики. Це має вирішальне значення для застосувань, де необхідний негайний аналіз даних, наприклад, у системах реагування на надзвичайні ситуації або моніторингу дорожнього руху в режимі реального часу.

Взаємозв'язок інтернету речей та великих даних пропонує численні переваги для різних секторів. Таке поєднання дозволяє проводити масштабний збір даних, аналіз та отримання аналітичних даних. Ці аналітичні дані можна використовувати для підвищення ефективності, покращення процесу прийняття рішень та оптимізації операцій. Вони також можуть виявляти нові можливості для інновацій, стимулюючи зростання бізнесу в різних галузях.

Зв’язок інтернету речей та великих даних також може бути пов’язаний із труднощами, особливо з ризиками, відносно інтернету речей. Зі збором та аналізом величезних обсягів конфіденційних даних зростає ризик витоків даних та кібератак.

Це підкреслює необхідність безпечних практик інтернету речей. Впроваджуючи надійні протоколи безпеки та постійно забезпечуючи моніторинг мережі інтернету речей, користувачі можуть захиститися від несанкціонованого доступу та забезпечити конфіденційність даних. Це дозволяє організаціям використовувати переваги інтернету речей та великих даних для впровадження інновацій і підвищення ефективності, водночас знижуючи рівень ризиків.

Метою статті є комплексний аналіз особливостей застосування ШІ в системах IoT з урахуванням обробки великих даних.

2. Архітектура IoT

Архітектура інтернету речей (IoT) — це багаторівнева структура, що представляється 3–6 рівнями в залежності від розподілу функцій. Вона керує потоком даних від датчиків (сенсорів) до хмари (мережевий рівень) для аналізу, що дозволяє автоматизувати процес. Загальні моделі зосереджені на інтеграції апаратного забезпечення (датчиків, шлюзів) із програмним забезпеченням (аналітика, додатки). Архітектура інтернету речей визначає, як датчики, пристрої, мережі та програми взаємодіють та інтегруються один з одним у рамках IoT-рішення, щоб забезпечити ефективний та безпечний збір, передачу та обробку даних для отримання аналітики в режимі реального часу. У стандарті ITU_T Y.4000/Y.2060 [3] описується чотирирівнева еталонна архітектура (рис. 1).

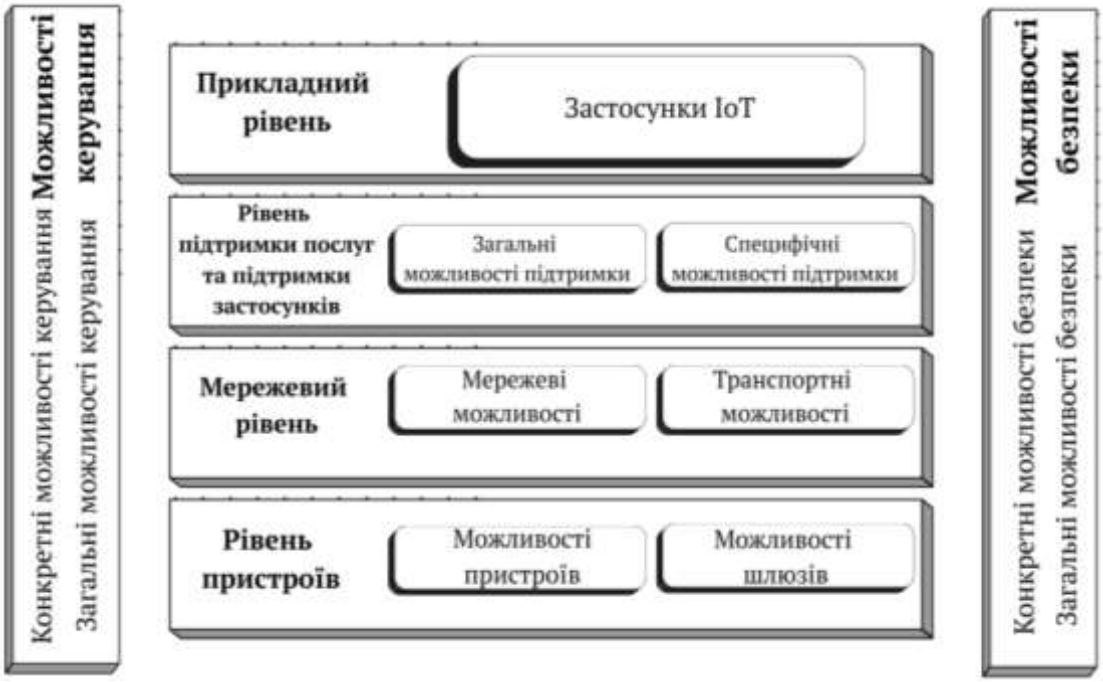

Рисунок 1 — Еталонна архітектура IoT за стандартом ITU_T Y.4000/Y.2060

Кожен рівень IoT орієнтований на виконання відповідних функцій.

1. Рівень пристроїв (сприйняття) забезпечує збирання даних реального світу, як-от температура, вологість, рух, освітлення, тиск тощо за допомогою датчиків. Для впливу на керовані середовища/об’єкти використовуються актуатори (виконавчі механізми) для виконання фізичних дій, як-то увімкнення/вимкнення пристроїв або налаштування парамет-

рів системи. Також ідентифікує та відстежує фізичні об’єкти за допомогою технологій, як-от RFID та NFC.

2. Мережевий (транспортний) рівень передає зняті дані з пристроїв інтернету речей до шлюзів, граничних вузлів або хмарних платформ. Підтримує надійний зв’язок на основі дротових та бездротових технологій, як-от Wi-Fi, Bluetooth, Zigbee, та стільникових мереж із використанням протоколів зв’язку для забезпечення ефективного й безпечного обміну даними між пристроями та системами.

3. Рівень підтримки застосунків та сервісів (проміжного програмного забезпечення) фільтрує, аналізує та зберігає вхідні дані для подальшого використання й прийняття рішень. На цьому рівні здійснюється керування підключеними пристроями, забезпечується сумісність та підтримується масштабованість розгортання систем інтернету речей. Також тут забезпечуються цілісність даних, автентифікація та безпечний доступ до сервісів інтернету речей.

4. Рівень застосунків надає панелі інструментів, мобільні додатки або веб-інтерфейси для моніторингу та керування системами інтернету речей. На цьому рівні оброблені дані представляються у змістовній формі для підтримки аналізу та прийняття рішень. Також тут реалізуються специфічні рішення у сфері інтелектуалізації середовища проживання, охорони здоров’я, промислової автоматизації та «розумних» міст.

Розробка надійної та ефективної архітектури інтернету речей вимагає дотримання стратегічного підходу, який поєднує масштабованість, сумісність, безпеку та економічну ефективність. Впровадження та правильне налаштування системи інтернету речей відповідно до конкретних потреб забезпечує збір та обробку даних щодо цільових потреб і дозволяє приймати обґрунтовані рішення на основі даних.

3. Великі дані в IoT

У сучасних системах IoT обробка даних може розподілятись між всіма рівнями з фокусом на специфіку відповідного рівня. Великі дані (BigData, BD) в інтернеті речей (IoT) стосуються масивних, високошвидкісних потоків даних, що генеруються підключеними датчиками та пристроями (IoT), які потребують розширеної аналітики для отримання практичних висновків. Вони перетворюють необроблені дані — від простих оновлень статусу до складних даних від датчиків — на результати прогнозної, прескриптивної аналітики для оптимізації операцій, покращення прогнозного обслуговування та підвищення ефективності в секторах, як-то виробництво, логістика та «розумні міста». Приклади джерел BD в IoT (BD-IoT) наведені на рис. 2.

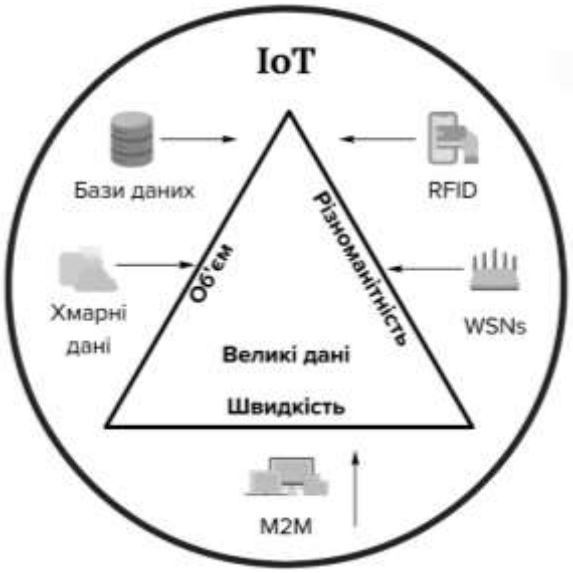

Рисунок 2 — Джерела BD в IoT [4]

Модель 5V великих даних — Volume, Velocity, Variety, Veracity and Value (обсяг, швидкість, різноманітність, достовірність та цінність) — визначає характеристики та проблеми управління масивними складними наборами даних. Вона допомагає організаціям вийти за рамки простого зберігання даних та отримати практичну інформацію для отримання конкурентної переваги. У контексті систем інтернету речей модель вимірювання продуктивності 5V має такі особливості.

1. Обсяг (volume) визначає обсяги даних. Системи BD-IoT генерують великі обсяги даних, враховуючи майже безкінечне потенційне горизонтальне розширення за рахунок підключення нових пристроїв до мережі Вектор оптимізації: імплементація методів локальної агрегації та стиснення на рівні периферійних шлюзів, оскільки передача всього об’єму даних на рівень хмари є нераціональною у випадку роботи з великими системами BD-IoT.

2. Швидкість (velocity) визначає інтенсивність генерування інформаційних потоків, а також вимоги до оперативності їх обробки. Динаміка надходження даних у системах IoT значно вища в порівнянні з традиційними системами обробки великих об’ємів даних. Як зазначалось раніше, у медичній та в низці деяких інших предметних областей критично важливо мінімізувати затримки під час обробки даних. Вектор оптимізації: залучення методів попередньої фільтрації даних на граничному рівні задля зниження обсягу паралельних транзакцій, що спрямовуються до хмарного середовища. Перенесення обчислювальних процесів для аналізу показників сенсорів на цей рівень мінімізує затримку.

3. Різноманітність (variety) відображає структурну гетерогенність інформації. За ознакою своєї структури в екосистемі BD-IoT дані бувають трьох видів: структуровані (як-то табличні показники температури), напівструктуровані (наприклад. формати обміну даними XML або JSON) та неструктуровані (потоки аудіо й відео з камер спостереження). Вектор оптимізації: з метою приведення інформації різних типів до уніфікованого формату перед початком глибокого аналізу необхідне розгортання гнучких NoSQL-рішень, а також стандартизованих протоколів адаптації на рівні шлюзів.

4. Достовірність (veracity). У сенсорних мережах неточність, ненадійність є доволі поширеним явищем із різних причин. До найпоширеніших належать: апаратні похибки неякісних датчиків, втрата пакетів у бездротових каналах зв'язку, зовнішні втручання. Вектор оптимізації: обробка хибних даних витрачає обмежені обчислювальні ресурси всієї мережі. З огляду на це, найбільш перспективним рішенням є розміщення запобіжників на якомога нижчих етапах, як-то рівень пристроїв чи граничний рівень. Роль запобіжників виконують алгоритми виявлення аномалій, фільтрації викидів та очищення даних тощо.

5. Цінність (value). Великі дані часто містять низьку «щільність цінності» (low value density): окремо взяте значення, згенероване сенсором (наприклад, одиничний запис про вібрацію), не має жодної практичної користі без решти інформації. Однак аналіз великих масивів таких даних у динаміці в своїй сукупності має надзвичайно високу цінність (наприклад, передбачення виходу обладнання з ладу). Вектор оптимізації: ключове завдання інтелектуальної обробки в BD-IoT полягає в тому, щоб за допомогою агрегування даних на рівні кінцевих пристроїв та граничних вузлів із застосуванням алгоритмів машинного навчання зберегти критично важливу інформацію в оптимізованому форматі.

Обробка BD на рівні хмари може призводити до виникнення критичних затримок та нераціонального використання трафіку. Частково ці проблеми можна вирішити перенесенням частини обробки даних на рівень граничних обчислень. У табл. 1 наведено порівняння особливостей обробки даних у хмарі та у граничній області IoT.

З табл. 1 можна зробити висновок, що при перенесенні обчислень у граничну область слід враховувати обмеженість обчислювальних ресурсів та можливе високе енергоспоживання проміжними мережевими пристроями, як-то маршрутизатори, шлюзи. Тому розподіл функцій обробки даних повинен ґрунтуватись на пошуку компромісу між наявними ресурсами відповідного рівня системи, з одного боку, та потрібною функціональністю з іншого.

Таблиця 1 — Особливості обробки даних у хмарі та у граничній області IoT

Критерій порівняння	Традиційна хмарна архітектура (cloud-only)	Архітектура з граничними обчисленнями (edge-cloud continuum)
Локалізація обробки	Централізована (у віддалених ЦОД)	Гібридна (розподілена між edge та cloud)
Затримка	Висока, залежить від мережі	Наднизька, обробка в реальному часі за потреби
Використання мережі	Високе	Оптимізоване
Ризики безпеки	Ризик перехоплення при передачі даних	Ризик перехоплення при передачі даних та потреба у захисті додаткових точок входу
Автономність	Низька (не працює без інтернету)	Висока, частина функціонала зберігається без інтернету

4. Використання стільникових мереж для побудови BD-IoT-систем

BD-IoT-системи, де сенсори та вузли знаходяться на великій відстані один від одного, використовують стільникові мережі. Найбільший розвиток побудова мережевого рівня за допомогою стільникових мереж отримала після впровадження п'ятого покоління мобільних мереж. Стандарт 5G впровадив низку нових сценаріїв використання, частина з яких закрила критичні прогалини, які заважали побудові надійних BD-IoT [5, 6]:

• mMTC (masive machine type communication) — сценарій використання, який дозволяє збільшити одночасну кількість пристроїв, підключених до мережі шляхом розширення топології мережі та впровадження оновлених алгоритмів.

• eMBB (enhanced mobile broadband) — сценарій використання, який дозволяє передачу даних на гігабітній швидкості.

• URLLC (ultra reliable low latency communications) — сценарій використання, який значно збільшує надійність мережі та одночасно з цим зменшує час затримки для запитів.

Нові сценарії використання дозволяють системам, які оперують мільйонами сенсорів на великих площах, відповідати критеріям моделі 3V (Volume, Velocity, Variety) (рис. 2).

MMTC забезпечує характеристику Volume та відповідає за пришвидшення обробки даних, які генерують сенсори. eMBB та URLLC відповідають за характеристику Velocity та дозволяють оброблювати дані й надавати відповіді на достатній швидкості. Завдяки мережевій архітектурі slicing, 5G може ефективно одночасно передавати як об'ємні дані (відео з сенсорів, звукові записи та ін.), так і невеликі дані (наприклад, команди для сенсорів), реалізуючи у такий спосіб Variety.

Мережі 5G наразі є найкращим варіантом для реалізації мережевого рівня для BD-IoT, які мають велику кількість сенсорів, розташованих на великий площі покриття. У перспективних мережах 6G покращення параметрів зв'язку буде забезпечуватись у тому числі за рахунок використання ШІ [7]. Насамперед, системи інтернету речей, які не використовують таку кількість трафіку, можуть залишатися на мережах минулого покоління. Порівняльний аналіз характеристик сучасних і перспективних мереж стільникового зв'язку, проведений на основі джерел [7–10], наведено у табл. 2.

Таблиця 2 — Порівняння мереж 4G, 5G та 6G

Характеристика	Мережа 4G	Мережа 5G	Мережа 6G
Максимальна швидкість	1 Гбіт/с	20 Гбіт/с	100 Гбіт/с
Кількість активних пристроїв на квадратний кілометр	100 000	До 1 000 000	До 10 000 000
Затримка сигналу	10 мс	1 мс	0,1 мс
Відсоток втрати пакетів	Між 0,1 та 1 %	0,001 %	<0,001 %

5. ШІ в IoT

Три типи аналітики BD-IoT на основі ШІ призначені для різних типів задач [11]. Прогнозна аналітика фокусується на аналізі попередніх даних для формування передбачення явищ (зазвичай використовується для технічного обслуговування). Прескриптивна аналітика пропонує розуміння того, як діяти на основі спостережуваних даних. Вона виходить за рамки прогнозування (прогнозний штучний інтелект), оптимізуючи рішення, пропонуючи, як впоратися з обмеженнями та майбутніми сценаріями для підвищення операційної ефективності, як-от управління запасами або логістика ланцюга поставок. Моніторинг у режимі реального часу використовується для відстеження статусу та місцезнаходження контрольованого середовища/об’єкта для швидкого реагування.

Використання алгоритмів штучного інтелекту та машинного навчання забезпечує виявлення прихованих закономірностей, прогнозування поведінки системи та прийняття стратегічних рішень [2]. Використання ШІ на граничному рівні потребує врахування ресурсних обмежень, тоді як хмарна інфраструктура пропонує майже необмежену масштабованість [12]. Хмарні сервіси дозволяють застосовувати ресурсоємні моделі глибокого навчання та обробки великих даних [13].

Існують пристрої граничного рівня, які підтримують алгоритми AI чи ML. Їх завдання полягає в тому, щоб самостійно проводити первинну обробку даних із можливою подальшою їх фільтрацією. Завдяки цій роботі, кількість навантаження на мережу суттєво зменшується [13, 14].

Завдання мережевого рівня полягає в інтеграції різних комунікаційних протоколів, як-то MQTT, CoAP, і технологій передачі даних, зокрема LPWAN, 5G і перспективні мережі 6G [15]. На мережевому рівні використання алгоритмів ШІ дозволяє оптимізувати використання ресурсів мережі.

5.1. ШІ на граничному рівні

Edge AI — це розгортання алгоритмів машинного навчання безпосередньо на локальних пристроях, як-то датчики інтернету речей, смартфони чи камери, замість того, щоб покладатися на централізовані хмарні сервери. Це дозволяє обробляти дані безпосередньо у джерелі, забезпечуючи прийняття рішень у режимі реального часу з низькою затримкою, покращену конфіденційність та функціональність без підключення до інтернету. Хоча хмарний ШІ підходить для навчання моделей на величезних наборах даних, граничний ШІ ідеально підходить для запуску (виведення) цих моделей на живих даних на граничному рівні IoT. Багато рішень використовують гібридний підхід, коли модель працює на пери-

ферії, але періодично надсилає дані назад у хмару, наприклад, для перенавчання та вдосконалення [12].

TinyML — це розділ (підмножина методів) машинного навчання, який реалізує концепцію створення та впровадження моделей машинного навчання на малопотужних мікроконтролерах з обмеженими ресурсами, як-то Arduino. Моделі машинного навчання вимагають значної обчислювальної потужності. TinyML — це технологія розгортання моделей машинного навчання на рівні пристроїв. Найбільший фактор, який відрізняє TinyML від класичного ML, це екстремальна обмеженість у ресурсах та висока ефективність. Середньостатистичний пристрій, на якому розгортається TinyML, має менше 256 кілобайт оперативної пам'яті та споживання менше 1 мВт [16].

Основними перевагами використання TinyML є [16]:

1. Перенос певної частини обчислень на рівень пристроїв знижує вимоги до обчислювальних навантажень інших рівнів системи, які йдуть далі в обробці даних. Реагування на рівні сенсорів дає найбільшу швидкість реакції у критичних ситуаціях.

2. Первинна обробка даних на рівні пристроїв дозволяє захистити чутливі дані, відправляючи на рівні вище вже де-персоніфіковані дані, тим самим дозволяючи легше проходити вимоги до конфіденційності за стандартами GDPR та HIPAA.

3. Можливість працювати в режимі офлайн, зберігаючи певну частину функціонала в умовах відсутнього чи нестабільного з'єднання.

4. Компактний розмір моделей (у середньому до 100 кілобайт) дозволяє легко оновлювати сенсори за мережевим протоколом.

Ключовим недоліком використання технології TinyML є децентралізація обчислювальних процесів, що призводить до збільшення складності системи у порівнянні з архітектурами, де виконується обробка даних лише на рівні хмари та граничних обчислень. Також потреба в оновленні моделей даних на пристроях збільшує поверхню атаки на систему, що вимагає впровадження складніших криптографічних алгоритмів для захисту. Також BD-IOT-системи мають більше точок вразливості.

Технологія TinyML покращує ефективність роботи систем BD-IoT. Можливість переносу певної розрахункової вимоги на рівень пристроїв полегшує навантаження на всі наступні рівні, дозволяючи більш горизонтальне розширення. Технологія фільтрації даних на рівні пристрою дає змогу зменшити загальний об'єм трафіку, що оптимізує роботу мережевого рівня. Технологія TinyML надає BD-IoT-системам здатність до автономної роботи, що є особливо важливим у контексті критичного обладнання, де час простою системи може призвести до тяжких наслідків.

Фреймворки TinyML забезпечують надійну й ефективну інфраструктуру, яка дозволяє організаціям та розробникам ефективно використовувати свої дані та розгортати передові алгоритми на периферійних пристроях. Такі фреймворки пропонують широкий спектр інструментів та ресурсів, спеціально розроблених для реалізації стратегічних ініціатив у сфері мініатюрного машинного навчання (Tiny Machine Learning). У статті [17] розглядається вісім найвідоміших фреймворків для впровадження TinyML, як-то TensorFlow Lite (TF Lite), Edge Impulse, PyTorch Mobile, uTensor, а також платформи, як-то STM32Cube.AI, NanoEdgeAIStudio, NXP eIQ та Microsoft's Embedded Learning Library. Також описано сумісні апаратні платформи й цільові програми для цих фреймворків, що дозволяє обрати найбільш підходящі фреймворки TinyML.

5.2. Інтелектуальні системи виявлення вторгнень у мережу (NIDS)

Швидке збільшення існуючих сенсорів та систем створює нові виклики для безпеки мереж BD-IoT. Апаратна складова рівня пристроїв має технічні обмеження, пов'язані з розміром обладнання, і, як наслідок, у можливостях захисту від втручання. Через низьку обчислювальну потужність неможливо запровадити складні криптографічні алгоритми. Також вимога

до сенсорів у роботі від батареї обмежує їх у виконанні додаткових операцій. які могли б збільшити безпеку з'єднання. Для вирішення цих недоліків більшість безпекових заходів переноситься на рівень хмарних сервісів та граничних обчислень, які вже проводять перевірку валідності даних. Критичну роль відіграє система виявлення вторгнень у мережу (надалі NIDS Network Intrusion Detection System), яка оперує на цьому рівні [18].

Ефективність NIDS може бути підвищена шляхом використання алгоритмів ШІ. Основна мета NIDS системи полягає в тому, щоб на основі параметрів запиту, отриманих на мережевому рівні, провести аналіз трафіку та знайти аномальну активність, яка може свідчити про несанкціонований трафік. Трафік із систем BD-IoT може мати велику кількість ознак, і детальний аналіз кожної з них може використовувати або занадто великий обсяг обчислювального ресурсу при його обмеженій кількості, або займати забагато часу, через що шкода для системи може бути заподіяна ще до виявлення несанкціонованого доступу.

Для вирішення цієї проблеми використовується метод зменшення ознак. Наразі найбільш актуальними є два методи зменшення ознак: вибір ознак та вилучення ознак. Також можливе використання одночасного поєднання цих двох методів [19].

Метод вибору ознак означає фільтрацію трафіку до конкретних ознак, які є найбільш показовими. Його головною перевагою є збереження початкової інформації, яка легко інтерпретується. Його головним недоліком є потенційна можливість втрати важливої комбінації ознак, яка могла би бути критична для знаходження несанкціонованого трафіку.

Метод вилучення ознак аналізує усі ознаки, але для збільшення швидкодії групує їх у загальні ознаки, які вказують на загальний стан певної частини трафіку. Його основною перевагою є можливість аналізу складних показників, які формуються з різних потенційних комбінацій. Його головним недоліком є загальність результатів, через що важко вказати конкретну причину проблеми у випадку, якщо трафік був відмічений як зловмисний.

Найбільш оптимальним результатом буде використання NIDS на двох рівнях системи різними методами. Метод вибору ознак працює найбільш актуально на обмеженій кількості параметрів (від 9 до 22), що ефективно для аналізу критичних даних, на які потрібна максимально швидка реакція. Граничний рівень є найбільш оптимальним для цього. Метод вилучення ознак більш підходящий для великої кількості ознак (30+), наприклад, в умовах широкого аналізу трафіку для знаходження потенційних загроз [18]. Через високі обчислювальні вимоги до аналізу такої кількості ознак найбільш доцільним буде його виконання на рівні хмари.

6. Висновки та напрями подальших досліджень

У результаті проведеного аналізу використання ШІ в IoT отримані такі висновки:

1. Встановлено, що перехід від традиційної хмарної архітектури до розподілених архітектур із використанням ШІ сприяє горизонтальній масштабованості мереж BD-IoT та мінімізує навантаження на хмару в умовах обмежених обчислювальних та мережевих ресурсів.

2. Обґрунтовано, що використання стандартів 5G та 6G є необхідною вимогою для масштабування систем BD-IoT великою кількістю сенсорів і забезпечення стабільної роботи в умовах географічного поширення на основі збільшення пропускної здатності мереж із використанням ШІ.

3. Визначено доцільність розподілу функцій NIDS між різними рівнями BD-IoT-системи відповідно до обсягу даних і необхідної швидкості реакції при застосуванні ШІ. Використання вибору ознак на рівні граничних обчислень забезпечує високу швидкість обробки критичних даних та інтерпретацію результатів, тоді як вилучення ознак у хмарному середовищі дозволяє ідентифікувати складні приховані атаки на великих масивах даних.

4. Показано, що використання підходів TinyML є перспективним напрямом розподілу обчислювального навантаження у BD-IoT-системах. Впровадження моделей машинного навчання безпосередньо на рівні кінцевих пристроїв дозволяє суттєво мінімізувати обсяги переданих даних до хмарної інфраструктури, підвищуючи при цьому енергоефективність системи та забезпечуючи можливість прийняття критичних рішень у режимі реального часу.

Незважаючи на досягнутий за останні роки великий прогрес в оптимізації методів обробки BD-IoT, залишаються невирішеними проблеми, які ще вимагають подальших наукових досліджень. Ключовими проблемами для вирішення є:

1. Розробка полегшених криптографічних протоколів безпеки для систем, які використовують моделі TinyML на рівні пристроїв. Децентралізація обчислень та потреба у регулярному оновленні моделей на пристроях значно розширюють поверхню атак, що вимагає створення інноваційних криптографічних алгоритмів, які зможуть оптимально працювати в умовах сильно обмежених апаратних можливостей.

2. Оптимізація систем NIDS з метою підвищення ефективності виявлення порушень мережевої безпеки та зменшення вимог до апаратної складової систем на граничному та хмарному рівнях. Потребують подальшого дослідження динамічні алгоритми, які зможуть автоматично перемикатись між різними видами моделей для збереження критичних ознак, інтерпретації даних та оптимального навантаження на обчислювальні ресурси.

3. Адаптація архітектурних рішень BD-IoT до переходу на технологію 6G для систем, які мають великий масштаб використання сенсорів. Доцільно дослідити необхідні зміни в актуальних архітектурах BD-IoT для оптимального використання стільникових мереж 6G як основного стандарту мережевого рівня.

4. Подолання дефіциту ресурсів на рівні пристроїв та граничних обчислень. Потрібно дослідити можливості динамічного розподілу навантаження між сенсорами та вузлами з метою оптимального використання апаратного ресурсу. Також актуальним є дослідження шляхів поліпшення у нових апаратних архітектурах мікроконтролерів для підвищення обчислювальної потужності.

СПИСОК ДЖЕРЕЛ

1. Hassebo A., Tealab M. Global Models of Smart Cities and Potential IoT Applications: A Review. *IoT*. 2023. Vol. 4, N 3. P. 366–411. DOI: https://doi.org/10.3390/iot4030017.

2. Коваленко О.Є. Ситуаційні моделі обробки інформації в системах інтернету речей. *Математичні машини і системи*. 2025. № 2. С. 15–23. DOI: https://doi.org/10.34121/1028-9763-2025-2-15-23.

3. ITU-T Rec. Y.2060 (06/2012). Overview of the Internet of things. DOI: https://www.itu.int/rec/T-REC-Y.2060-201206-I.

4. Lee K. The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business Horizons*. 2015. Vol. 58, Issue 4. P. 431–440. DOI: https://doi.org/ 10.1016/j.bushor.2015.03.008.

5. ITU-R M.[IMT-2020.TECH PERF REQ] — Minimum Requirements Related to Technical Performance for IMT–2020 Radio Interface(s), document ITU-R M.2410-0. *International Telecommunication Union — Recommendations*. 2017. Nov. [Online]. URL: https://www.itu.int/pub/R-REP-M.2410-2017.

6. Ren R., Wang J., Yu J., Zhu X., Wan X., Lu H. Joint Resource Allocation for Multiplexing eMBB, URLLC and mMTC Traffics Based on DRL. *IEEE 99th Vehicular Technology Conference (VTC2024-Spring)*. Singapore, Singapore, 2024. P. 1–5. DOI: https://doi.org/10.1109/VTC2024-Spring62846.2024.10683585.

7. Robin Ch., Nankya M., Akl R. 6G Networks and the AI Revolution — Exploring Technologies, Applications, and Emerging Challenges. *Sensors*. 2024. N 6. P. 1888. DOI: https://doi.org/10.3390/s24061888.

8. ITU-R. Requirements related to technical performance for IMT-Advanced radio interface(s). *Report ITU-R M.2134*. 2008. URL: https://www.itu.int/pub/R-REP-M.2134-2008.

9. ITU-R. IMT Vision — Framework and overall objectives of the future development of IMT for 2020 and beyond. *Recommendation ITU-R M.2083-0*. 2015. URL: https://www.itu.int/rec/R-REC-M.2083/en.
10. 3GPP. Service requirements for the 5G system. 3rd Generation Partnership Project (3GPP), TS 22.261 V15.0.0. 2018. URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2933.
11. El Morr C., Ali-Hassan H. Descriptive, Predictive, and Prescriptive Analytics. *Analytics in Healthcare. SpringerBriefs in Health Care Management and Economics*. Springer, Cham. 2019. https://doi.org/10.1007/978-3-030-04506-7_3.
12. Коваленко О.Є. Інтелектуалізація граничних обчислень інтернету речей. *Математичні машини і системи*. 2024. № 3–4. С. 50–68. DOI: https://doi.org/10.34121/1028-9763-2024-3-4-50-68.
13. Ficili I., Giacobbe M., Tricomi G., Puliafito A. From Sensors to Data Intelligence: Leveraging IoT, Cloud, and Edge Computing with AI. *Sensors*. 2025. Vol. 25 (6). P. 1763. https://doi.org/10.3390/s25061763.
14. Marjani M., Nasaruddin F., Gani A. et al. Big IoT Data Analytics: Architecture, Opportunities, and Open Research Challenges. *IEEE Access*. 2017. Vol. 5. P. 5247–5261. DOI: https://doi.org/10.1109/ACCESS.2017.2689040.
15. Nguyen D.C., Ding M., Pathirana P.N. et al. 6G Internet of Things: A Comprehensive Survey. *IEEE Internet of Things Journal*. 2022. Vol. 9, N 1. P. 359–383. DOI: https://doi.org/10.1109/JIOT.2021.3103320.
16. Yahyati C., Lamaakal I., Maleh Y. et al. A Systematic Review of State-of-the-Art TinyML Applications in Healthcare, Education, and Transportation. *IEEE Access*. 2025. Vol. 13. P. 204513–204562. DOI: https://doi.org/10.1109/ACCESS.2025.3633575.
17. DFRobot. Top 8 TinyML Frameworks and Compatible Hardware Platforms. 2024. Jul 12. URL: https://www.dfrobot.com/blog-13921.html.
18. Li, J., Othman M.S., Chen H. et al. Optimizing IoT intrusion detection system: feature selection versus feature extraction in machine learning. *Journal of Big Data*. 2024. Vol. 11. P. 36. DOI: https://doi.org/10.1186/s40537-024-00892-y.
19. Mishra P., Varadharajan V., Tupakula U., Pilli E. A detailed investigation and analysis of using machine learning techniques for intrusion detection. *IEEE Communications Surveys & Tutorials*. 2019. Vol. 21, N 1. P. 686–728. DOI: https://doi.org/10.1109/COMST.2018.2847722.

Стаття надійшла до редакції 03.02.2026 / прийнята до друку 28.04.2026