Міністерство освіти і науки України
Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

за спеціальністю 123 «Комп'ютерна інженерія»

Тема наукової роботи: Методи запобігання витоку інформації за допомогою штучного інтелекту

Виконав _____ Д. П. Семенцов

Керівник роботи _____

Консультант[1] _____

Нормоконтроль _____

Завідувач кафедри _____

Кривий Ріг
2024

[1] Пишеться тільки у разі наявності додаткового консультанта, у зворотному випадку – видалити рядок.

Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

Ступінь вищої освіти магістр
Спеціальність 123 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ
Завідувач кафедри, голова циклової комісії

_____ А. І. Купін

"____" _____ 20__ року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

_____
(прізвище, ім'я, по батькові)

1. Тема роботи _____
_____
_____
керівник роботи _____,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від "___"_____ 20__року №__
2. Строк подання студентом роботи _____
3. Вихідні дані до роботи _____
_____
_____
_____
_____
_____
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)_____
_____
_____
_____
_____
_____
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)_____
_____
_____
_____
_____

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання_____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка

Студент _____ _____
(підпис) (прізвище та ініціали)

Керівник роботи _____ _____
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка**:** 93 сторінок, 59 рисунків, 7 таблиць, 1 додаток, 20 використаних джерел.

Об’єкт дослідження: автоматизована система керування виробництвом комплексу згущування хвостової пульпи.

Робота складається з п’яти розділів. У першому розділі розглянуто основні методи та підходи до запобігання витоку інформації з використанням технологій штучного інтелекту. Зроблено аналіз сучасних технологій, визначено переваги та обмеження використання штучного інтелекту в забезпеченні інформаційної безпеки.

Другий розділ присвячено аналізу сучасних підходів до запобігання витоку інформації. Проведено огляд методів інтеграції штучного інтелекту, зокрема машинного навчання, глибинного навчання та обробки природної мови. Висвітлено особливості та ефективність застосування таких технологій для побудови безпечних корпоративних систем.

Третій розділ містить результати моделювання, налаштування та тестування інтелектуальної системи запобігання витокам даних. Проаналізовано архітектуру системи, методи збору та обробки даних, а також оцінено ефективність різних моделей машинного навчання.

У четвертому розділі розглянуто питання інтеграції розробленої системи в реальні умови. Виконано аналіз продуктивності та можливості масштабування системи. Проведено оцінку її ефективності в реальному часі.

П’ятий розділ присвячено практичній реалізації програми. Детально описано принцип роботи, архітектуру та функціональність розробленої програми, представлено фрагменти коду з поясненнями та проведено оцінку результатів її впровадження.

Ключові слова: автоматизація, штучний інтелект, машинне навчання, запобігання витокам даних, UEBA, інформаційна безпека.

Explanatory Note: 93 pages, 59 figures, 7 tables, 1 appendices, 20 references.

Object of Research: automated control system for the operation of the tailings thickening complex.

The work consists of five sections. The first section examines the main methods and approaches for preventing information leakage using artificial intelligence technologies. An analysis of modern technologies is conducted, highlighting the advantages and limitations of using artificial intelligence for information security.

The second section focuses on analyzing modern approaches to preventing information leakage. A review of artificial intelligence integration methods is provided, including machine learning, deep learning, and natural language processing. The features and efficiency of such technologies for building secure corporate systems are outlined.

The third section presents the results of modeling, configuring, and testing an intelligent system for preventing information leaks. The system's architecture, methods for data collection and processing, and the efficiency of various machine learning models are analyzed.

The fourth section addresses the integration of the developed system into real-world conditions. The system's performance and scalability are analyzed, along with its efficiency in real-time operation.

The fifth section is devoted to the practical implementation of the program. It provides a detailed description of the program's operation principle, architecture, and functionality, along with code fragments and explanations. The results of its deployment are also evaluated.

Keywords**:** automation, artificial intelligence, machine learning, data leakage prevention, UEBA, information security.

ЗМІСТ

Оглавление

ПЕРЕЛІК СКОРОЧЕНЬ

- AI – Artificial Intelligence (штучний інтелект);
- ML – Machine Learning (машинне навчання);
- DL – Deep Learning (глибинне навчання);
- NLP – Natural Language Processing (обробка природної мови);
- UEBA – User and Entity Behavior Analytics (аналітика поведінки користувачів і об'єктів);
- DLP – Data Loss Prevention (запобігання витокам даних);
- IAM – Identity and Access Management (управління ідентифікацією та доступом);
- EDR – Endpoint Detection and Response (виявлення та реагування на кінцевих пристроях);
- SIEM – Security Information and Event Management (управління інформацією та подіями безпеки);
- CNN – Convolutional Neural Networks (згорткові нейронні мережі);
- RNN – Recurrent Neural Networks (рекурентні нейронні мережі);
- SVM – Support Vector Machines (метод опорних векторів);
- GDPR – General Data Protection Regulation (Загальний регламент захисту даних).

ВСТУП

Актуальність роботи

У сучасному світі, де промисловість і видобувна галузь стають ключовими компонентами економіки, ефективне управління виробничими процесами та оптимізація ресурсів є надзвичайно важливими завданнями. Вимірювання ефективності виробництва дозволяє оцінити, наскільки добре компанія використовує свої ресурси для створення продукції, що є критично важливим для забезпечення конкурентоспроможності на ринку.

Зокрема, у гірничодобувному секторі значні фінансові ресурси інвестуються у пошук безпечних та екологічно чистих методів переробки та зберігання відходів виробництва. Одним із важливих елементів цього процесу є автоматизація дренажних насосних систем у комплексах згущування хвостів. Надійна робота цих систем забезпечує видалення зайвої рідини з дренажних ям, оптимізуючи виробничі процеси, знижуючи втрати ресурсів і підвищуючи екологічну стійкість підприємств.

Сучасні технології автоматизації, інтегровані з передовими підходами до управління, є критично важливими для вирішення цих завдань. У цій магістерській роботі досліджується процес автоматизації дренажних насосних систем на прикладі комплексу згущування хвостів шламового господарства Акціонерного товариства «Південний гірничо-збагачувальний комбінат».

Мета та завдання дослідження

Метою роботи є розробка програмного комплексу для керування насосами дренажних ємностей із власним людино-машинним інтерфейсом (ЛМІ) та автоматизація процесів вибору обладнання для автоматизованих систем керування виробництвом. Для досягнення цієї мети у роботі передбачено виконання низки завдань, таких як аналіз технічних вимог до автоматизованих систем керування, визначення ключових параметрів для вибору обладнання, розробка алгоритму багатокритеріального вибору на основі методу суміщеного ідеалу, реалізація цього алгоритму в програмному забезпеченні, оцінка ефективності розробленого рішення через математичне моделювання та експертне опитування, а також проведення тестування системи для перевірки її відповідності заданим критеріям. Дослідження також включає оптимізацію роботи насосів шляхом інтеграції автоматизації, що спрямована на зменшення витрат енергії, підвищення точності керування та зниження залежності від участі персоналу в процесах.

Об’єкт дослідження: Автоматизована система керування виробництвом комплексу згущування хвостової пульпи.

Предмет дослідження. Програмно-апаратний комплекс для оптимізації роботи дренажних насосів у високопродуктивних згущувачах.

Методи дослідження. Для досягнення мети було застосовано теоретичний аналіз наукової літератури та технічної документації, системний аналіз функціонування дренажних насосних систем, а також методи статистичної обробки даних. Експертні методи, такі як метод Делфі та метод анкетування, доповнили аналіз, а для вибору оптимального обладнання використовувався метод суміщеного ідеалу. Метод візуалізації був застосований для створення графіків і схем, що ілюструють роботу системи.

Наукова новизна та практичне значення. У роботі виявлено недоліки у функціонуванні вузла дренажних ємностей високопродуктивних згущувачів, які проявляються у відсутності автоматизованої роботи. Запропоновано новий підхід до оптимізації цього вузла за рахунок впровадження автоматизації та методів багатокритеріального вибору обладнання. Також розроблено алгоритм автоматизації, що базується на методі суміщеного ідеалу.

Практичне значення отриманих результатів. Розроблений програмний комплекс забезпечує автоматизацію роботи насосів дренажних ємностей, підвищуючи ефективність їхньої роботи, знижуючи витрати електроенергії та участь персоналу в процесі. Інтеграція цієї системи до існуючих автоматизованих систем керування виробництвом сприяє економічній доцільності та збільшенню ресурсу роботи обладнання.

Апробація роботи

Результати дослідження були апробовані на конференції КІСМ-2023 Криворізького національного університету, що підтверджує їх наукову та практичну значущість.

Розділ 1. Методи запобігання витоку інформації за допомогою штучного інтелекту

1.1. Проблематика дослідження

Ефективне управління технологічними процесами залишається одним із ключових чинників успішності сучасних підприємств. [1, 6, 15] Зокрема, автоматизація виробничих вузлів стає важливим кроком для покращення роботи обладнання, зниження витрат енергії та підвищення загальної продуктивності. У світі, де технології швидко змінюються, автоматизація стає не лише засобом вирішення оперативних завдань, але й стратегічним інструментом для забезпечення стабільності та конкурентоспроможності підприємств у майбутньому.

Сучасні реалії, такі як зростання масштабів виробництва та посилення екологічних стандартів, висувають підвищені вимоги до автоматизованих систем керування. Такі системи вже не просто виконують запрограмовані завдання, але й адаптуються до змін умов роботи. Їх впровадження дозволяє мінімізувати вплив людського фактора, що знижує ризик помилок та забезпечує більшу безпеку. Окрім цього, автоматизовані системи здатні швидко реагувати на динамічні зміни у виробничому середовищі, забезпечуючи стабільність роботи обладнання навіть у складних умовах. Вони також підвищують точність управління критично важливими процесами, що позитивно впливає на ефективність та економічність виробництва.

Важливо зазначити, що автоматизація сьогодні не обмежується лише програмними рішеннями. Інтеграція новітніх технологій, таких як штучний інтелект та машинне навчання, відкриває нові можливості для підвищення ефективності управління виробництвом. Завдяки цим інструментам підприємства можуть отримувати детальний аналіз показників роботи обладнання, виявляти приховані проблеми та приймати обґрунтовані рішення на основі об’єктивних даних. Такі підходи дозволяють не лише підвищити ефективність роботи підприємства, а й забезпечити його адаптацію до сучасних викликів ринку.

.

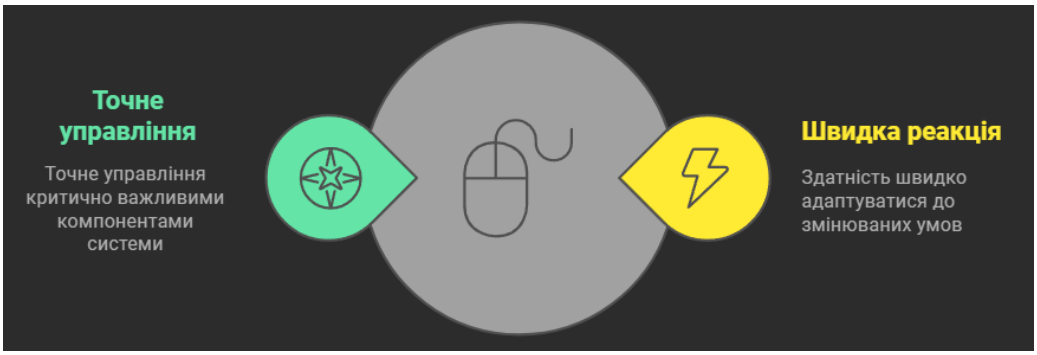

Рисунок 1.1 - Покращення ефективності системи

Одним із найважливіших вузлів, що потребують особливої уваги у виробничих системах, є дренажні насосні системи. Їхня ефективна робота має критичне значення для забезпечення стабільності виробничих процесів. Основна функція таких систем полягає у видаленні надлишкової рідини, що накопичується у технологічних та дренажних ємностях. Ця рідина може створювати ризики затоплення обладнання, погіршувати умови праці персоналу та призводити до непередбачених простоїв. Завдяки роботі насосних систем забезпечується оптимальний стан виробничого середовища, що, своєю чергою, сприяє збільшенню продуктивності та зниженню витрат на обслуговування.

Оптимізація роботи дренажних систем є не лише технічним, а й економічним завданням. Сучасні підприємства прагнуть мінімізувати експлуатаційні витрати, продовжити термін служби обладнання та зменшити споживання енергії. Застарілі або ненадійні дренажні системи можуть не лише викликати збої у роботі виробничих процесів, а й спричиняти фінансові втрати через необхідність частого ремонту або заміни обладнання. Завдяки впровадженню автоматизації стає можливим не лише уникнути таких проблем, але й суттєво покращити технічні характеристики насосних систем, що дозволяє знизити загальні витрати на їхнє обслуговування.

Крім економічних аспектів, автоматизація дренажних насосних систем відіграє важливу роль у підвищенні екологічної відповідальності підприємств. Неправильна або несвоєчасна робота таких систем може призвести до викиду небезпечних речовин у навколишнє середовище, що порушує екологічні стандарти та може стати причиною штрафних санкцій. Крім того, подібні ситуації негативно впливають на репутацію підприємства, що є важливим чинником у сучасному конкурентному середовищі. Тому модернізація дренажних систем за допомогою автоматизації є не лише технологічною вимогою, але й необхідністю для забезпечення відповідності нормативним стандартам і підвищення довіри з боку суспільства.

Автоматизація дренажних насосних систем вимагає комплексного підходу, який включає аналіз існуючих технологій, оцінку технічних вимог та впровадження інноваційних рішень. Сучасні автоматизовані системи використовують новітні підходи до моніторингу та управління, включаючи технології штучного інтелекту, машинного навчання та дистанційного

управління. Це дозволяє системам працювати у режимі реального часу, виявляти потенційні проблеми на ранніх етапах і автоматично реагувати на них.

Таким чином, модернізація дренажних насосних систем є важливим кроком на шляху до досягнення нових стандартів у продуктивності, стабільності роботи та екологічній безпеці підприємств. Інтеграція автоматизованих рішень забезпечує підприємствам можливість оптимізувати свої виробничі процеси, скоротити витрати на обслуговування та підвищити рівень екологічної відповідальності.

1.2 Концепція та напрямки дослідження

Для автоматизації дренажних насосних систем у комплексах згущування хвостів важливо врахувати специфічні особливості роботи таких систем, що зумовлюються умовами виробництва та фізико-хімічними характеристиками рідини. Основне завдання автоматизації полягає у забезпеченні стабільної роботи насосного обладнання в умовах змінного навантаження, оперативному реагуванні на коливання параметрів рідини, таких як щільність, в'язкість, об’єм та тиск, а також мінімізації енергоспоживання. Такі заходи спрямовані на підвищення ефективності роботи системи, її надійності та тривалості експлуатації.

Розробка комплексного підходу до автоматизації включає інтеграцію сучасних методів моніторингу та управління з програмними рішеннями, які здатні адаптуватися до специфіки виробничого процесу. Головною метою є створення програмного забезпечення, яке легко інтегрується з існуючими автоматизованими системами керування виробництвом (АСКВ). Програмний комплекс передбачає впровадження модулів для збору та аналізу даних у реальному часі, прогнозування можливих відхилень у роботі насосів, а також автоматичного реагування на виникнення позаштатних ситуацій.

Дослідження охоплює кілька ключових напрямків. На першому етапі проводиться інженерний аналіз сучасних технологій автоматизації дренажних систем, що дозволяє виявити їхні переваги та недоліки. Результати аналізу стають основою для розробки алгоритмів, які забезпечують високу точність роботи системи навіть у нестабільних умовах. Особливу увагу приділено розробці адаптивних алгоритмів, які враховують не лише технічні параметри насосного обладнання, але й зовнішні фактори, такі як зміни у фізико-хімічних властивостях рідини або варіативність тиску.

Важливою частиною дослідження є симуляційне моделювання, яке дає змогу перевірити ефективність розроблених рішень у різноманітних виробничих умовах. Під час симуляцій оцінюється продуктивність системи, її здатність реагувати на позаштатні ситуації та стабільність роботи в умовах підвищеного навантаження. Наприклад, симуляції можуть включати аналіз роботи насосів при різкій зміні обсягів рідини або при порушенні режиму тиску. Результати моделювання використовуються для вдосконалення системи та оптимізації її технічних параметрів. Це дозволяє зменшити витрати на обслуговування та підвищити ресурс роботи насосів.

Запропонована концепція автоматизації орієнтована на впровадження передових технологій обробки даних, машинного навчання та прогнозної аналітики. Завдяки цьому система не лише забезпечує автоматизацію рутинних процесів, але й підвищує їхню продуктивність, надійність та економічну ефективність. Використання технологій прогнозування дозволяє завчасно ідентифікувати потенційні відхилення у роботі насосів, зменшити ризики аварійних ситуацій та мінімізувати втрати.

Крім того, важливим аспектом є екологічна відповідність запропонованої системи. Автоматизовані дренажні системи сприяють зниженню негативного впливу на довкілля шляхом оптимального управління відходами та рідинами, що використовуються у виробничих процесах. Завдяки цьому підприємства можуть відповідати сучасним екологічним стандартам, забезпечуючи як фінансові, так і репутаційні переваги.

Таким чином, автоматизація дренажних насосних систем є стратегічним кроком у напрямку підвищення ефективності та екологічної відповідальності виробничих процесів. Інтеграція таких рішень забезпечує підприємствам конкурентні переваги та стабільність роботи в умовах постійно змінюваного ринкового середовища.

1.3. Тенденції розвитку ШІ в сфері інформаційної безпеки

Упродовж останніх років технології штучного інтелекту (ШІ) дедалі частіше застосовуються у сфері кібербезпеки, що відкриває нові можливості для ефективної протидії витокам інформації та іншим загрозам. В умовах сучасного інформаційного середовища, де кількість кіберзагроз постійно зростає, використання ШІ стає одним із найперспективніших напрямків. Інтеграція таких технологій дає змогу значно підвищити точність виявлення загроз, швидкість реагування на них і адаптивність систем безпеки до змін у корпоративному середовищі.

Одним із ключових аспектів впровадження ШІ у кібербезпеку є інтеграція поведінкового аналізу. Системи, що базуються на алгоритмах машинного навчання, здатні створювати детальні профілі нормальної поведінки користувачів і пристроїв. Ці профілі включають дані про звичні часи активності, типові операції з файлами, IP-адреси та інші параметри. На основі цих профілів система може ідентифікувати аномалії, які можуть свідчити про потенційні загрози. Наприклад, якщо користувач починає виконувати незвичні дії, такі як доступ до конфіденційних файлів у нетиповий час або використання незнайомого пристрою, система автоматично визначає це як відхилення від норми. Це дозволяє значно знизити ризики витоку даних і забезпечити надійний захист корпоративної інформації, водночас знижуючи кількість помилкових спрацьовувань [3, 17].

Ще однією важливою тенденцією є розвиток автономних систем виявлення та реагування (ADR). Завдяки інтеграції ШІ ці системи можуть функціонувати в автономному режимі, забезпечуючи безперервний моніторинг і аналіз подій у реальному часі. Основною перевагою таких систем є здатність не лише виявляти загрози, але й автоматично реагувати на них. Наприклад,

система може блокувати доступ до ресурсів у разі виявлення підозрілої активності, сповіщати адміністратора або виконувати заздалегідь визначені сценарії реагування. Це дозволяє значно знизити час між виявленням загрози та її нейтралізацією, що є критично важливим для запобігання потенційним витокам даних.

Значного розвитку отримали пояснювані моделі ШІ (Explainable AI). Традиційні алгоритми, попри їхню високу ефективність, часто функціонують як «чорна скринька», що створює труднощі для аналітиків, які мають оперативно реагувати на інциденти. Пояснювані моделі дозволяють зробити процес прийняття рішень прозорим і зрозумілим. Це сприяє більш ефективному використанню ШІ в корпоративному середовищі, адже забезпечує довіру до результатів аналізу й інтеграцію з існуючими бізнес-процесами.

Ще одним ключовим напрямком є адаптивне навчання. Сучасні системи ШІ здатні постійно вдосконалювати свої моделі, навчаючись на нових даних. Це дозволяє їм ефективно адаптуватися до нових типів загроз і атак. Наприклад, аналіз нових атак, таких як фішингові кампанії чи раніше невідомі вразливості програмного забезпечення, дозволяє системі автоматично оновлювати свої алгоритми. Це особливо важливо в умовах постійно змінюваного кіберсередовища, де нові типи загроз з’являються практично щодня.

Загалом, впровадження ШІ в кібербезпеку сприяє створенню високоточних, адаптивних і ефективних систем, здатних протидіяти сучасним викликам. Ці технології дозволяють компаніям не лише підвищити рівень захисту, але й оптимізувати використання ресурсів, зменшити час реагування на інциденти та зміцнити довіру клієнтів і партнерів. Усі ці фактори є критично важливими для побудови надійного захисту в умовах сучасного інформаційного суспільства [6, 7].

1.4. Існуючі підходи до запобігання витоку інформації

Дослідження сучасних технологій підтвердило, що інтеграція традиційних методів забезпечення безпеки з новітніми підходами, заснованими на штучному інтелекті, значно покращує захист інформаційних систем.

Одним із найперспективніших напрямків використання ШІ у сфері кібербезпеки є вдосконалення DLP-систем (Data Loss Prevention). Раніше ці системи зосереджувалися на аналізі текстового вмісту файлів та електронних листів для виявлення конфіденційної інформації, часто покладаючись на попередньо задані шаблони та правила. Інтеграція алгоритмів штучного інтелекту, зокрема методів обробки природної мови (NLP), дозволяє DLP-системам аналізувати контекст і розпізнавати складніші патерни. Це забезпечує можливість виявляти чутливу інформацію, таку як номери кредитних карток, персональні дані або фінансову інформацію, навіть якщо вони приховані або зашифровані. Наприклад, система може ідентифікувати фрази чи терміни, які вказують на витік даних, навіть якщо текстовий вміст навмисно викривлений. Це значно підвищує ефективність DLP-систем у боротьбі з витоками даних.

EDR-системи (Endpoint Detection and Response) також демонструють значні переваги завдяки впровадженню алгоритмів машинного навчання. Традиційні EDR-системи покладаються на попередньо визначені сигнатури загроз, що може обмежувати їхню здатність виявляти нові або модифіковані типи атак. Алгоритми машинного навчання дають змогу аналізувати поведінку користувачів і пристроїв у реальному часі, створюючи динамічні профілі, які описують звичну активність. Наприклад, якщо користувач отримує доступ до файлів або ресурсів, до яких він раніше не звертався, або якщо пристрій підключається з нової географічної локації, система автоматично виявляє таку поведінку як аномальну. Це дозволяє оперативно реагувати на потенційні загрози, наприклад, блокуючи підозрілі дії або сповіщаючи адміністратора.

IAM-системи (Identity and Access Management) також значно вдосконалюються завдяки використанню ШІ. Традиційно права доступу у таких системах встановлюються статично, що не завжди враховує динамічну природу поведінки користувачів. Інтеграція ШІ дозволяє IAM-системам аналізувати поведінкові дані і змінювати рівень доступу залежно від контексту. Наприклад, якщо система фіксує підозрілі дії користувача, такі як запити на доступ до файлів, які раніше не були в зоні його відповідальності, може бути ініційована додаткова перевірка, наприклад багатофакторна аутентифікація. Це допомагає значно знизити ризики компрометації облікових записів і захистити критично важливі ресурси компанії.

Таким чином, інтеграція технологій штучного інтелекту в DLP, EDR та IAM-системи відкриває нові горизонти у сфері інформаційної безпеки. Це дозволяє компаніям не лише підвищити рівень захисту, але й швидше та ефективніше реагувати на сучасні загрози, забезпечуючи адаптивність і надійність систем у динамічному цифровому середовищі.

Рисунок 1.2 - Ключові функції штучного інтелекту в інформаційній безпеці

Таблиця 1.1 - порівняння основних видів захисту

Система	Основний фокус	Переваги	Обмеження
DLP	Контроль за передачею даних	Захист чутливої інформації	Не забезпечує поведінкового аналізу
EDR	Виявлення загроз на кінцевих пристроях	Реальний час	Вимагає постійного моніторингу
IAM	Управління ідентифікацією та доступом	Несанкціонований доступ	Не виявляє аномалії у поведінці

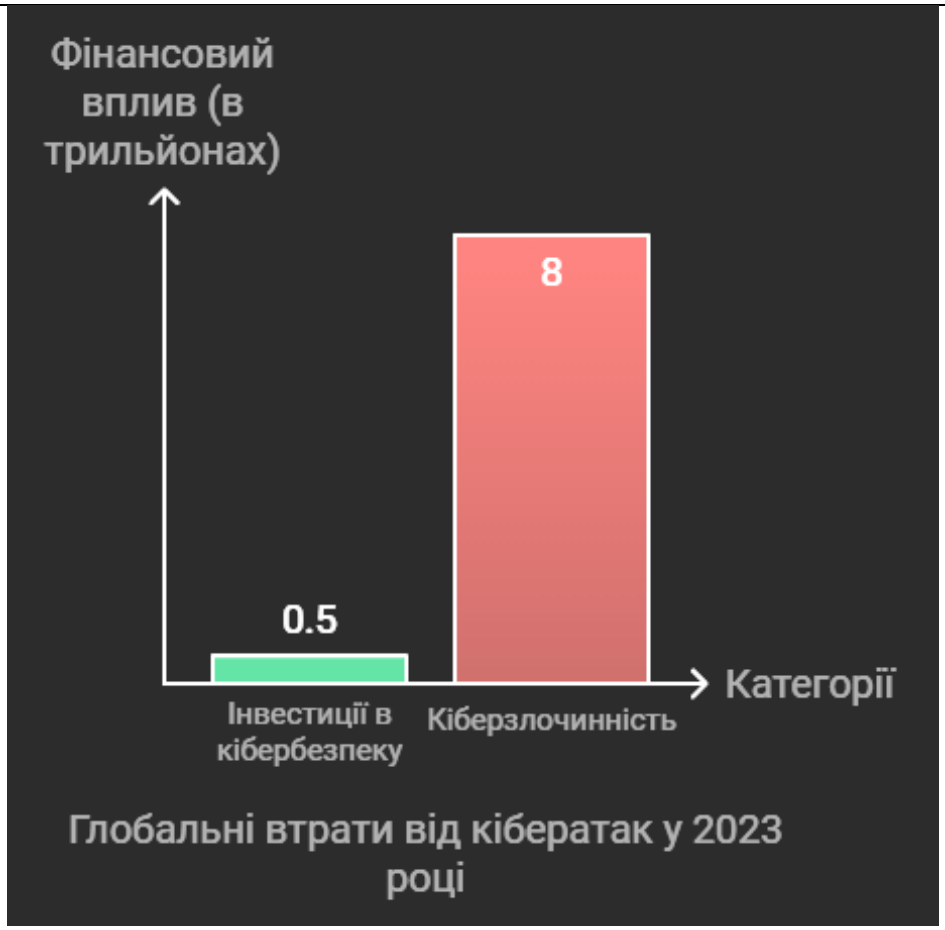

Рисунок 1.3 - Графік втрат від кібератак

1.5. Огляд алгоритмів машинного навчання для виявлення аномалій

Технології штучного інтелекту виступають ключовим інструментом для виявлення аномалій у поведінці користувачів та пристроїв.

Сучасні алгоритми забезпечують автоматизацію процесів аналізу великих обсягів даних, що дозволяє значно підвищити рівень безпеки в корпоративному середовищі. Одними з найпоширеніших алгоритмів, які застосовуються в таких системах, є:

Support Vector Machines (SVM) створюють гіперплощину для поділу даних на класи, наприклад, "нормальні" та "аномальні". Цей підхід ідеально підходить для завдань з високовимірними даними завдяки своїй здатності

знаходити оптимальні рішення навіть у складних просторах. Проте SVM потребує значних обчислювальних ресурсів і може бути менш ефективним при аналізі великих наборів даних у реальному часі. [4, 12]

Дерева рішень є зручними для розробки моделей прийняття рішень, оскільки базуються на простих логічних умовах. Вони забезпечують швидке навчання та прийняття рішень, що робить їх корисними для завдань, де важливий час обробки. Однак їх точність може бути обмеженою при роботі зі складними наборами даних, що вимагають більш складного аналізу.

Нейронні мережі та методи глибинного навчання, такі як згорткові нейронні мережі (CNN) і рекурентні нейронні мережі (RNN), демонструють виняткові результати у виявленні складних закономірностей у поведінці користувачів. Ці алгоритми здатні ідентифікувати найдрібніші деталі в масивних наборах даних, що дозволяє їм виявляти аномалії з високою точністю навіть у реальному часі. Їх застосування особливо актуальне для систем, які аналізують великі обсяги даних і потребують швидкого виявлення потенційних загроз.

Алгоритми в системах UEBA: У таких системах, як UEBA (User and Entity Behavior Analytics), ці алгоритми дозволяють формувати профілі нормальної поведінки користувачів і пристроїв, визначати відхилення від норми та автоматизувати процес реагування на потенційні загрози. Це забезпечує надійну основу для ефективного моніторингу та запобігання інцидентам витоку даних.

На Рисунку 1.4 представлено порівняння точності основних алгоритмів машинного навчання, які використовуються у системах UEBA:

- Support Vector Machines (SVM) демонструють високу точність (85%) та ефективно працюють із високовимірними даними, проте вимагають значних обчислювальних ресурсів.
- Дерева рішень забезпечують швидкий процес навчання і прийняття рішень, але їхня точність (78%) поступається іншим методам.
- Нейронні мережі демонструють найвищу точність (92%), що робить їх найефективнішими для виявлення аномалій у великих обсягах даних. [9, 3, 18]

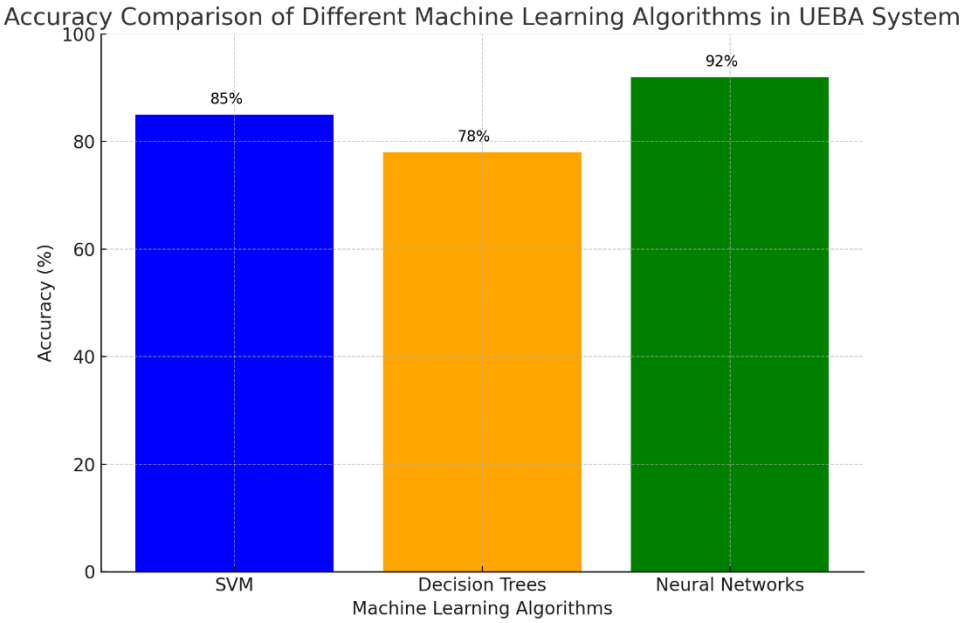

Рисунок 1.4 - Порівняння точності різних алгоритмів машинного навчання в системі UEBA

Графік демонструє, що хоча всі три алгоритми мають свої переваги та недоліки, нейронні мережі забезпечують найбільшу точність, особливо при обробці великих і складних наборів даних. Це робить їх ідеальним вибором для використання в умовах сучасних викликів кібербезпеки.

Рисунок 1.5 - Покращення захисту даних за допомогою ШІ

1.6. Принцип роботи системи UEBA

Система UEBA (User and Entity Behavior Analytics) — це інноваційний інструмент кібербезпеки, орієнтований на моніторинг і аналіз поведінки користувачів і пристроїв у корпоративних мережах. Її головною метою є виявлення аномалій, які можуть сигналізувати про потенційні загрози, як зовнішнього, так і внутрішнього характеру. Завдяки здатності до адаптації до індивідуальних патернів поведінки кожного користувача, UEBA забезпечує точніше виявлення відхилень і значно підвищує ефективність виявлення загроз.

Етапи роботи системи UEBA

Збір даних є початковим етапом, на якому система отримує інформацію з різноманітних джерел: журнали подій, мережевий трафік, активність користувачів і дії кінцевих пристроїв. Зібрані дані піддаються нормалізації, що включає їх очищення від шуму, стандартизацію форматів та підготовку для подальшого аналізу. Цей процес дозволяє підвищити якість даних і забезпечити їхню сумісність із алгоритмами машинного навчання. Приклад: якщо користувач завантажує великий обсяг даних у нетиповий час, система реєструє цю подію для подальшого аналізу.

Побудова моделей поведінки базується на алгоритмах машинного навчання, які створюють профілі «нормальної» поведінки кожного користувача та пристрою. Ці моделі враховують такі характеристики, як час доступу до ресурсів, типи файлів, з якими взаємодіє користувач, і звичні IP-адреси. Постійне оновлення профілів дозволяє системі адаптуватися до змін у поведінці користувачів, знижуючи кількість хибних спрацьовувань. Приклад: для відділу продажів система може враховувати їхній звичний доступ до комерційних документів і типові години активності.

Виявлення аномалій здійснюється шляхом порівняння поточної поведінки користувача з базовими моделями. Якщо система фіксує значні відхилення, вони позначаються як аномалії. Для підвищення точності цього процесу застосовуються алгоритми глибинного навчання, зокрема LSTM, здатні ідентифікувати приховані закономірності у складних наборах даних. Приклад: якщо співробітник HR отримує доступ до фінансових звітів, що не входять до його компетенції, система маркує це як потенційну загрозу.

Автоматичне реагування дозволяє системі оперативно реагувати на виявлені аномалії. Реакція може варіюватися від сповіщення адміністратора до автоматичного блокування дій користувача чи обмеження доступу до ресурсів. Усі дії документуються для подальшого аналізу. Приклад: у разі виявлення спроби передачі конфіденційних даних за межі мережі система може автоматично заблокувати цю операцію.

Переваги використання UEBA

- Адаптивність до загроз: система автоматично підлаштовується під нові поведінкові патерни користувачів і пристроїв.

- Інтеграція з іншими інструментами: UEBA може поєднуватися з DLP, EDR та IAM-системами для створення комплексної екосистеми кіберзахисту.
- Швидка реакція: автоматизація процесів дозволяє зменшити час реагування на інциденти до мінімуму.
- Ефективність при внутрішніх загрозах: система аналізує аномалії, які можуть вказувати на компрометацію облікових записів або інсайдерські атаки.

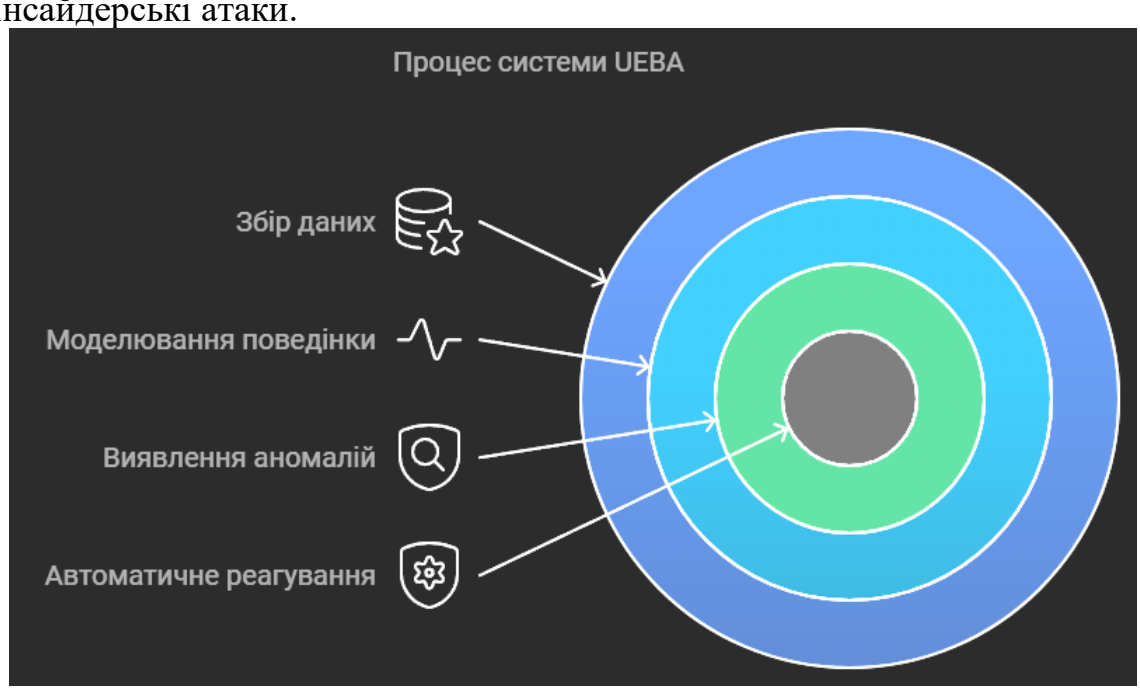

Рисунок 1.6 – Принцип роботи UEBA

Таким чином, UEBA забезпечує комплексний підхід до забезпечення кібербезпеки шляхом детального аналізу поведінки користувачів і пристроїв. Вона допомагає виявляти загрози ще на початкових етапах, значно знижуючи ризик витоків інформації. Інтеграція таких систем дозволяє компаніям не лише підвищити рівень безпеки, але й підвищити ефективність управління ризиками, що є критичним у сучасних умовах зростаючих кіберзагроз

1.7. Переваги та обмеження використання ШІ у запобіганні витокам даних

Використання штучного інтелекту для запобігання витокам даних має свої переваги та обмеження, які потрібно враховувати при інтеграції ШІ-систем у корпоративну інфраструктуру.

Переваги:

Автоматизація та адаптивність: ШІ автоматично аналізує поведінкові патерни та виявляє нові загрози, адаптуючись до змін у поведінці користувачів.

Зниження кількості хибних спрацьовувань: Завдяки машинному навчанню, ШІ-системи можуть відрізняти аномалії від нормальних дій, що знижує кількість помилкових сповіщень.

Проактивний підхід до безпеки: ШІ дозволяє передбачати потенційні загрози на основі історичних даних та діяти на випередження.

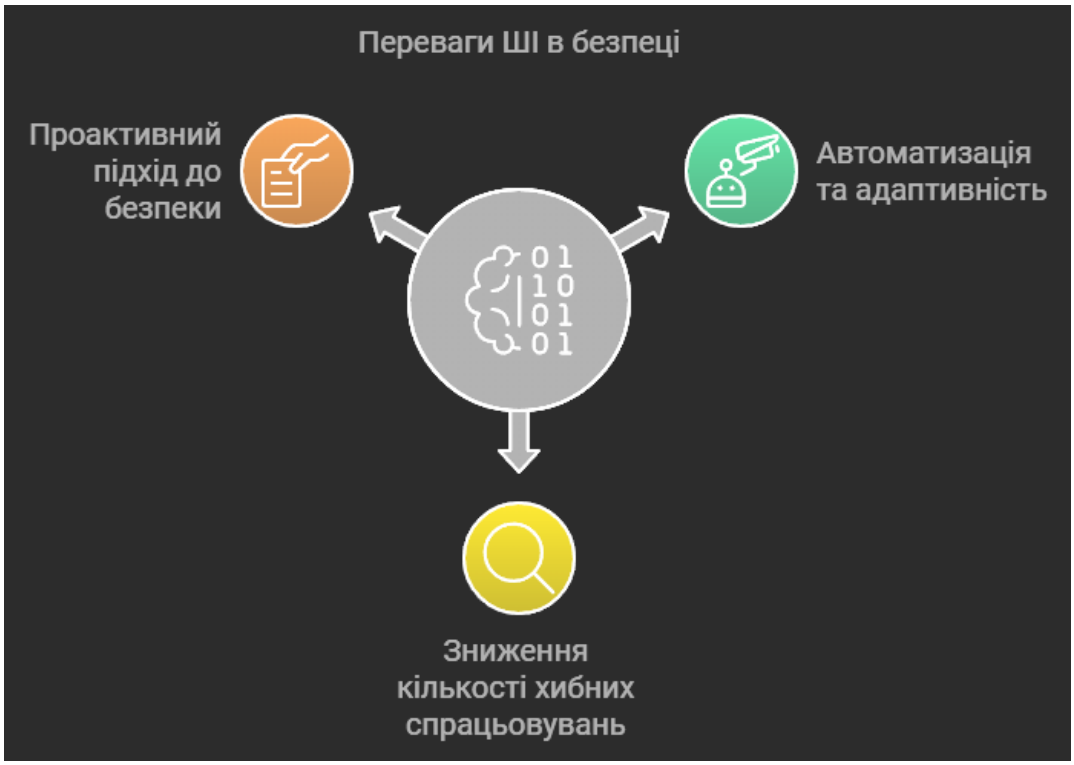

Рисунок 1.7 - Перевега ШІ в безпеці

Обмеження:

Залежність від якості даних є однією з ключових характеристик моделей штучного інтелекту, які потребують значної кількості точних і репрезентативних даних для ефективного навчання. Низька якість або недостатній обсяг даних можуть стати причиною некоректних результатів і зниження продуктивності системи. Високий рівень споживання ресурсів також є важливим аспектом: алгоритми штучного інтелекту, особливо ті, що використовують методи глибинного навчання, вимагають значних обчислювальних потужностей для аналізу великих обсягів інформації, що може впливати на їх доступність у середовищах із обмеженими ресурсами. Складність налаштування та підтримки**:** Впровадження ШІ вимагає спеціалізованих знань для налаштування та моніторингу моделей, що може створити додаткові витрати для організації.

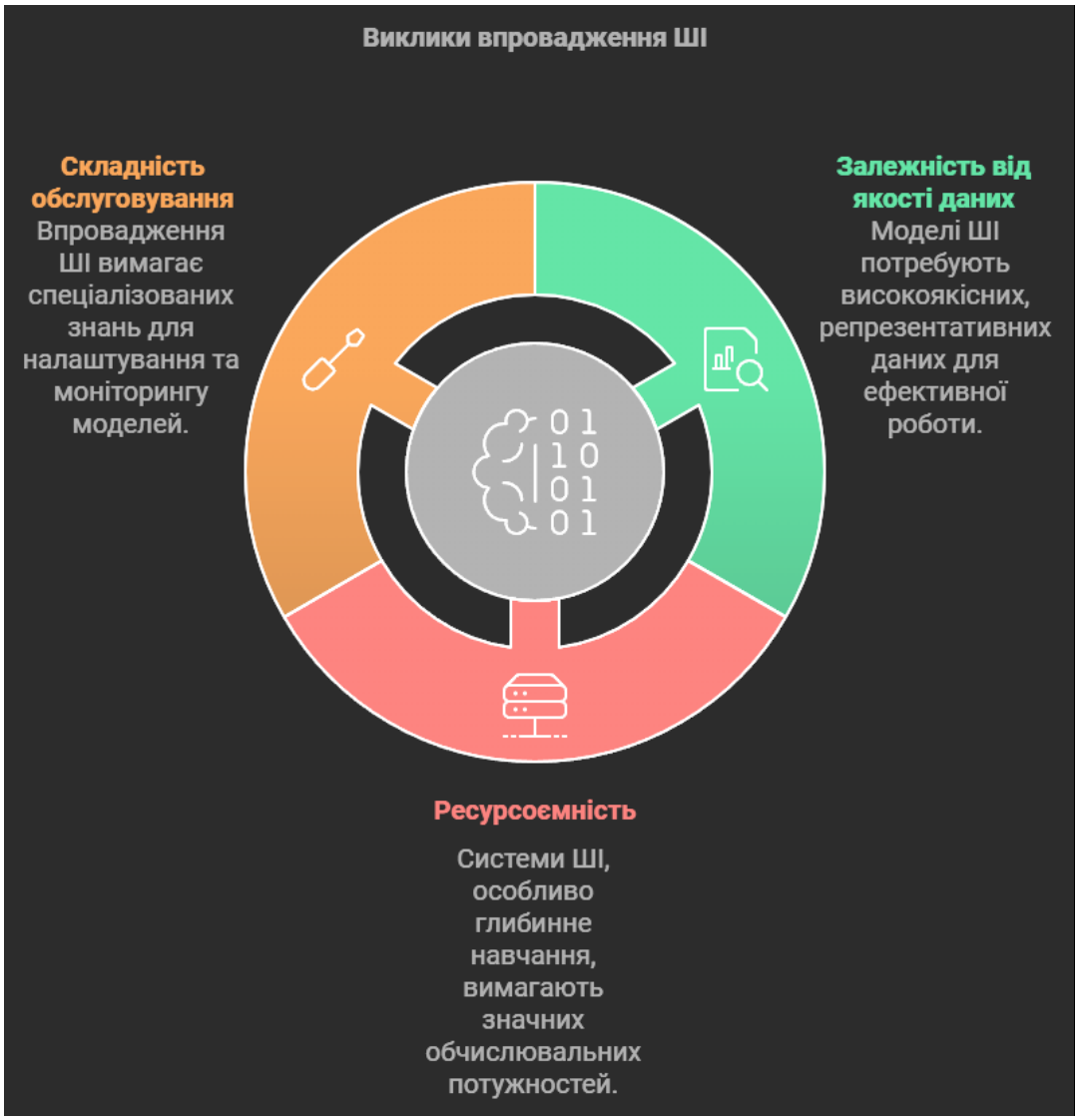

Рисунок 1.8 - Виклики впровадження ШІ

Штучний інтелект пропонує потужні інструменти для підвищення точності та швидкості виявлення загроз, але різні підходи мають свої особливості. Рисунок 1.8 демонструє порівняння точності між двома підходами – машинним навчанням (ML) і глибинним навчанням (DL**)** – у виявленні аномалій.

- Машинне навчання (ML): Точність машинного навчання становить 82%. Цей підхід дозволяє ефективно виявляти загрози, особливо при обробці структурованих даних. Однак у складних і динамічних середовищах він може бути обмеженим, оскільки потребує більше налаштувань для досягнення оптимальних результатів.
- Глибинне навчання (DL): Завдяки багатошаровим нейронним мережам, глибинне навчання забезпечує точність до 90%. Це робить його ідеальним для обробки складних і великих наборів даних, де необхідно розпізнавати приховані патерни та аномалії.

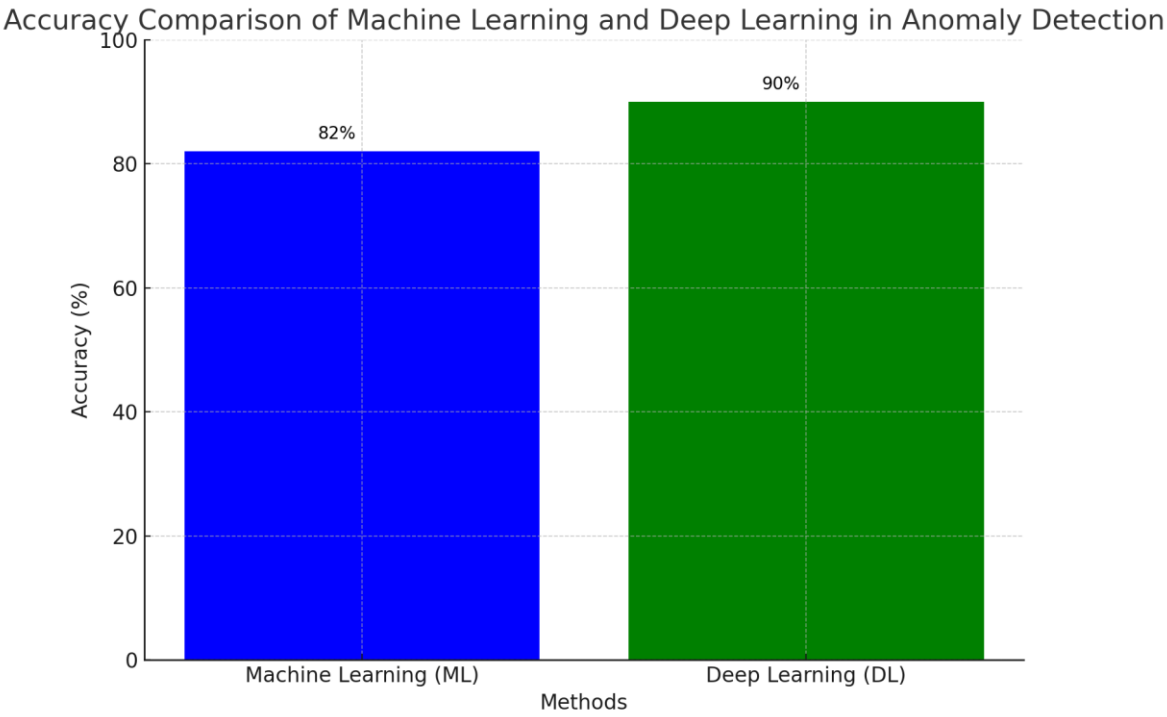

Рисунок 1.9 – Порівняння точності машинного навчання та глибинного навчання у виявленні аномалій

Графік ілюструє, що глибинне навчання є більш точним і гнучким для завдань виявлення аномалій у кібербезпеці, що підтверджує його переваги у боротьбі з сучасними кіберзагрозами.

Висновки до розділу 1

У першому розділі було розглянуто основні методи та підходи до запобігання витоку інформації в корпоративних інформаційних системах. Основними засобами захисту є системи DLP, EDR, та IAM, кожна з яких має свої сильні сторони і обмеження. Також було детально розглянуто роль штучного інтелекту в забезпеченні безпеки даних. Технології машинного навчання, глибинного навчання та обробки природної мови дозволяють створювати адаптивні системи, здатні до автоматичного виявлення загроз і аномалій у поведінці користувачів та системи.

Розгляд сучасних технологій показав, що поєднання традиційних методів захисту, таких як DLP, EDR та IAM, з технологіями штучного інтелекту суттєво підвищує рівень безпеки інформаційних систем. ШІ допомагає адаптуватися до нових видів загроз, роблячи системи захисту більш гнучкими, точними та здатними до роботи в режимі реального часу.

Розділ 2: Сучасні підходи до запобігання витокам інформації з елементами штучного інтелекту

2.1. Інтеграція штучного інтелекту в існуючі підходи до запобігання витокам інформації

Традиційні методи захисту від витоків інформації, такі як базові Data Loss Prevention (DLP) системи, забезпечують фундаментальний рівень безпеки. Вони працюють за принципом визначених правил і шаблонів для аналізу даних та їх передачі. Проте в сучасних умовах, коли обсяги інформації постійно зростають, а кіберзагрози стають дедалі складнішими, цих методів часто виявляється недостатньо. Наявні системи мають обмежену здатність до виявлення нових типів загроз, які не відповідають відомим шаблонам, та не завжди можуть вчасно реагувати на нетипові сценарії. У такому контексті інтеграція технологій штучного інтелекту (ШІ) відкриває нові можливості для вдосконалення існуючих рішень у сфері інформаційної безпеки.

Застосування ШІ у DLP-системах дозволяє значно підвищити їхню ефективність. Зокрема, однією з ключових переваг є здатність аналізувати великі обсяги інформації у реальному часі. Алгоритми машинного навчання дозволяють таким системам адаптуватися до поведінки користувачів і пристроїв, створюючи індивідуальні профілі для кожного. Наприклад, система може визначати «нормальний» патерн роботи конкретного співробітника, аналізуючи, з якими файлами він зазвичай працює, у який час та з яких пристроїв. Це дає змогу ідентифікувати аномалії, які вказують на можливі спроби витоку інформації або компрометацію облікового запису.

Окрім аналізу поведінки, ШІ дозволяє впроваджувати алгоритми обробки природної мови (NLP), які допомагають DLP-системам краще ідентифікувати конфіденційні дані. Наприклад, завдяки NLP система може розпізнавати номери кредитних карток, персональні дані чи корпоративні секрети навіть у випадках, коли вони приховані у текстах або подані у нестандартних форматах. Це суттєво підвищує здатність систем виявляти приховані загрози, які можуть бути пропущені традиційними методами.

Ще однією перевагою є можливість динамічної адаптації до нових загроз. Сучасні системи з використанням ШІ здатні постійно навчатися на нових даних, що дозволяє їм швидко реагувати на зміни у поведінці користувачів або появу нових видів атак. Наприклад, якщо зловмисник намагається завантажити конфіденційні файли на зовнішній носій або відправити їх через непідконтрольні сервіси, система миттєво реєструє такі дії як аномальні та вживає заходів, наприклад, блокує операцію або сповіщає адміністратора.

Автоматизація реагування є ще однією важливою перевагою ШІ в DLP-системах. Раніше більшість систем лише повідомляли про інциденти, залишаючи прийняття рішень адміністраторам. Зараз же сучасні алгоритми ШІ здатні самостійно оцінювати ризики та виконувати попередньо запрограмовані дії, наприклад, обмежувати доступ до даних або тимчасово блокувати обліковий запис. Це дозволяє значно скоротити час реагування на інциденти,

що особливо важливо в умовах масштабних корпоративних середовищ із великим обсягом інформаційних потоків.

Попри очевидні переваги, впровадження ШІ в DLP-системи також має свої виклики. Одним із них є необхідність у великих обсягах якісних даних для навчання моделей. Неточності в даних можуть призводити до хибних спрацьовувань або, навпаки, до пропуску реальних загроз. Ще один виклик — це потреба в значних обчислювальних ресурсах для забезпечення роботи алгоритмів у реальному часі. Тим не менш, переваги, які надають такі системи, значно перевищують їхні недоліки, особливо для великих компаній, які прагнуть забезпечити максимальний рівень захисту своєї інформації.

Інтеграція ШІ в DLP-системи — це стратегічний крок до створення більш надійного, адаптивного та ефективного інструменту захисту корпоративних даних. Завдяки таким технологіям компанії можуть не лише виявляти та блокувати потенційні витоки інформації, а й значно підвищити свою готовність до сучасних кіберзагроз.

2.2. Використання штучного інтелекту для виявлення аномалій та запобігання витокам даних

Штучний інтелект є потужним інструментом, що дозволяє підвищити ефективність та точність виявлення загроз у інформаційних системах. Основні технології ШІ для запобігання витокам інформації включають:

- Машинне навчання (ML)**:** Машинне навчання дозволяє будувати моделі нормальної поведінки користувачів і пристроїв, що допомагає виявляти аномалії. Наприклад, система може навчитися визначати звичний час доступу користувача до файлів або типові файли, до яких він звертається. Будь-яке відхилення від цих патернів, таке як спроба доступу до конфіденційних даних у незвичайний час або з нового пристрою, буде розцінено як потенційна загроза. Алгоритми кластеризації допомагають групувати подібну поведінку користувачів, а виявлення аномалій відбувається шляхом ідентифікації поведінки, що не належить до жодного з кластерів нормальної активності.
- Глибинне навчання (DL)**:** Використовуючи багатошарові нейронні мережі, глибинне навчання дозволяє виявляти складні патерни в великих обсягах даних. Це особливо корисно для аналізу мережевого трафіку, де глибинне навчання може відстежувати нетипові комунікації та ідентифікувати раніше невідомі загрози. Наприклад, дослідження показали, що глибинне навчання збільшує точність виявлення атак на 20–30% порівняно з традиційними методами, особливо коли йдеться про нові типи атак, які не відповідають відомим шаблонам. Застосування згорткових нейронних мереж (CNN) дозволяє аналізувати зображення, схеми та інші візуальні дані, що може бути корисно для виявлення нетипових моделей трафіку.
- Обробка природної мови (NLP): NLP забезпечує аналіз текстових даних, таких як електронні листи та документи, для автоматичного виявлення конфіденційної інформації та запобігання її

передачі за межі організації. [7, 14, 20] Це особливо важливо для уникнення витоків інформації через електронну пошту чи месенджери, де ймовірність витоку даних є високою. Наприклад, NLP-системи здатні визначати текстові патерни, які містять чутливу інформацію, як-от номери банківських рахунків чи особисті дані. Згідно з даними компанії McAfee, використання NLP у DLP-системах знижує ризик витоку даних на 25%.

На Рисунку 2.5 зображено архітектуру роботи системи, що використовує ШІ для запобігання витокам даних. Вона складається з кількох етапів, кожен з яких відіграє важливу роль у виявленні аномалій та реагуванні на загрози.

Рисунок 2.5 – Архітектура роботи системи з використанням машинного та глибинного навчання для запобігання витокам даних

2.3. Переваги та обмеження сучасних підходів

Розглянуті методи мають як переваги, так і обмеження, що робить їх оптимальними для певних завдань, але менш ефективними для інших.

Переваги сучасних методів:

Швидке виявлення загроз у реальному часі: ШІ-системи можуть оперативно виявляти загрози, що дозволяє швидко реагувати на потенційні інциденти.

Адаптивність та автоматичне навчання**:** Сучасні алгоритми машинного навчання здатні до автоматичного навчання на нових даних, що допомагає системі адаптуватися до нових видів атак.

Комплексний захист**:** Комбінація DLP, EDR, IAM і ШІ забезпечує багаторівневий підхід до захисту даних, що охоплює як виявлення аномалій, так і реагування на загрози.

Обмеження сучасних методів:

Залежність від якості даних: Результативність роботи систем, заснованих на штучному інтелекті, безпосередньо залежить від обсягу та якості вхідних даних. Використання нерепрезентативних або недостатньо якісних

даних може знизити точність прогнозів та призвести до помилкових висновків. Складність інтерпретації: Багато моделей штучного інтелекту, зокрема ті, що базуються на глибокому навчанні, мають непрозорий характер роботи. Це створює труднощі у розумінні їхніх рішень, що може ускладнювати пояснення результатів кінцевим користувачам.

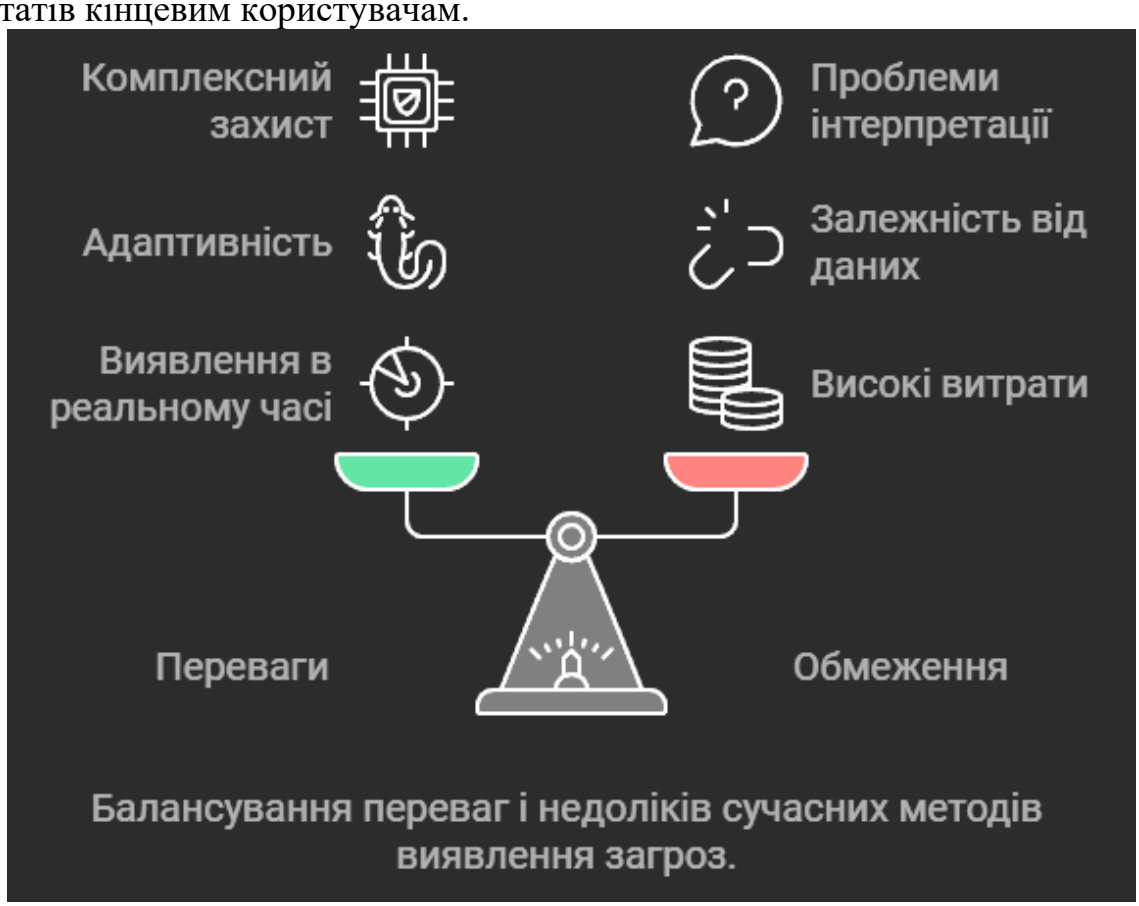

Рисунок 2.6 - Переваги та обмеження сучасних підходів

2.4. Обробка та підготовка даних для навчання ШІ-систем

Ефективність роботи систем штучного інтелекту (ШІ) значною мірою залежить від якості даних, які використовуються для навчання та тестування моделей. Системи, орієнтовані на забезпечення інформаційної безпеки, вимагають ретельно організованого процесу збору, обробки та захисту даних. Ці етапи забезпечують створення надійної основи для моделювання поведінки користувачів та виявлення потенційних загроз.

Збір даних є початковим і одним із найважливіших етапів, адже якість даних безпосередньо впливає на точність моделей. Інформація надходить із різноманітних джерел, таких як лог-файли операційних систем, мережевий трафік, історія взаємодії користувачів із додатками, активність пристроїв тощо. Ці дані формують базу для створення профілів нормальної поведінки, які допомагають виявляти відхилення від стандартів. Наприклад, якщо користувач регулярно взаємодіє з певними файлами у стандартний робочий час, система визначає це як норму. Однак спроби отримати доступ до незнайомих ресурсів у нетиповий час можуть бути позначені як потенційна загроза.

На другому етапі — попередня обробка даних — інформація проходить етап очищення, нормалізації та стандартизації. Основна мета цього етапу

полягає у видаленні шуму, усуненні дублікатів та підготовці даних до подальшого використання в алгоритмах машинного навчання. Наприклад, події, які трапляються надзвичайно рідко або не є релевантними для навчання моделі, фільтруються, щоб уникнути впливу на кінцеві результати. Нормалізація даних забезпечує єдиний формат для всіх джерел інформації, що сприяє стабільній роботі алгоритмів. Наприклад, різні часові формати, одиниці виміру або структури записів приводяться до єдиного стандарту, щоб виключити помилки у процесі аналізу.

Окрім цього, попередня обробка може включати формування додаткових ознак (feature engineering), які допомагають моделі краще ідентифікувати патерни поведінки. Наприклад, можуть створюватися нові ознаки, які враховують взаємозв'язки між різними подіями, такими як частота доступу до певних ресурсів або зміни у географічному розташуванні пристрою.

На третьому етапі здійснюється забезпечення конфіденційності даних. Це особливо важливо в умовах дотримання сучасних регуляцій, таких як Загальний регламент захисту даних (GDPR). Конфіденційність даних забезпечується за рахунок анонімізації або шифрування. Анонімізація включає перетворення персональних даних у знеособлені форми, що виключають можливість ідентифікації конкретних користувачів. Наприклад, замість збереження реальних імен або IP-адрес, система може використовувати псевдоніми або хешовані значення. Це дозволяє проводити аналіз без порушення конфіденційності. Шифрування даних під час зберігання та передачі додає ще один рівень захисту, запобігаючи доступу до інформації у разі несанкціонованого проникнення.

Правильна організація цих етапів є вирішальною для успішної роботи ШІ-системи. Збір релевантних даних забезпечує наявність достатньої інформації для моделювання. Попередня обробка гарантує, що система отримує тільки якісну та підготовлену інформацію, яка сприяє точному навчанню. Захист конфіденційності, у свою чергу, дозволяє уникнути етичних та правових проблем, зберігаючи при цьому функціональність системи.

Таким чином, усі три етапи — збір, обробка та захист даних — формують базу для створення ефективних ШІ-систем, здатних виявляти аномалії у поведінці користувачів та оперативно реагувати на загрози. Дотримання цих процесів гарантує стабільну роботу моделі, її адаптивність та надійність у сучасних умовах швидко змінюваного кіберсередовища.

Рисунок 2.7 - Процес збору, обробки та захисту даних для аналізу в системах ШІ

2.5. Методи виявлення аномалій у ШІ-системах

Методи виявлення аномалій у ШІ-системах, які активно використовуються в сфері кібербезпеки, відіграють ключову роль у забезпеченні ефективності цих систем. Кожен із них має свої унікальні особливості, що дозволяють вирішувати конкретні завдання виявлення загроз і відхилень.

Одним із найпоширеніших підходів є методи кластеризації, які застосовуються для групування схожих елементів у наборах даних. Вони допомагають ідентифікувати об’єкти або події, які не відповідають звичним патернам. Наприклад, алгоритми K-means або DBSCAN дозволяють виділити групи з нормальними поведінковими характеристиками і виявити аномалії. У системах моніторингу мережевого трафіку це дозволяє ідентифікувати незвичайну активність користувача, яка може свідчити про потенційну загрозу, як-от спробу несанкціонованого доступу.

Інший популярний метод — підхід на основі дерев рішень, який часто використовується для класифікації та прогнозування. Дерева рішень будують моделі на основі простих логічних умов, що робить їх зрозумілими та ефективними в практичному застосуванні. Наприклад, система безпеки може оцінювати ризик доступу до конфіденційних даних у нетиповий час, а потім позначати такі дії як потенційно небезпечні.

Сучасні методи багатошарових нейронних мереж дозволяють проводити аналіз великих обсягів даних у реальному часі. Зокрема, згорткові нейронні мережі (CNN) ефективно працюють із просторовими даними, наприклад, для виявлення патернів у мережевому трафіку. Цей підхід особливо корисний для виявлення загроз у структурованих наборах даних, де важливо враховувати взаємозв'язок між різними елементами інформації.

Рекурентні нейронні мережі (RNN) забезпечують високу точність аналізу послідовних даних, таких як тимчасові ряди. Вони застосовуються для ідентифікації змін у поведінці користувачів, які виникають у часі. Наприклад, ці мережі можуть аналізувати активність користувачів, виявляючи аномальні

патерни, як-от різкі збільшення кількості запитів до конфіденційних ресурсів за короткий проміжок часу.

Методи опорних векторів (SVM) є ще одним потужним інструментом для розпізнавання відхилень, особливо у багатовимірних наборах даних. SVM створює гіперплощину, яка розділяє дані на класи, наприклад, "нормальні" та "аномальні". Це дозволяє точно ідентифікувати аномалії навіть у складних наборах даних із великою кількістю характеристик.

Ще одним важливим підходом є аналіз тимчасових рядів, який базується на оцінці поведінки користувачів у часі. Наприклад, система може моніторити активність користувачів на основі тимчасових інтервалів, визначаючи відхилення, які свідчать про можливу загрозу, як-от спроби доступу до критичних ресурсів у незвичайний час.

Ці методи забезпечують різносторонній підхід до виявлення аномалій, дозволяючи системам на основі ШІ залишатися точними, адаптивними та швидкими у протидії загрозам. Інтеграція таких методів у сучасні системи кібербезпеки робить їх ефективним інструментом для виявлення та запобігання потенційним ризикам у корпоративному середовищі.

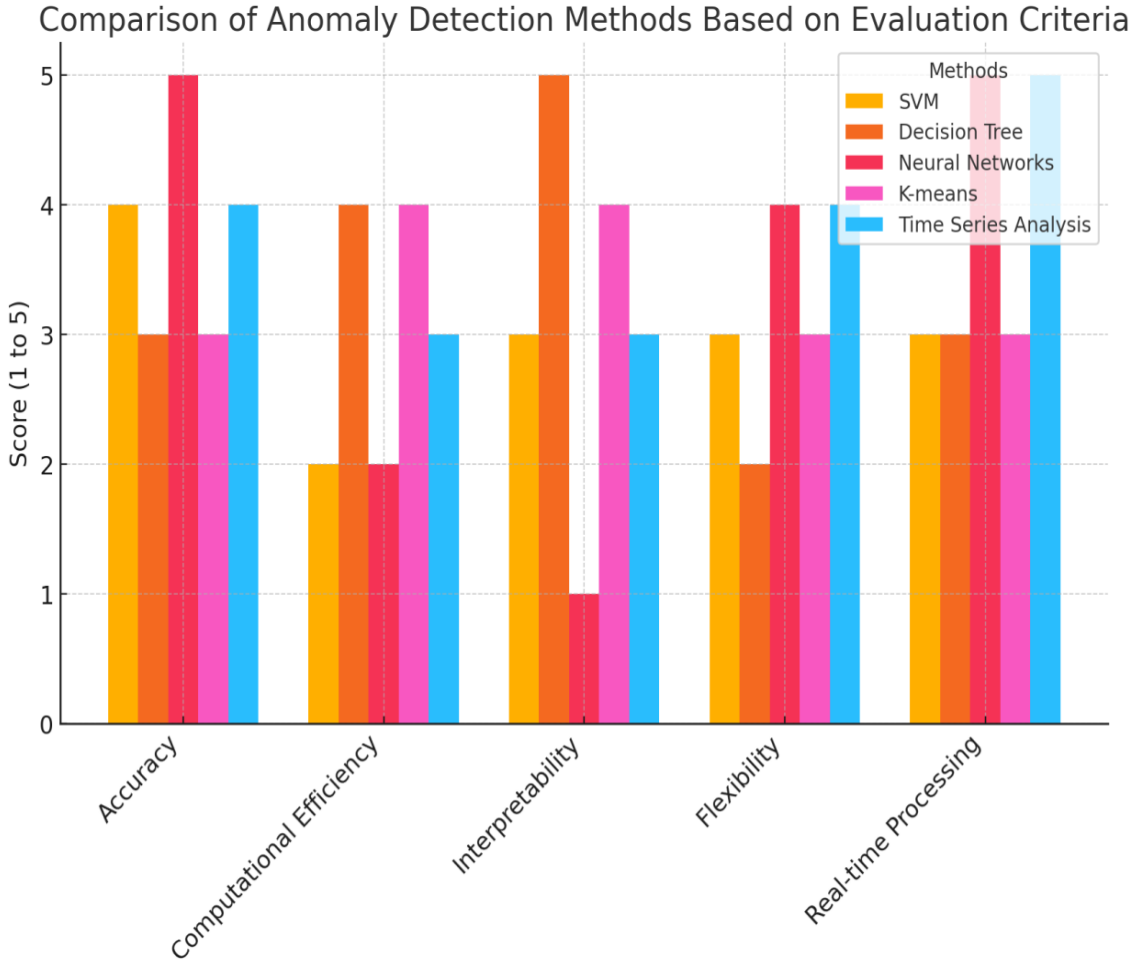

Рисунок 2.8 – Порівняння основних методів виявлення аномалій на основі машинного навчання

2.6. Приклади застосування ШІ для запобігання витокам даних

Реальні приклади застосування штучного інтелекту у запобіганні витокам даних демонструють суттєві переваги цих технологій у вдосконаленні корпоративних систем безпеки. Один із таких прикладів — виявлення нетипових дій користувачів. Алгоритми машинного навчання здатні аналізувати поведінку користувачів і помічати аномалії, наприклад, доступ до конфіденційних файлів у неробочий час або з незнайомих пристроїв. Такі дії автоматично позначаються як підозрілі або блокуються до з'ясування обставин, що значно зменшує ризик витоку інформації.

Інший приклад — аналіз тексту для ідентифікації конфіденційної інформації за допомогою обробки природної мови (NLP). Ця технологія використовується для сканування вмісту електронної пошти, документів та повідомлень у чатах, щоб виявити чутливу інформацію, яка може бути передана за межі компанії. Зокрема, система на базі NLP може розпізнати номера кредитних карток, персональні дані або іншу критично важливу інформацію, блокуючи її передачу в разі порушення корпоративних правил безпеки.

Ще одним важливим напрямком є захист від фішинг-атак. ШІ-системи, що базуються на обробці тексту та класифікації, здатні аналізувати вміст листів і повідомлень, виявляючи фішингові атаки. Наприклад, вони можуть визначати шкідливі посилання або вкладення й автоматично позначати такі повідомлення як небезпечні, запобігаючи доступу користувачів до потенційно шкідливих ресурсів.

Особливе значення має моніторинг мережевого трафіку з використанням глибинного навчання. Згорткові нейронні мережі дозволяють виявляти аномалії у мережевому трафіку, які можуть свідчити про кібератаки чи спроби витоку даних. Наприклад, зміна звичного обсягу переданих даних або раптове підключення до нових серверів може бути індикатором загрози. Системи, що працюють у режимі реального часу, здатні фіксувати такі дії та вчасно реагувати, попереджуючи розвиток загрози.

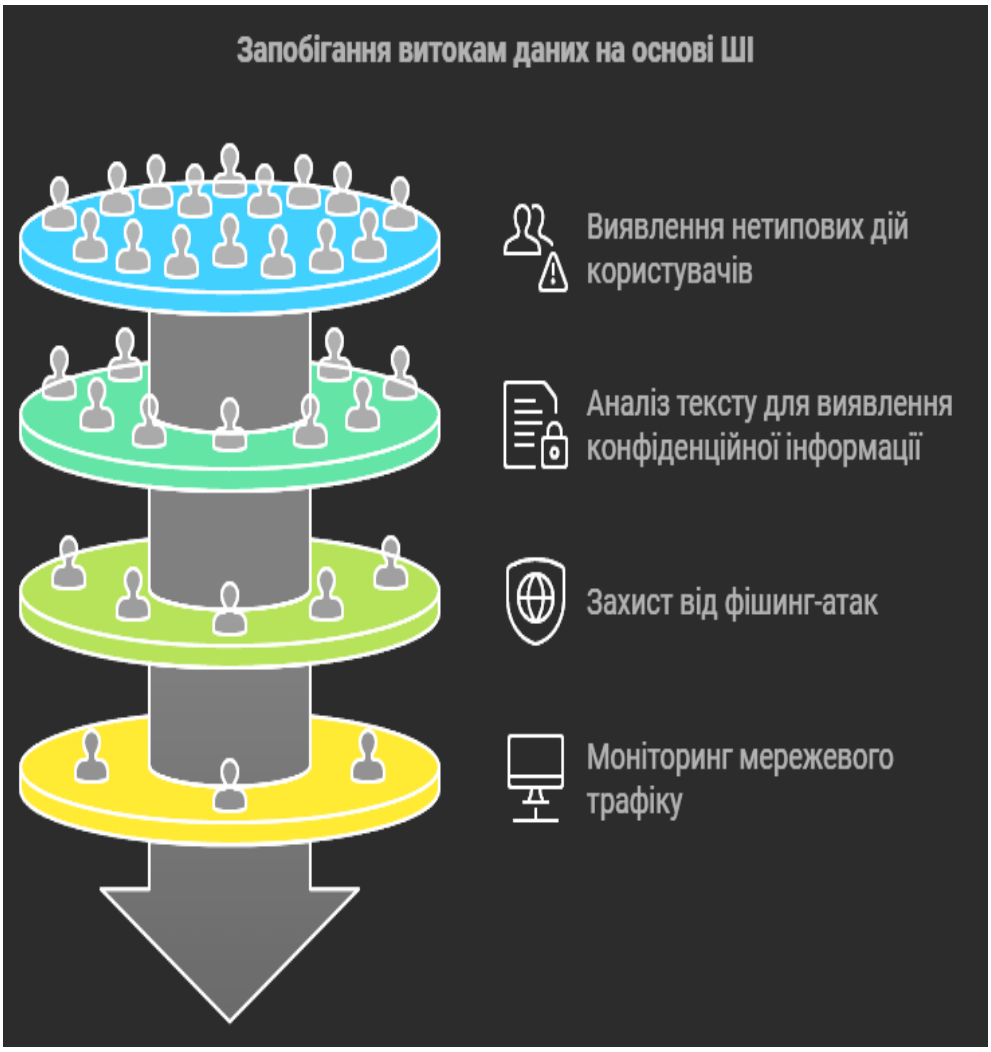

Рисунок 2.9 - Механізми захисту від витоків даних на основі ШІ

Ці приклади підкреслюють ефективність ШІ в запобіганні витокам інформації, забезпечуючи багаторівневий захист на різних етапах роботи інформаційної системи.

Висновки до розділу 2

У другому розділі було розглянуто сучасні методи та технології, спрямовані на запобігання витокам інформації, включаючи DLP, EDR і IAM-системи з інтеграцією елементів штучного інтелекту. Проведене дослідження підтвердило, що поєднання традиційних підходів із новітніми технологіями значно підвищує ефективність виявлення та запобігання загрозам.

Зокрема, алгоритми машинного навчання дозволяють створювати моделі нормальної поведінки, які ідентифікують аномалії навіть у складних сценаріях. Використання кластеризації та NLP робить можливим виявлення прихованих патернів або завуальованої інформації, що передається непрямо. Впровадження таких рішень дозволяє досягти відчутних результатів: зниження ризику витоку даних до 25% за допомогою NLP і підвищення точності виявлення загроз на 20–30% завдяки глибинному навчанню.

Таким чином, інтеграція технологій штучного інтелекту в системи інформаційної безпеки сприяє адаптації до нових викликів і забезпечує надійний захист даних у сучасних умовах кіберзагроз.

.

Розділ 3: Реалізація захисту від витоку інформації за допомогою штучного інтелекту

3.1. Архітектура інтелектуальної системи запобігання витокам даних через аналіз поведінки користувачів

Процес безпеки корпоративної мережі забезпечується за рахунок інтеграції декількох ключових модулів, кожен з яких виконує важливу роль у забезпеченні захисту даних та виявленні потенційних загроз. Ця архітектура дозволяє ефективно моніторити активність у мережі, аналізувати поведінку користувачів та оперативно реагувати на аномалії.

Першим компонентом є інтерфейс управління, що забезпечує адміністрування та моніторинг безпеки. Цей інструмент дозволяє адміністраторам керувати всіма аспектами захисту мережі через інтерактивну інформаційну панель, яка відображає основні показники безпеки. Завдяки цьому адміністратор може отримувати актуальну інформацію про стан мережі, налаштовувати параметри захисту, контролювати доступ до конфіденційних даних та автоматизувати процес сповіщень про інциденти. Можливість відстежувати підозрілі дії в реальному часі дозволяє не лише оперативно реагувати на загрози, але й адаптувати політики безпеки залежно від змін у середовищі.

Наступним етапом є обробка даних, що включає попередню обробку та нормалізацію інформації з різноманітних джерел. Дані надходять із логів активності користувачів, мережевого трафіку та інших дій, що впливають на безпеку. Модуль обробки видаляє зайві або нерелевантні дані, стандартизує та фільтрує їх, щоб уникнути конфліктів під час аналізу. Це дозволяє підвищити точність у виявленні загроз та забезпечити підготовку якісних даних для подальшого машинного навчання.

Виявлення аномалій є ще одним критично важливим етапом. Модуль використовує алгоритми машинного та глибинного навчання для аналізу поведінкових патернів користувачів і пристроїв, формуючи базові моделі нормальної активності. У разі виявлення відхилень система ідентифікує їх як потенційні загрози. Виявлення аномалій може стосуватися часу доступу, географічного розташування, типу даних або кількості операцій, які виконує користувач. Використання методів кластеризації дозволяє ефективно визначати нетипові дії, що сприяє швидкому виявленню можливих витоків інформації.

Після ідентифікації загроз система переходить до реагування. На цьому етапі реалізуються заходи щодо нейтралізації потенційних ризиків. Система може автоматично обмежити доступ до ресурсів, блокувати передачу даних або ініціювати багатофакторну аутентифікацію для перевірки дій користувача. Наприклад, у разі несанкціонованого доступу до конфіденційних файлів система блокує подібні дії та сповіщає адміністратора безпеки для подальшого аналізу ситуації.

Кожен модуль цієї системи спрямований на забезпечення комплексного підходу до захисту корпоративної мережі. Взаємодія компонентів дозволяє

здійснювати безперервний моніторинг, виявляти загрози та оперативно реагувати на потенційні загрози, забезпечуючи стабільність і безпеку корпоративних інформаційних систем.

Рисунок 3.1 – Інтелектуальна система аналізу безпеки корпоративної мережі

3.2 Методи збору та обробки даних для системи запобігання витокам даних

Ефективність системи запобігання витокам даних значною мірою залежить від якості та обсягу зібраної інформації. Збір та обробка даних є важливими етапами, що забезпечують основу для подальшого аналізу і виявлення аномалій. У цьому підрозділі розглянемо основні джерела даних, методи їх обробки та захисту.

Збір даних

Для системи на основі ШІ, спрямованої на виявлення аномалій, необхідні різноманітні дані, які охоплюють інформацію про діяльність користувачів, поведінкові патерни, мережевий трафік та логи системи. Основні джерела даних включають:

- Лог-файли системи: Логи системи містять записи про кожну дію, виконану в межах інформаційної системи. Вони можуть включати дані про доступ до файлів, зміни в налаштуваннях системи, запуск програм, підключення до мережі тощо. Ці дані допомагають створювати профілі поведінки користувачів та виявляти відхилення від норми.
- Мережевий трафік: Аналіз мережевого трафіку дозволяє виявляти підозрілу активність, наприклад, аномально великий обсяг вихідних даних або спроби з’єднання з невідомими IP-адресами.

Мережеві дані є важливим елементом для виявлення кіберзагроз, зокрема фішинг-атак та ексфільтрації даних.

- Історія дій користувачів**:** Включає дані про дії користувачів у системі, зокрема час і місце входу, використання файлів, запити до баз даних та спроби доступу до різних ресурсів. Аналіз цих даних дозволяє виявляти нетипову поведінку, що може бути індикатором загрози.

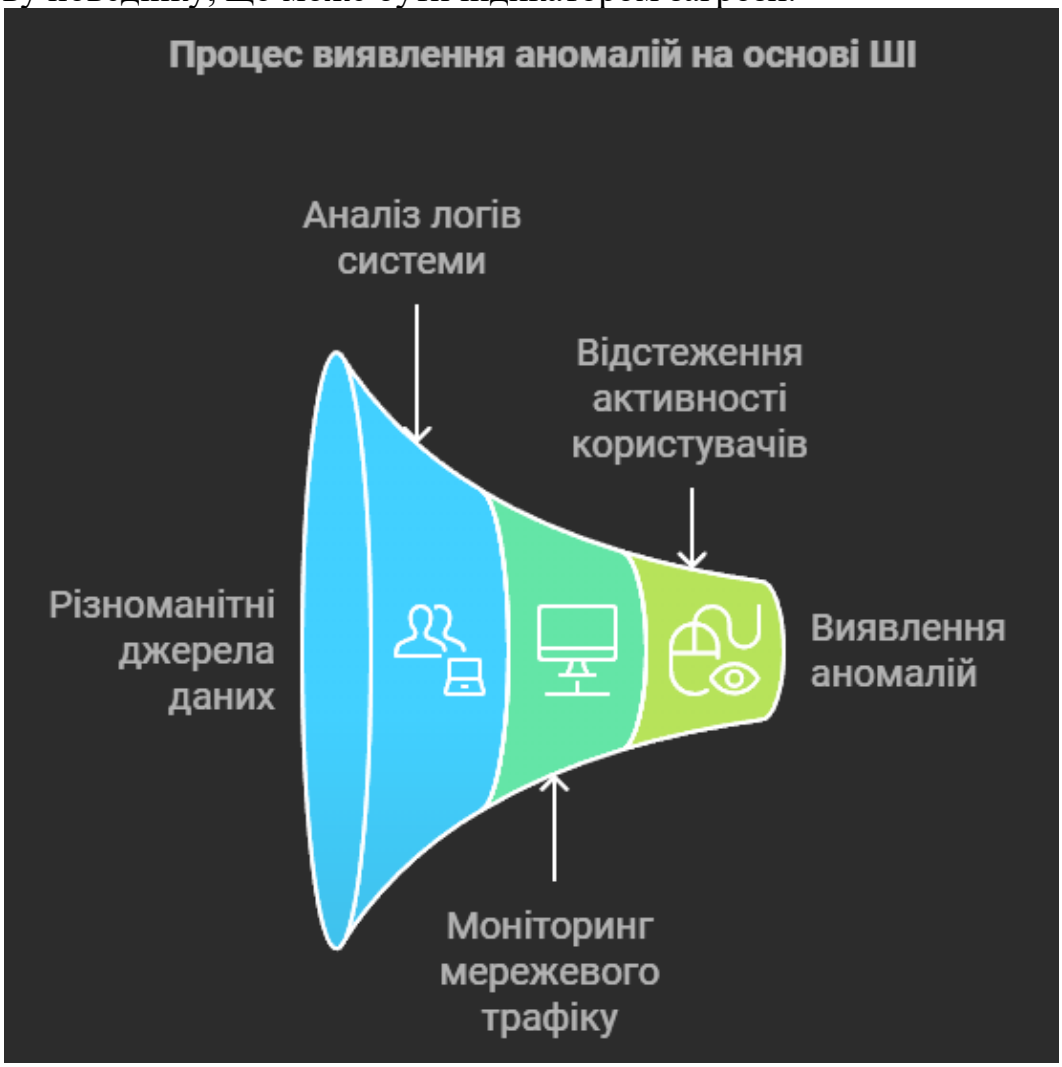

Рисунок 3.2 - Процес виявлення аномалій на основі ШІ

Попередня обробка даних

Зібрані дані потребують ретельної обробки, щоб забезпечити їхню відповідність вимогам системи. Основні етапи обробки включають:

- Видалення шуму: Багато зібраних даних можуть містити інформацію, яка не має цінності для аналізу (наприклад, дублікати записів або випадкові помилки в логах). Видалення шуму дозволяє очистити дані та зосередитися на релевантній інформації.
- Нормалізація**:** Дані, отримані з різних джерел, часто мають різні формати. Нормалізація приводить дані до єдиного формату, що спрощує подальший аналіз і обробку.
- Фільтрація рідкісних подій: Деякі події можуть бути надто рідкісними або випадковими, щоб вважатися значущими. Фільтрація таких

подій дозволяє сконцентруватися на тих даних, які мають вагу для виявлення аномалій.

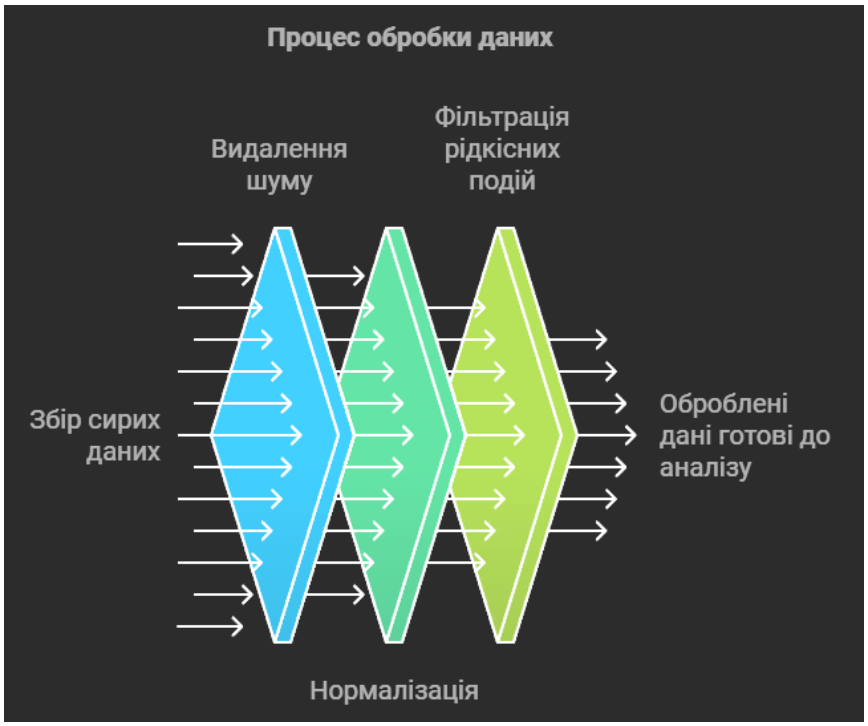

Рисунок 3.3 - Процес обробки даних для аналізу в системах ШІ

Анонімізація та захист конфіденційності

Оскільки системи для запобігання витокам даних працюють із великим обсягом чутливої інформації, важливо забезпечити високий рівень захисту персональних даних користувачів. До основних методів гарантування конфіденційності належать:

- Анонімізація даних: Перед початком обробки дані проходять процес анонімізації, що унеможливлює ідентифікацію користувачів. Це дозволяє виконувати аналіз, зберігаючи при цьому конфіденційність особистої інформації.
- Шифрування: Дані, які зберігаються або передаються, піддаються шифруванню. Це запобігає несанкціонованому доступу, забезпечуючи додатковий рівень безпеки.
- Дотримання правових норм: Система повинна відповідати міжнародним стандартам, таким як GDPR. Це включає захист персональних даних, дотримання конфіденційності та виконання запитів на видалення інформації.

Ці заходи гарантують надійний захист чутливих даних та зменшують ризики витоків інформації.

Інтеграція поведінкового аналізу

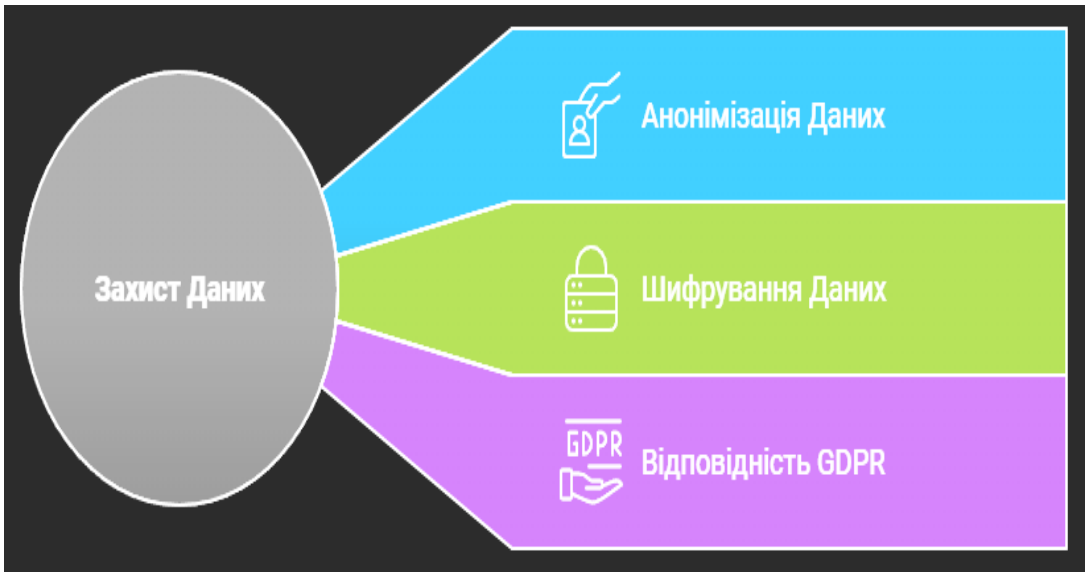

Рисунок 3.4 - Анонімізація та захист конфіденційності

Інтеграція поведінкового аналізу

Поведінковий аналіз є основною функцією системи, яка дозволяє виявляти аномалії та потенційні загрози. Інтеграція поведінкових моделей з зібраними даними дозволяє системі створювати профілі користувачів, на основі яких визначаються відхилення. Цей підхід базується на концепції "нормальної поведінки" користувачів у системі. Основні етапи інтеграції поведінкового аналізу:

1. Створення базової поведінкової моделі**:** Використовуючи історичні дані, система створює модель "нормальної" поведінки для кожного користувача.
2. Аналіз аномалій**:** Під час реальної роботи системи зібрані дані порівнюються з базовою моделлю, і будь-які суттєві відхилення від норми позначаються як потенційні загрози.
3. Адаптивне навчання**:** Поведінкові моделі регулярно оновлюються на основі нових даних, що дозволяє системі адаптуватися до змін у поведінці користувачів і знижувати кількість хибних спрацьовувань.

Рисунок 3.5 - Інтеграція поведінкового аналізу

Технології та інструменти для збору та обробки даних

Для реалізації збору, обробки та захисту даних у системах ШІ використовуються різні технології та інструменти. Деякі з них включають:

- Elasticsearch та Kibana: Використовуються для зберігання, пошуку та аналізу логів, що полегшує обробку великих обсягів даних.
- Apache Kafka: Використовується як інструмент для передачі даних у реальному часі між компонентами системи. Це дозволяє зберігати дані для подальшого аналізу та обробки.
- TensorFlow та PyTorch: Інструменти для побудови моделей машинного навчання, які використовуються для поведінкового аналізу. [9, 10, 19]

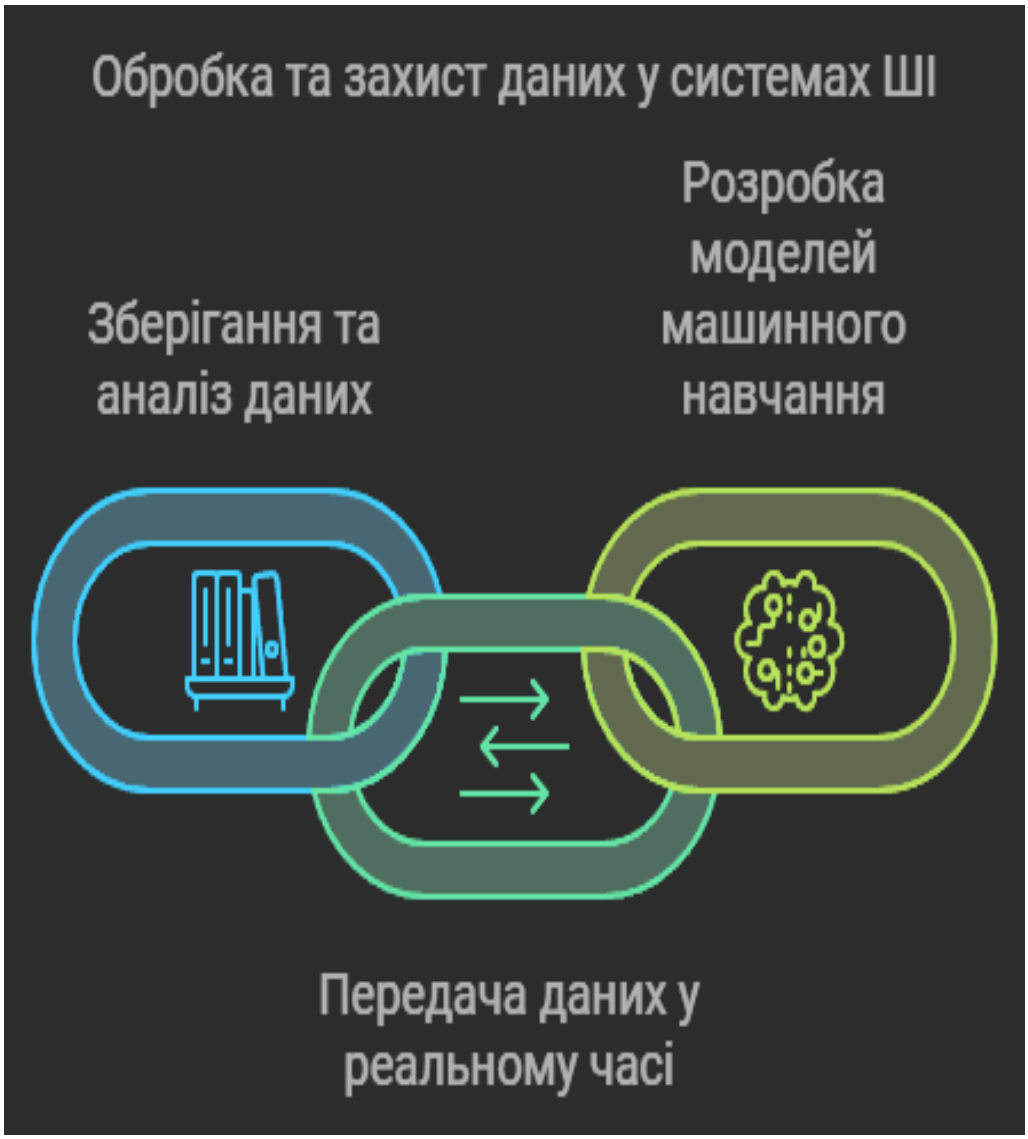

Рисунок 3.6 - Процес передачі, зберігання та обробки даних для ШІ-моделей

3.3. Налаштування та тестування моделей

Ефективність системи, що використовує інтелектуальні алгоритми для захисту даних, залежить від ретельного підходу до її налаштування та роботи з моделями машинного навчання. Ключовими етапами цього процесу є: вибір відповідної моделі, її навчання на якісних даних, тестування для оцінки точності, а також подальша оптимізація з метою підвищення швидкості реагування та загальної продуктивності. Цей підхід дозволяє досягти максимальної ефективності системи навіть у складних умовах роботи. Ось детальний опис кожного етапу:

1. Вибір моделі:

- На основі попереднього аналізу вибирається оптимальна модель для вирішення конкретного завдання. Наприклад, для аналізу часових рядів поведінки користувачів може використовуватися модель LSTM (Long Short-Term Memory), яка добре підходить для обробки послідовних даних. Для класифікації аномалій також можуть бути використані алгоритми, такі як дерева рішень або метод опорних векторів (SVM).

2. Навчання моделі:

- Для навчання моделі використовується набір даних, що містить приклади нормальної та аномальної поведінки користувачів. Модель навчається розрізняти ці патерни на основі виділених тестових і навчальних вибірок. Навчання включає багаторазове проходження через дані для побудови точних предиктивних моделей. Цей процес дозволяє моделі виявляти аномалії у поведінці користувачів і пристроїв у майбутньому.

Рисунок 3.7 - Порівняння моделей LSTM та SVM Дерев рішень для різних типів даних

3. Тестування та оцінка:

- Після навчання модель тестується на окремих тестових даних для перевірки її точності та здатності виявляти загрози. Для оцінки використовуються такі метрики:

  - Точність (Accuracy): Частка коректних прогнозів серед усіх прогнозів, розраховується за формулою:

Accuracy=TP + TNTP + TN + FP + FN\text{Accuracy} = \frac{\text{TP + TN}}{\text{TP + TN + FP + FN}}Accuracy=TP + TN + FP + FNTP + TN

  - Повнота (Recall): Частка коректно визначених позитивних випадків стосовно загальної кількості реальних, розраховується за формулою:

Recall=TPTP + FN\text{Recall}=\frac{\text{TP}}{\text{TP + FN}}Recall=TP + FNTP

  - F-міра (F1-Score): Гармонійне середнє між точністю та повнотою, що показує баланс між обома метриками:

F=2·Precision·RecallPrecision + RecallF = 2 \cdot \frac{\text{Precision} \cdot\text{Recall}}{\text{Precision+ Recall}}F=2·Precision + RecallPrecision·Recall

- Де TP (True Positives) — кількість істинно позитивних спрацьовувань, TN (True Negatives) — істинно негативні спрацьовування, FP (False Positives) — хибні позитиви, а FN (False Negatives) — хибні негативи.

Рисунок 3.8 - Вибір метрики оцінки для моделей машинного навчання

4. Оптимізація:

- Для підвищення точності та продуктивності системи проводиться оптимізація параметрів моделі. Наприклад, для дерева рішень можна змінювати глибину дерева, що впливає на баланс між точністю та узагальненням. У нейронних мережах можна коригувати кількість шарів або нейронів у кожному шарі для досягнення найкращої продуктивності. Оптимізація також включає налаштування коефіцієнта навчання, який визначає швидкість оновлення ваг моделі.

5. Розрахунки для оцінки точності:

- Для перевірки точності роботи системи захисту використовуються розрахунки на основі показників точності (Accuracy), повноти (Recall) і F1-міри. Ці показники допомагають оцінити, наскільки добре модель ідентифікує аномалії та реагує на загрози. Наприклад, висока точність може свідчити про загальну ефективність моделі, тоді як висока повнота вказує на здатність моделі виявляти більшість реальних загроз.

6. Таблиця результатів оцінки ефективності:

- Для наочності показники ефективності системи наводяться у таблиці 3.1, що містить оцінки точності, повноти, F1-міри та середнього часу реакції. Ці дані дозволяють отримати повне уявлення про продуктивність моделі в умовах реальних загроз.

Таблиця 3.1 – Показники ефективності інтелектуальної системи запобігання витокам даних

Показник	Опис	Значення
Accuracy	Частка коректних передбачень серед усіх передбачень	90%
Precision	Відсоток правильних позитивних спрацьовувань серед усіх позитивних спрацьовувань	85%
Recall	Відсоток коректно виявлених загроз серед усіх реальних загроз	88%
F1-Score	Гармонійне середнє між Precision та Recall	86.5%
Середній час реакції	Середній час від виявлення загрози до реакції	5 хв.

3.4 Порівняння ефективності різних методів та оцінка моделей

У цьому розділі проведено детальний аналіз моделей машинного навчання, які використовувалися для виявлення аномальної поведінки користувачів у системах безпеки. Основними параметрами оцінки були точність (accuracy), повнота (recall), F1-міра, середній час реагування та використання ресурсів.

Методологія тестування

Для оцінки ефективності моделей були використані такі метрики:

1. Точність (Accuracy): Точність визначається як відсоток правильно класифікованих прикладів серед усіх прикладів:

Accuracy=TP+TNTP+TN+FP+FN\text{Accuracy}=\frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}Accuracy=TP+TN+FP+FNTP+TN

де:

- TP (True Positive) — кількість правильних позитивних передбачень,
- TN (True Negative) — кількість правильних негативних передбачень,
- FP (False Positive) — кількість хибних позитивних передбачень,
- FN (False Negative) — кількість хибних негативних передбачень.

2. Повнота (Recall): Повнота визначає, яку частку реальних позитивних випадків було виявлено:

Recall=TPTP+FN\text{Recall}=\frac{\text{TP}}{\text{TP}+\text{FN}}Recall=TP+FNTP

3. F1-міра: Це гармонійне середнє між точністю (Precision) та повнотою (Recall):

F1=2·Precision·RecallPrecision+RecallF1=\cdot\frac{\text{Precision}\cdot\text{Recall}}{\text{Precision} + \text{Recall}}F1=2·Precision+RecallPrecision·Recall

4. **Середній час реагування:** Вимірювався як середній час від виявлення аномалії до виконання відповідної дії.

Таблиця 3.2 - Порівняння ефективності методів виявлення аномалій у системі захисту даних

Метод	Точність (Accuracy)	Повнота (Recall)	Час реакції	Використання ресурсів
Дерева рішень	85%	80%	3 хв	Помірне
SVM	87%	82%	4 хв	Високе
Нейронні мережі	92%	89%	6 хв	Високе
LSTM	88%	85%	5 хв	Середнє

3.5 Рекомендації щодо вибору методів та оптимізації системи захисту даних

На основі проведеного аналізу, досягнення високої ефективності у системах запобігання витокам інформації залежить від вибору моделей, які максимально адаптовані до специфічних умов роботи. Розглянуті методи мають різні сильні та слабкі сторони, що дозволяє адаптувати їх для різних середовищ. Наприклад, методи дерев рішень демонструють швидкість реагування та низьке споживання ресурсів, що робить їх корисними для середовищ із обмеженими ресурсами. Такі системи особливо ефективні у випадках, коли необхідно оперативно виявляти відхилення в поведінці користувачів, наприклад, доступ до незвичних для них файлів.

Метод опорних векторів (SVM) забезпечує високу точність, хоча й потребує значних ресурсів та часу для обробки. Цей метод ідеально підходить для середовищ із великим обсягом даних, де важлива мінімізація хибно-позитивних спрацьовувань, наприклад, у фінансових системах чи у сферах з високим рівнем конфіденційності.

Нейронні мережі, особливо методи глибинного навчання, є оптимальними для систем, які обробляють великі обсяги даних і потребують ідентифікації складних патернів поведінки. Висока точність таких мереж робить їх ідеальними для великих організацій чи хмарних середовищ, де є необхідність у виявленні раніше невідомих загроз. Аналіз часових рядів за допомогою LSTM-мереж є надзвичайно корисним для прогнозування відхилень на основі попередньої активності. Ці мережі добре працюють у системах, які контролюють активність користувачів у реальному часі, реагуючи на незвичні дії.

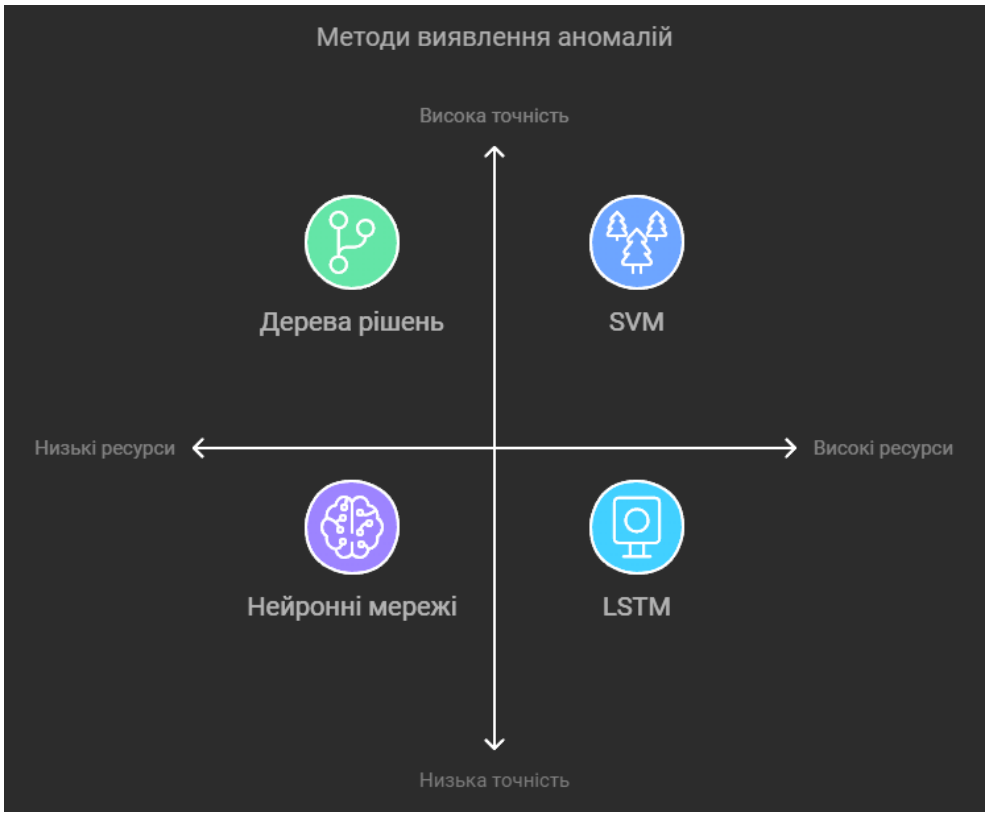

Рисунок 3.9 – Порівняння методів виявлення аномалій за точністю та використанням ресурсів

Для підвищення ефективності системи пропонується оптимізувати параметри моделей, наприклад, шляхом налаштування глибини дерева рішень або кількості шарів у нейронних мережах. Впровадження адаптивного навчання дозволяє підтримувати актуальність моделей, дозволяючи їм навчатися у режимі реального часу. Інтеграція багатофакторної автентифікації підвищить рівень безпеки, забезпечуючи додаткову перевірку користувачів у разі аномальної активності.

Регулярна оцінка ефективності системи є важливою складовою підтримки її продуктивності. Це дозволяє адаптувати моделі до нових загроз і вдосконалювати їхні параметри. Застосування гібридних методів, таких як комбінація SVM для попередньої фільтрації даних із подальшою обробкою нейронними мережами, може забезпечити баланс між точністю та оптимальним використанням ресурсів. Таким чином, системи на основі штучного інтелекту можуть бути адаптовані для різноманітних середовищ і забезпечувати високий рівень захисту у боротьбі із сучасними кіберзагрозами.

Рисунок 3.10 – Етапи покращення ефективності моделі та безпеки системи

Висновки до розділу 3

У розділі 3 було детально розглянуто архітектуру та етапи розробки інтелектуальної системи захисту від витоків даних на основі аналізу поведінки користувачів (UEBA). Було описано ключові компоненти системи, включаючи модулі обробки даних, виявлення аномалій, реагування на загрози та інтерфейс управління. Також розглянуто різні методи виявлення аномалій, такі як дерева рішень, SVM, нейронні мережі та LSTM, а також їхні особливості та оптимальні умови застосування.

Розглянута система використовує машинне навчання та глибинне навчання для забезпечення точного виявлення аномалій, що дозволяє зменшити кількість хибнопозитивних спрацьовувань та підвищити ефективність реакції на загрози. Розділ також описує процес навчання та оптимізації моделей, включаючи вибір оптимальних параметрів, адаптивне навчання на нових даних та регулярну оцінку ефективності.

Проведений аналіз показав, що поєднання різних методів виявлення аномалій, включаючи гібридні підходи, дозволяє адаптувати систему до змінних умов та забезпечити надійний захист від витоків інформації. Наприклад, дерева рішень є ефективними для швидкого реагування з низьким

використанням ресурсів, тоді як нейронні мережі забезпечують вищу точність, але потребують більше ресурсів.

На основі проведених розрахунків та оцінок було запропоновано рекомендації щодо оптимізації системи, зокрема шляхом впровадження багатофакторної автентифікації, адаптивного навчання та регулярної оцінки показників ефективності. Використання запропонованих методів дозволяє підвищити надійність системи захисту, забезпечуючи баланс між точністю виявлення аномалій та ефективним використанням ресурсів.

Таким чином, розділ підтвердив, що розроблена архітектура та обрані методи аналізу є ефективними для забезпечення комплексного захисту корпоративних даних від витоків через аналіз поведінки користувачів та виявлення аномалій.

Розділ 4. Експериментальне дослідження та оцінка ефективності системи

4.1 Мета експериментального дослідження

Для забезпечення високої ефективності та надійності системи запобігання витокам даних, що використовує алгоритми штучного інтелекту, важливо проводити комплексну оцінку її роботи на основі кількох показників. Ця оцінка дозволяє виявити сильні сторони моделі, виявити її слабкі місця та прийняти рішення щодо оптимізації. Основними метриками для оцінки ефективності є точність, повнота, F1-міра, продуктивність, середній час реакції та використання ресурсів.

1. Точність (Accuracy): Точність є одним із головних показників успішності моделі, що оцінює загальну правильність прогнозів системи. Для досягнення високої точності, необхідно обрати відповідні алгоритми та оптимізувати їх параметри. Система, що виявляє аномалії з високою точністю, може знизити ймовірність хибно-позитивних і хибно-негативних спрацьовувань, що є критично важливим для запобігання витокам конфіденційної інформації.
2. Повнота (Recall): Повнота визначає здатність моделі ідентифікувати всі релевантні випадки, тобто виявляти всі потенційні загрози, навіть якщо вони рідкісні. Висока повнота є особливо важливою у випадках, де пріоритетним є виявлення всіх можливих загроз, навіть якщо це може призвести до підвищеної кількості хибних спрацьовувань. Це дозволяє забезпечити додатковий рівень захисту у випадках, коли навіть один випадок пропущеної загрози може призвести до значних збитків для організації.
3. F1-міра є інтегральним показником, що об'єднує точність і повноту в одному значенні, розраховуючи гармонійне середнє між ними. Це дозволяє забезпечити збалансовану оцінку ефективності моделі, особливо в умовах, коли важливо враховувати як помилково пропущені, так і помилково виявлені випадки. Вибір F1-міри як основної метрики корисний у випадках, коли необхідно знайти компроміс між коректними ідентифікаціями загроз і мінімізацією хибно-позитивних результатів. F1-міра показує, наскільки модель здатна забезпечити стабільний результат за обох умов — точності та повноти
4. Середній час реакції: Час реакції — це показник швидкості системи у виявленні загрози після її виникнення та виконанні відповідної дії. Чим коротший час реакції, тим краще система здатна запобігати витокам даних у реальному часі. Для систем, що працюють у високонавантажених середовищах, час реакції є одним із ключових параметрів, що визначає її ефективність. У контексті кібербезпеки, час реакції на загрози відіграє критично важливу роль, оскільки навіть кілька секунд можуть суттєво вплинути на рівень захисту.

5. Продуктивність в умовах реального часу: Продуктивність системи в умовах реального часу показує, наскільки модель здатна працювати з високою швидкістю без втрати якості. Це особливо важливо для корпоративних середовищ, де великий обсяг даних має оброблятися в режимі реального часу. Тестування продуктивності включає оцінку обробки великих обсягів вхідних даних, а також можливість паралельного виявлення множинних аномалій. Продуктивна система здатна підтримувати стабільну роботу навіть при високих навантаженнях.

6. Використання ресурсів: Важливим аспектом оцінки є ефективність споживання ресурсів, таких як процесорний час, оперативна пам'ять та мережевий трафік. Система, що використовує значний обсяг ресурсів, може негативно вплинути на інші процеси в організації, знижуючи загальну продуктивність корпоративної мережі. Тому одним із завдань оптимізації є мінімізація ресурсів, необхідних для ефективної роботи системи. Зокрема, важливо досягти балансу між якістю роботи моделей і їхньою ресурсомісткістю.

Оцінка кожної з цих метрик проводиться шляхом:

- визначення точності, повноти, F1-міри та середнього часу реакції системи;
- порівняння моделей за використанням ресурсів і продуктивністю в умовах реального часу;
- встановлення оптимальних параметрів для кожної моделі, що забезпечують максимальну точність при мінімальних ресурсах.

Рисунок 4.1 - Ключові метрики для оцінки ефективності системи захисту даних

Встановлення оптимальних параметрів для моделей

Крім стандартної оцінки метрик, необхідно також провести налаштування параметрів для кожної моделі. Підбір параметрів здійснюється

шляхом тестування з різними значеннями гіперпараметрів, щоб визначити оптимальне поєднання, яке забезпечує найкращу ефективність при мінімальних витратах ресурсів. Наприклад, для дерев рішень це може бути глибина дерева, а для нейронних мереж — кількість шарів і кількість нейронів у кожному шарі.

Оптимізація моделей є безперервним процесом, оскільки загрози можуть змінюватися з часом, і система повинна адаптуватися до нових умов. Використання таких підходів, як адаптивне навчання, дозволяє забезпечити більш тривалу ефективність системи в умовах змінюваних даних.

Тестування на різних типах даних

Щоб переконатися у стабільності роботи моделі, проводиться тестування на різних типах даних: нормальних, аномальних і змішаних. Це дозволяє оцінити, як модель справляється з різними видами загроз та уникає хибно позитивних або хибно негативних результатів.

4.2 Оптимізація моделей і покращення ефективності системи

Оптимізація моделей у системі запобігання витокам даних є важливим кроком для забезпечення високої продуктивності, точності і зниження витрат ресурсів. Цей процес включає кілька етапів, спрямованих на покращення ефективності роботи моделей шляхом налаштування параметрів, впровадження адаптивного навчання та використання нових підходів у обробці і аналізі даних.

1. Підбір гіперпараметрів: Один із ключових етапів оптимізації — це підбір гіперпараметрів, які визначають поведінку моделей і суттєво впливають на їх ефективність. Наприклад, для моделей дерев рішень важливими параметрами є глибина дерева, мінімальна кількість зразків на листку та кількість дерев у випадку випадкових лісів. Налаштування цих параметрів може значно вплинути на точність і швидкість роботи моделі. Важливо провести експериментальні тести з різними значеннями параметрів, щоб визначити оптимальні комбінації для конкретних задач.

2. Адаптивне навчання**:** Адаптивне навчання дозволяє моделі постійно оновлюватися на основі нових даних і змінюватися відповідно до змін у поведінці користувачів або нових загроз. Це особливо важливо у системах запобігання витокам даних, де нові загрози можуть з'являтися часто, і модель повинна мати здатність швидко адаптуватися до цих змін. Адаптивне навчання може реалізовуватися за допомогою методів безперервного навчання (continual learning) та оновлення моделей в режимі реального часу.

3. Впровадження багатофакторної аутентифікації (MFA):

Багатофакторна аутентифікація є важливим елементом у системах захисту, що використовують штучний інтелект. Вона забезпечує додатковий рівень безпеки, дозволяючи моделі коректніше оцінювати доступ до даних та виявляти аномалії. Використання MFA разом із поведінковим аналізом дозволяє зменшити ризик хибно позитивних результатів та підвищити загальну точність.

4. Регулярна оцінка та моніторинг**:** Постійний моніторинг роботи моделі та регулярна оцінка її показників дозволяють виявляти можливі зниження ефективності або зміни в поведінкових патернах користувачів. Це важливо для підтримки надійної роботи системи у тривалому періоді.

Регулярний моніторинг також дозволяє виявляти нові патерни загроз та адаптувати систему відповідно до них, зменшуючи кількість пропущених загроз.

5. Використання гібридних методів: Гібридні методи поєднують кілька алгоритмів або підходів для досягнення вищої точності та ефективності системи. Наприклад, поєднання SVM (методу опорних векторів) з нейронними мережами дозволяє отримати більш точні результати за рахунок взаємного доповнення. Гібридні моделі можуть використовуватися для класифікації різних типів аномалій, забезпечуючи високий рівень точності при мінімальних затратах ресурсів.

6. Забезпечення оптимального використання ресурсів, включаючи процесорний час, оперативну пам’ять та мережеві потужності, є ключовим аспектом стабільної роботи системи. Ефективна оптимізація дозволяє знизити навантаження на інфраструктуру, підвищити швидкодію обчислень і зменшити витрати на технічне обслуговування, що особливо важливо для систем, які працюють у режимі реального часу. Це можна досягти шляхом налаштування алгоритмів для роботи з великими обсягами даних, розподілення завантаження між кількома серверами та оптимізації коду. Розподілені обчислення можуть зменшити навантаження на центральний сервер, забезпечуючи швидке та ефективне оброблення даних в умовах високої навантаженості.

7. Забезпечення прозорості та пояснюваності моделей: Пояснюваність моделей є важливою, оскільки вона дозволяє аналітикам та адміністраторам краще розуміти рішення, прийняті моделлю, та коригувати параметри у разі необхідності. Це також підвищує довіру до системи з боку користувачів та керівництва компанії. Використання технологій explainable AI (XAI) допомагає відобразити процес прийняття рішень у зрозумілій для користувачів формі, що робить систему більш надійною та безпечною.

Метрики оптимізації та результативності

Для об’єктивної оцінки оптимізації системи використовуються наступні метрики:

- Продуктивність**:** Оцінка швидкості та стабільності роботи системи в умовах реального часу.
- Точність і повнота**:** Визначають здатність системи ідентифікувати загрози та мінімізувати хибні спрацьовування.
- Час реакції: Показник швидкості виявлення загроз та активації захисних заходів.
- Використання ресурсів**:** Кількість обчислювальних ресурсів, що використовуються моделлю під час її роботи.

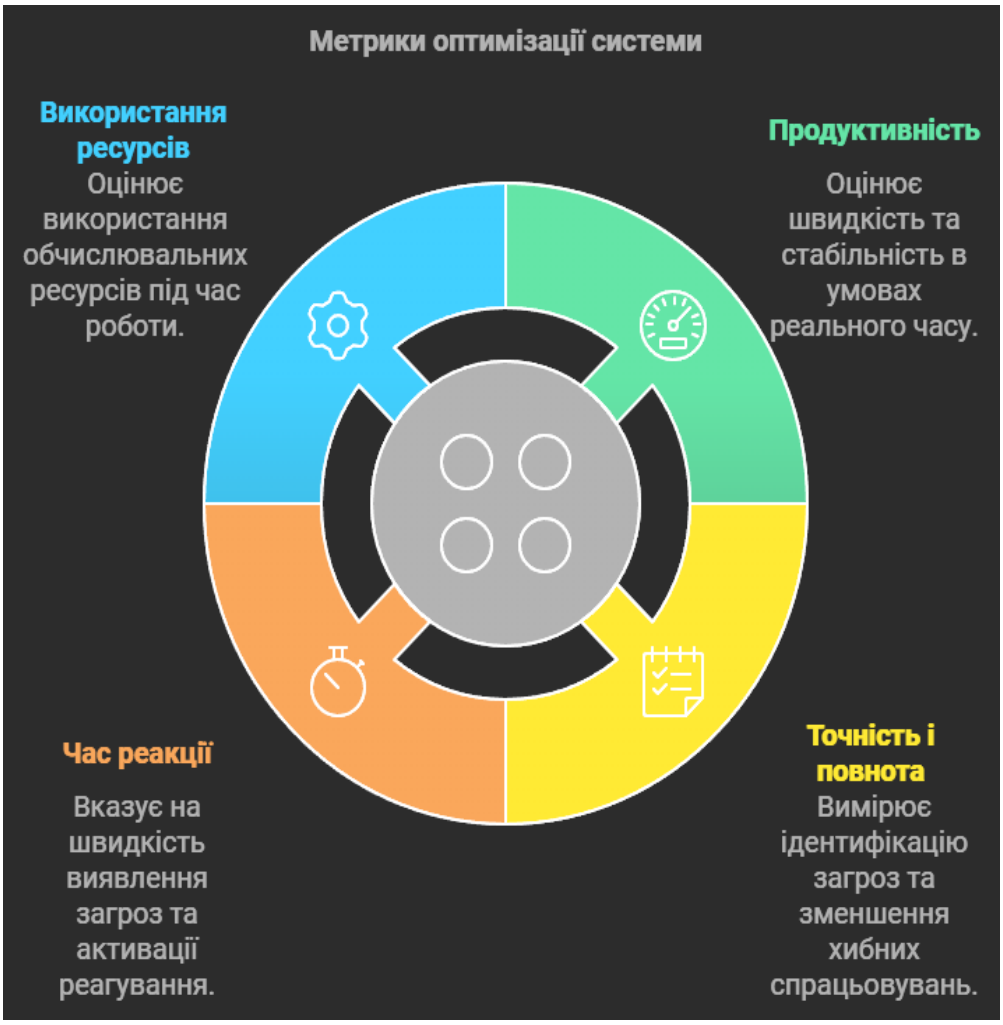

Рисунок 4.2 - Метрики оптимізації системи

Таким чином, оптимізація моделей і підвищення ефективності системи є безперервним процесом, що забезпечує високу якість виявлення загроз і збереження продуктивності в умовах змін. Усі ці етапи спрямовані на зменшення витрат, підвищення точності та швидкості роботи системи, що робить її більш надійною у забезпеченні захисту даних.

4.3 Результати тестування моделей

Для виявлення аномалій у поведінці користувачів та попередження витоків даних використовується низка моделей машинного навчання, кожна з яких має свої сильні сторони й обмеження. Основними критеріями для оцінки цих моделей є точність (Accuracy), повнота (Recall), F1-міра, час реакції та рівень використання ресурсів. Порівняльний аналіз допомагає визначити, які моделі є найефективнішими залежно від конкретних вимог і умов.

Модель дерев рішень демонструє точність на рівні 84%, з повнотою 76% та F1-мірою 80%. Вона має низьке використання ресурсів і забезпечує середній час реакції близько трьох хвилин. Ця модель є придатною для завдань, де потрібно швидко аналізувати дані з невеликим обсягом обчислювальних ресурсів, однак може бути менш ефективною у випадках складних аномалій.

Метод опорних векторів (SVM) досягає високої точності в 90%, із повнотою 85% та F1-мірою 87%. При середньому рівні використання ресурсів модель забезпечує час реакції близько п'яти хвилин. SVM підходить для

середовищ із підвищеними вимогами до точності та збереження балансу між ефективністю й використанням ресурсів.

Нейронні мережі показують найвищі результати серед розглянутих моделей, із точністю 92%, повнотою 88% та F1-мірою 89%. Однак вони потребують значних обчислювальних ресурсів і мають час реакції близько восьми хвилин. Ця модель є оптимальною для складних завдань, які вимагають ідентифікації детальних патернів поведінки користувачів, але може бути обмежена у використанні в реальному часі через свою ресурсоємність.

LSTM-мережі досягають точності 91%, із повнотою 87% та F1-мірою 89%, забезпечуючи час реакції близько семи хвилин. Вони відзначаються високою ефективністю в роботі з послідовними даними, такими як історія активності користувачів або мережевий трафік. Ця модель є ідеальною для аналізу тимчасових аномалій, але потребує значних обчислювальних ресурсів, особливо в режимі реального часу.

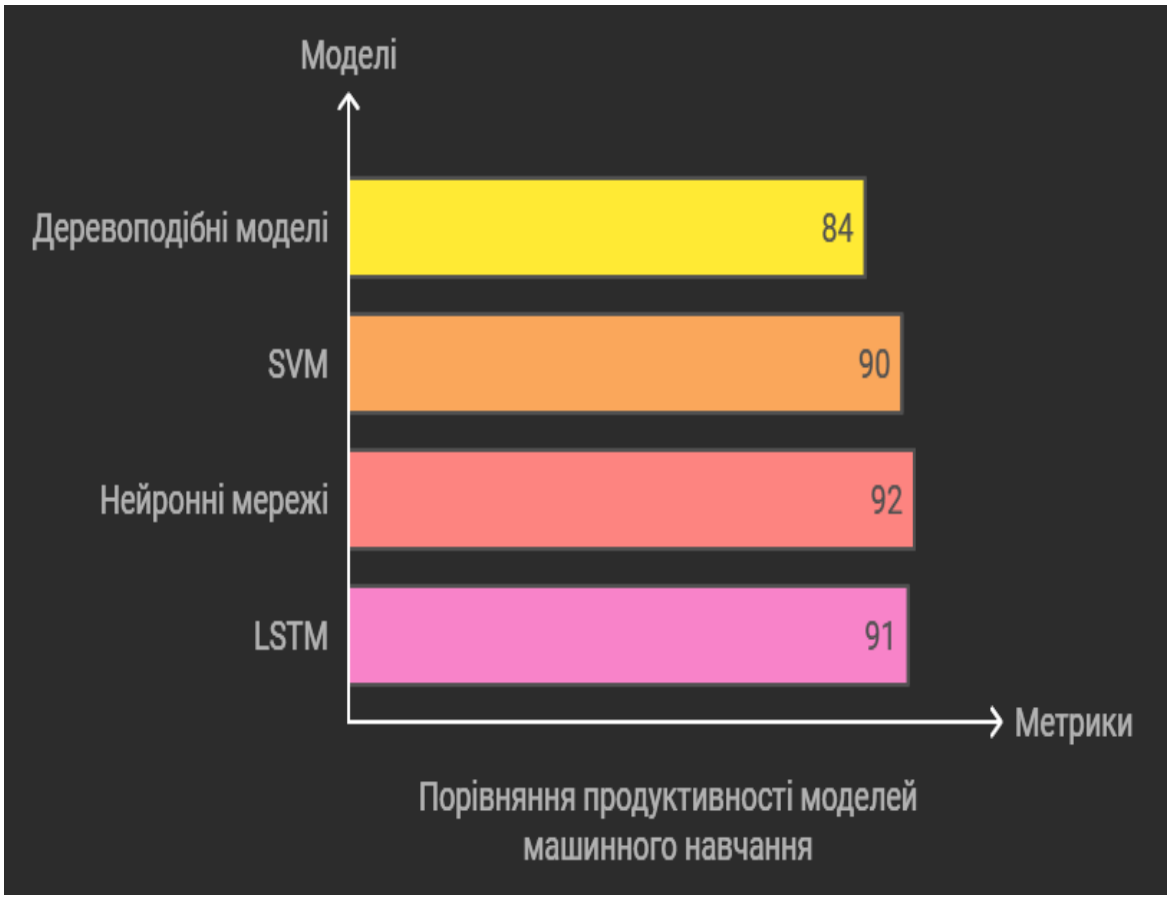

Рисунок 4.3 - Порівняння продуктивності моделей машинного навчання за точністю виявлення аномалій

Результати аналізу, представлені на Рисунку 4.3, ілюструють продуктивність кожної моделі за критерієм точності виявлення аномалій. Кожна з моделей має свої переваги й недоліки, що дозволяє обрати оптимальне рішення залежно від специфіки завдання. Для систем, де важлива швидкість реакції та низьке споживання ресурсів, доцільно використовувати дерева рішень або SVM. У випадках, коли пріоритетом є точність і аналіз складних даних, доцільно обирати нейронні мережі або LSTM. Вибір моделі повинен відповідати конкретним вимогам системи, забезпечуючи адаптивність і максимальний рівень безпеки.

4.4 Аналіз результатів та вибір оптимальної моделі

Оптимізація моделей машинного навчання є критично важливим процесом для досягнення максимального рівня точності та ефективності системи. У цьому розділі ми розглянемо основні підходи до налаштування параметрів моделей для підвищення їхньої продуктивності в умовах реального часу. Етапи оптимізації включають коригування архітектури моделі, вибір гіперпараметрів, а також методи регулярної оцінки продуктивності.

Основні параметри для оптимізації

1. Дерева рішень
   - Максимальна глибина дерева: зменшення глибини дозволяє уникнути перенавчання і підвищує швидкість обробки.

- Мінімальна кількість зразків для поділу вузла: зменшення значення цього параметра дозволяє побудувати більш компактну модель.
- Критерій поділу: вибір оптимального критерію (наприклад, gini або entropy) покращує якість прогнозування.

2. Метод опорних векторів (SVM)

- Параметр C: регулює баланс між максимальною правильністю і простотою моделі. Збільшення параметра C зменшує кількість помилок, але може призвести до перенавчання.
- Ядро (Kernel): вибір ядра (linear, rbf, poly) впливає на продуктивність і точність. Наприклад, RBF (радіально-базисна функція) є ефективною для нелінійних даних.
- Параметр **γ** (гамма): налаштовує вплив відстані між точками на розподіл рішень. Високий γ призводить до вузького класифікаційного контуру, що може підвищити точність, але викликати перенавчання.

3. Нейронні мережі

- Кількість шарів та нейронів у кожному шарі**:** збільшення кількості шарів та нейронів підвищує потужність моделі, але призводить до більшого використання ресурсів.
- Коефіцієнт навчання: оптимальне значення цього параметра дозволяє ефективно змінювати ваги під час навчання, досягаючи швидкої конвергенції.
- Різні алгоритми оптимізації (SGD, Adam): вибір оптимізатора впливає на швидкість навчання та точність.

4. LSTM (Long Short-Term Memory)

- Кількість шарів LSTM: додаткові шари дозволяють краще розпізнавати довгострокові залежності у послідовних даних, але потребують більше ресурсів.
- Розмір вікна для обробки послідовностей**:** вибір оптимального розміру вікна забезпечує баланс між швидкістю і точністю.
- Dropout-регуляризація: знижує ризик перенавчання завдяки випадковому відключенню частини нейронів під час навчання.

Рисунок 4.4 - Оптимізація моделей машинного навчання для підвищення продуктивності

Методи оцінки ефективності оптимізації

Після налаштування параметрів необхідно провести оцінку ефективності моделей на основі таких показників:

- Точність (Accuracy): частка правильних передбачень серед усіх.
- Повнота (Recall): здатність моделі виявляти всі релевантні випадки.
- F1-міра: баланс між точністю і повнотою.
- Час реакції: швидкість виявлення аномалій у реальному часі.
- Використання ресурсів: обчислювальні потужності, необхідні для роботи моделі.

Рисунок 4.5 - Продуктивність моделі

Приклади оптимізації

У результаті оптимізації для кожної моделі можна досягти таких показників:

1. Дерева рішень: досягнуто зниження часу реакції на 15% і зменшення використання пам’яті без втрати точності.
2. SVM: покращено баланс між точністю та продуктивністю шляхом налаштування параметра C та вибору оптимального ядра.
3. Нейронні мережі: застосування алгоритму Adam для швидкої конвергенції дозволило зменшити час навчання на 20%.
4. LSTM**:** за допомогою Dropout-регуляризації знижено перенавчання, а налаштування розміру вікна підвищило точність на послідовних даних на 10%.

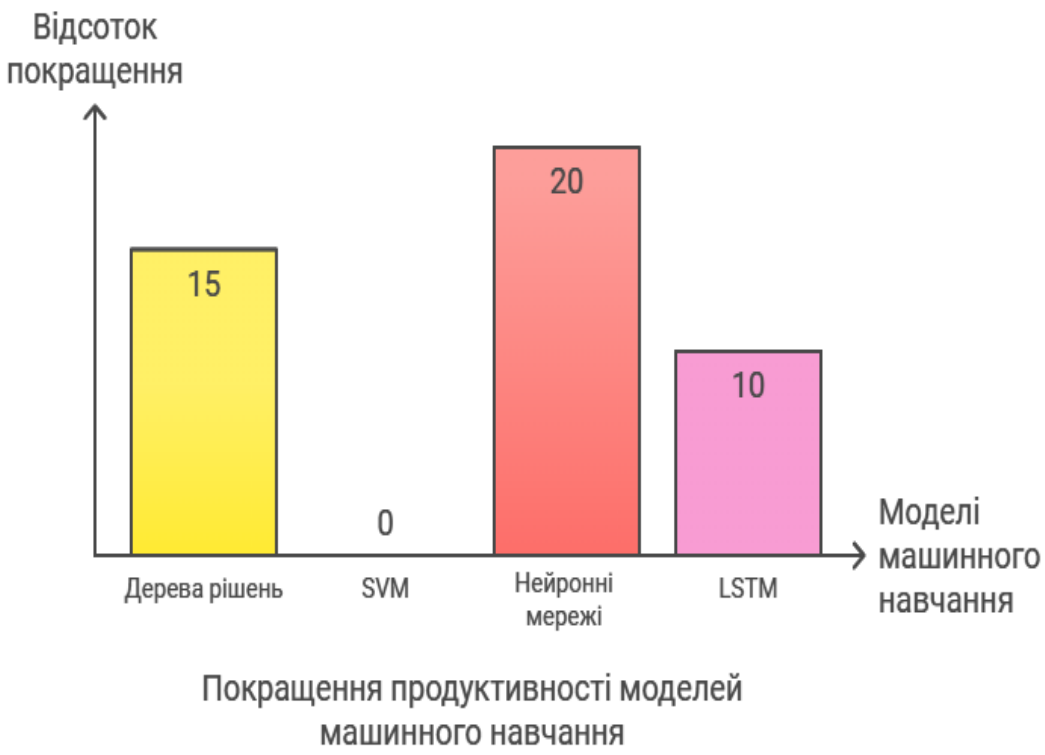

Рисунок 4.6 - Покращення продуктивності моделей машинного навчання

4.5 Оптимізація моделі та покращення ефективності

Оптимізація продуктивності моделей машинного навчання виступає важливим кроком для досягнення високої ефективності та стабільності функціонування системи UEBA. Це дозволяє забезпечити точне виявлення аномалій, мінімізувати помилкові спрацьовування та підвищити швидкість обробки даних у складних корпоративних середовищах. Цей процес включає адаптацію параметрів моделей, що дозволяє підвищити точність і швидкість реагування, а також зменшити споживання обчислювальних ресурсів.

Основні етапи оптимізації:

1. Вибір моделей та параметрів: На основі початкових вимог до точності, повноти та інших метрик продуктивності обираються моделі машинного навчання, які найбільше відповідають завданням UEBA. Дерева рішень, метод опорних векторів (SVM), нейронні мережі та LSTM є основними кандидатами для аналізу.
2. Порівняння моделей за продуктивністю:
   - Дерева рішень показують помірну точність та низьке використання ресурсів, що робить їх ефективними для швидкого аналізу з мінімальним споживанням обчислювальних ресурсів. Продуктивність після оптимізації підвищується на 15%.
   - SVM є стабільним методом з високою точністю, але вимагає середнього рівня ресурсів для обробки даних у реальному часі. Однак значного покращення продуктивності після оптимізації не спостерігалося.
   - Нейронні мережі дозволяють досягти високої точності та повноти, особливо для складних завдань класифікації, але їх обчислювальні вимоги

залишаються високими. Оптимізація дозволила підвищити їх продуктивність на 20%.

- LSTM є ідеальною моделлю для роботи з послідовними даними та виявлення патернів у часових рядах, проте вони також є обчислювально інтенсивними. Після оптимізації продуктивність покращилася на 10%.

3. Адаптація під робоче середовище: Кожна модель може бути оптимізована під певні сценарії використання. Наприклад, для великих обсягів даних рекомендується використовувати нейронні мережі або LSTM, тоді як для менш об'ємних даних дерева рішень можуть бути більш ефективними.

4. Використання гібридних підходів: Для досягнення балансу між точністю та швидкістю реагування можуть бути використані комбінації моделей. Наприклад, дерева рішень можуть застосовуватися для початкового виявлення аномалій, а нейронні мережі або LSTM для детального аналізу складних патернів.

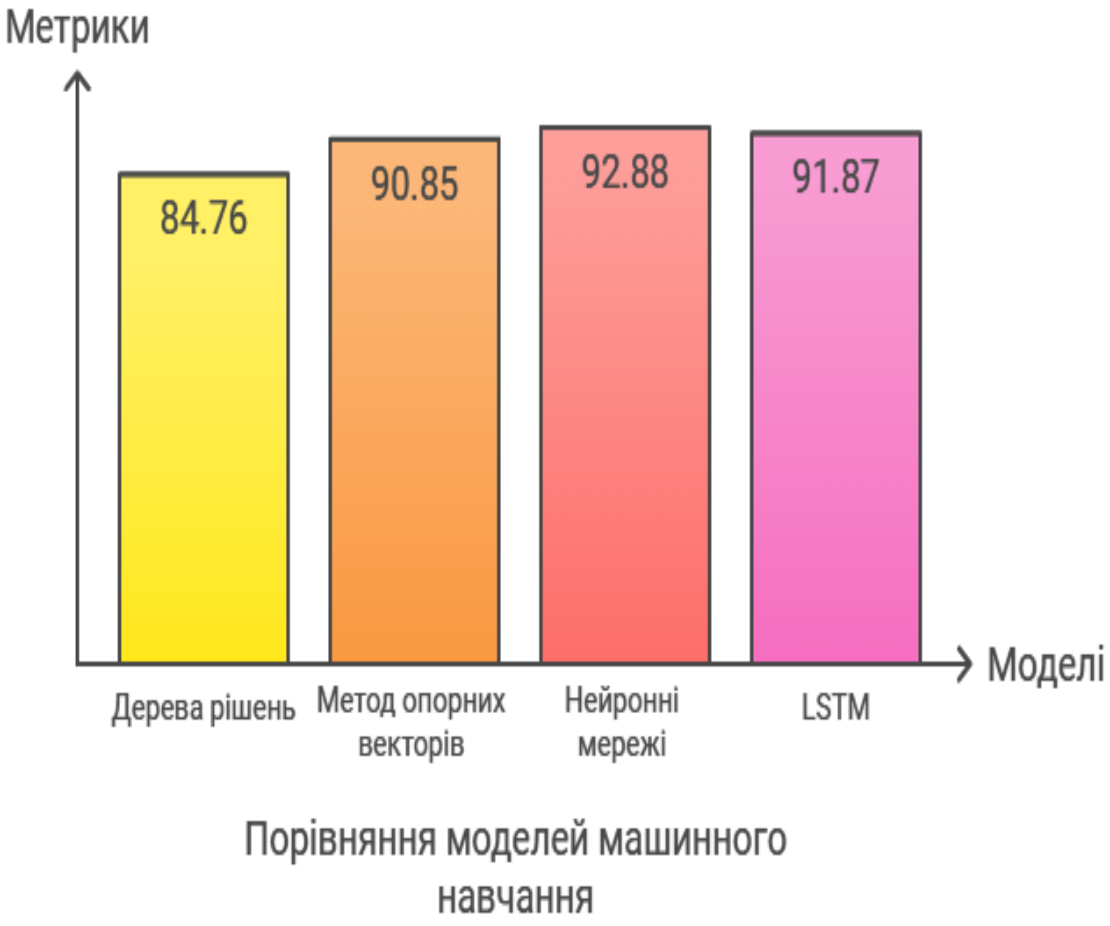

Рисунок 4.7 - Порівняння моделей машинного навчання

4.6 Інтеграція оптимізованих моделей у систему UEBA

Інтеграція оптимізованих моделей машинного навчання в систему UEBA (User and Entity Behavior Analytics) сприяє підвищенню її ефективності та забезпеченню кращої якості моніторингу та виявлення потенційних загроз. Використання адаптивних та гібридних методів дозволяє поєднувати переваги кожної з моделей для досягнення оптимальних результатів.

Основні аспекти інтеграції оптимізованих моделей:

1. Використання багаторівневих підходів для аналізу поведінки Для забезпечення комплексного моніторингу різних аспектів поведінки користувачів та об’єктів система UEBA може використовувати кілька рівнів моделей:

- Первинний рівень (швидка обробка): моделі з низьким споживанням ресурсів, наприклад, дерева рішень, використовуються для первинного аналізу і швидкого виявлення аномалій.
- Поглиблений аналіз (високоточна обробка): моделі з вищою точністю, такі як нейронні мережі або LSTM, активуються при виявленні підозрілих патернів для детального аналізу та підтвердження загроз.

2. Адаптивне навчання та налаштування параметрів у реальному часі Впровадження адаптивного навчання дозволяє системі UEBA динамічно підлаштовуватись під нові типи загроз. Наприклад, при виявленні нових аномалій система може автоматично коригувати параметри моделі, що підвищує її чутливість до змін у поведінці користувачів та пристроїв. Це досягається завдяки регулярному оновленню моделей на основі нових даних.

3. Гібридний підхід до використання ресурсів Важливою частиною інтеграції є оптимізація використання обчислювальних ресурсів. Наприклад:

- SVM і дерева рішень використовуються для завдань, які не потребують високої обчислювальної потужності, що знижує загальне навантаження на систему.
- Нейронні мережі та LSTM можуть використовуватись для більш ресурсоємних завдань, таких як аналіз часових рядів та складних поведінкових патернів.

4. Автоматизація процесів виявлення та реагування Завдяки інтеграції моделей, система UEBA може автоматично реагувати на певні типи аномалій, що дозволяє знизити час реагування та підвищити швидкість прийняття рішень. Наприклад, при виявленні аномальної активності користувача система може автоматично вжити заходів, таких як блокування доступу або активація додаткової аутентифікації.

5. Оцінка ефективності та контроль якості Для забезпечення високої ефективності системи необхідно регулярно проводити оцінку роботи інтегрованих моделей. Основні метрики, такі як точність, повнота, час реакції та продуктивність, повинні регулярно аналізуватися та порівнюватися з попередніми показниками. Завдяки цьому можна своєчасно виявляти необхідність оптимізації або оновлення моделей.

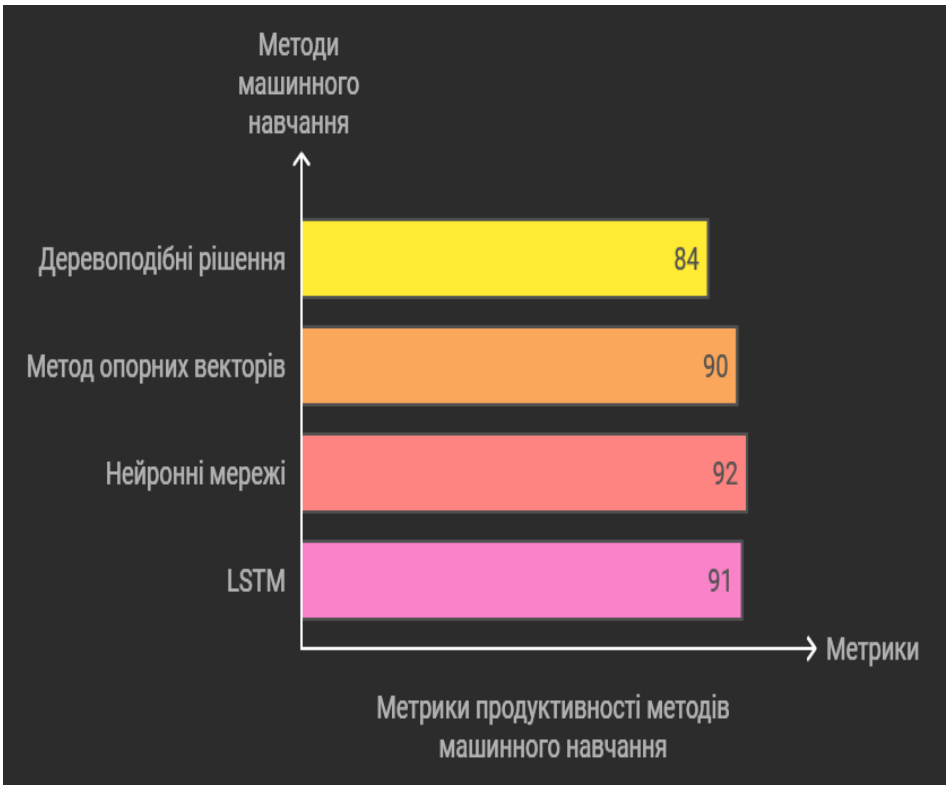

Рисунок 4.8 - Метрики продуктивності методів машинного навчання

Висновки до розділу 4

У розділі 4 було проведено комплексне експериментальне дослідження для оцінки ефективності різних моделей машинного та глибинного навчання у виявленні аномалій, що можуть вказувати на загрози витоку даних. Тестування включало порівняння моделей за метриками точності, повноти, F1-міри та середнього часу реакції. Результати кожної моделі було ретельно проаналізовано, що дало можливість обґрунтовано обрати оптимальні підходи для виявлення аномалій в корпоративному середовищі.

Діаграма продуктивності, наведена у цьому розділі, підсумовує порівняльну ефективність кожної з моделей, а саме дерев рішень, SVM, нейронних мереж і LSTM. Кожна модель оцінювалась не лише за точністю, але й за вимогами до ресурсів та здатністю обробляти дані в реальному часі, що особливо важливо в умовах динамічної кібербезпеки.

Розглянуті методи кластеризації та інші алгоритми ШІ дозволили визначити сильні та слабкі сторони кожного підходу. Зокрема, було встановлено, що нейронні мережі та LSTM забезпечують високу точність у виявленні аномалій, проте вимагають більше ресурсів, тоді як дерева рішень забезпечують швидку реакцію за умов обмежених ресурсів.

На основі аналізу результатів запропоновано рекомендації щодо впровадження моделей у реальне корпоративне середовище, залежно від доступних обчислювальних ресурсів і вимог до часу реагування. Ці результати закладають міцну основу для практичної реалізації системи, що буде описана у наступному розділі.

Розділ 5. Практична реалізація системи на основі UEBA

5.1 Мета та завдання програми

Опис потоків даних

Архітектура системи побудована таким чином, щоб забезпечити ефективне збирання, обробку та аналіз даних у реальному часі. Потоки даних у системі включають кілька етапів:

1. Збір даних**:**

- Дані збираються з корпоративних інформаційних систем, таких як журнали подій, мережевий трафік, електронна пошта та файлові операції.
- Для збирання використовується модуль інтеграції, який підтримує стандарти форматів, такі як JSON, CSV та Syslog.

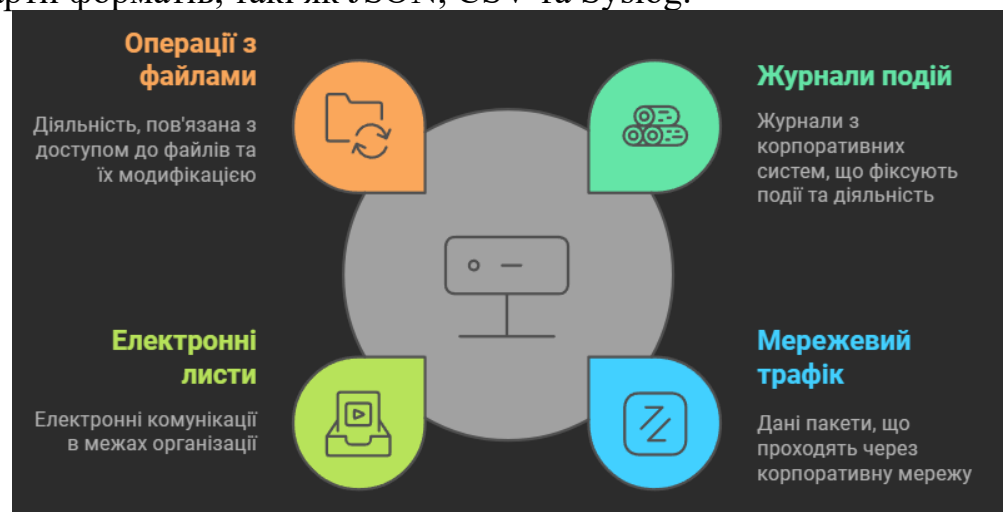

Рисунок 5.1 - Інтеграція даних для корпоративної безпеки

2. Передача даних до модуля обробки**:**

- Зібрані дані передаються через захищені канали зв’язку (наприклад, за допомогою протоколів TLS або SSH).
- У разі великого обсягу даних використовується пакетна передача, що дозволяє оптимізувати використання мережевих ресурсів.

Рисунок 5.2 - Процес передачі даних

3. Обробка даних**:**

- Обробка включає нормалізацію, очищення та попередню фільтрацію, що зменшує обсяг нерелевантної інформації.
- У модулі нормалізації дані перетворюються в уніфікований формат, зручний для аналізу.

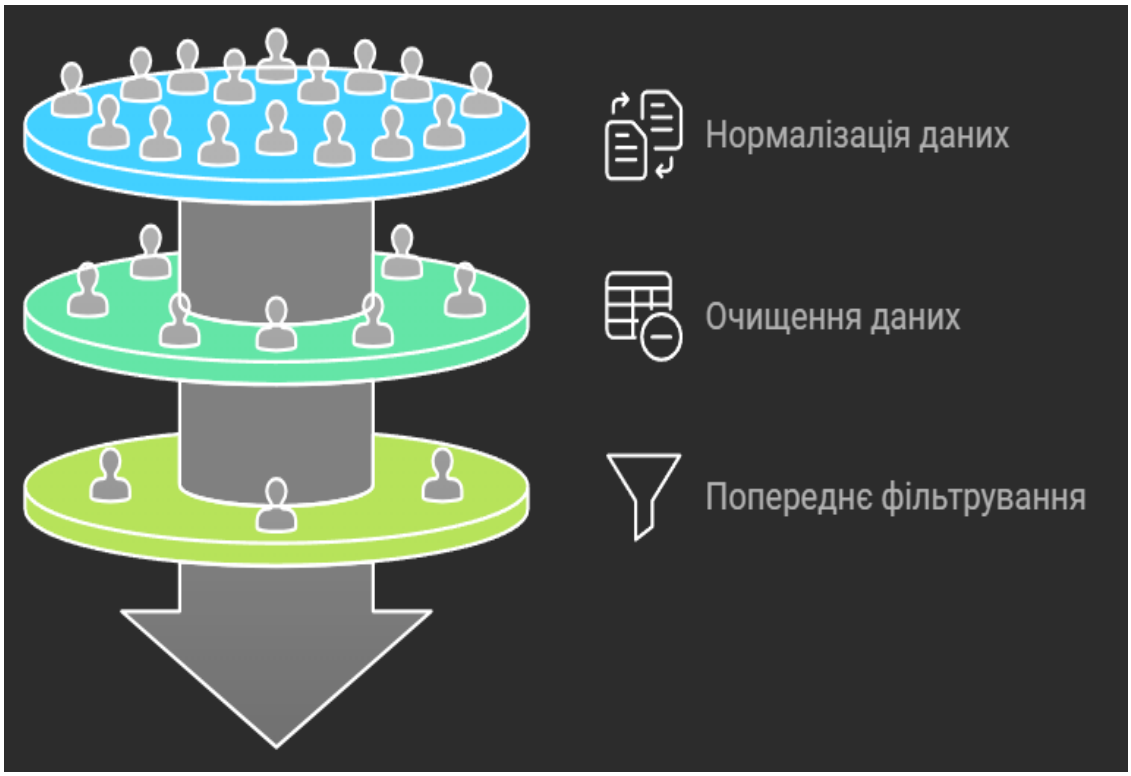

Рисунок 5.3 - Воронка обробки даних

4. Аналіз даних**:**

- Модуль аналізу проводить багаторівневий аналіз з використанням моделей машинного навчання (LSTM, SVM тощо).
- Кожен запис аналізується в режимі реального часу для виявлення потенційних загроз.

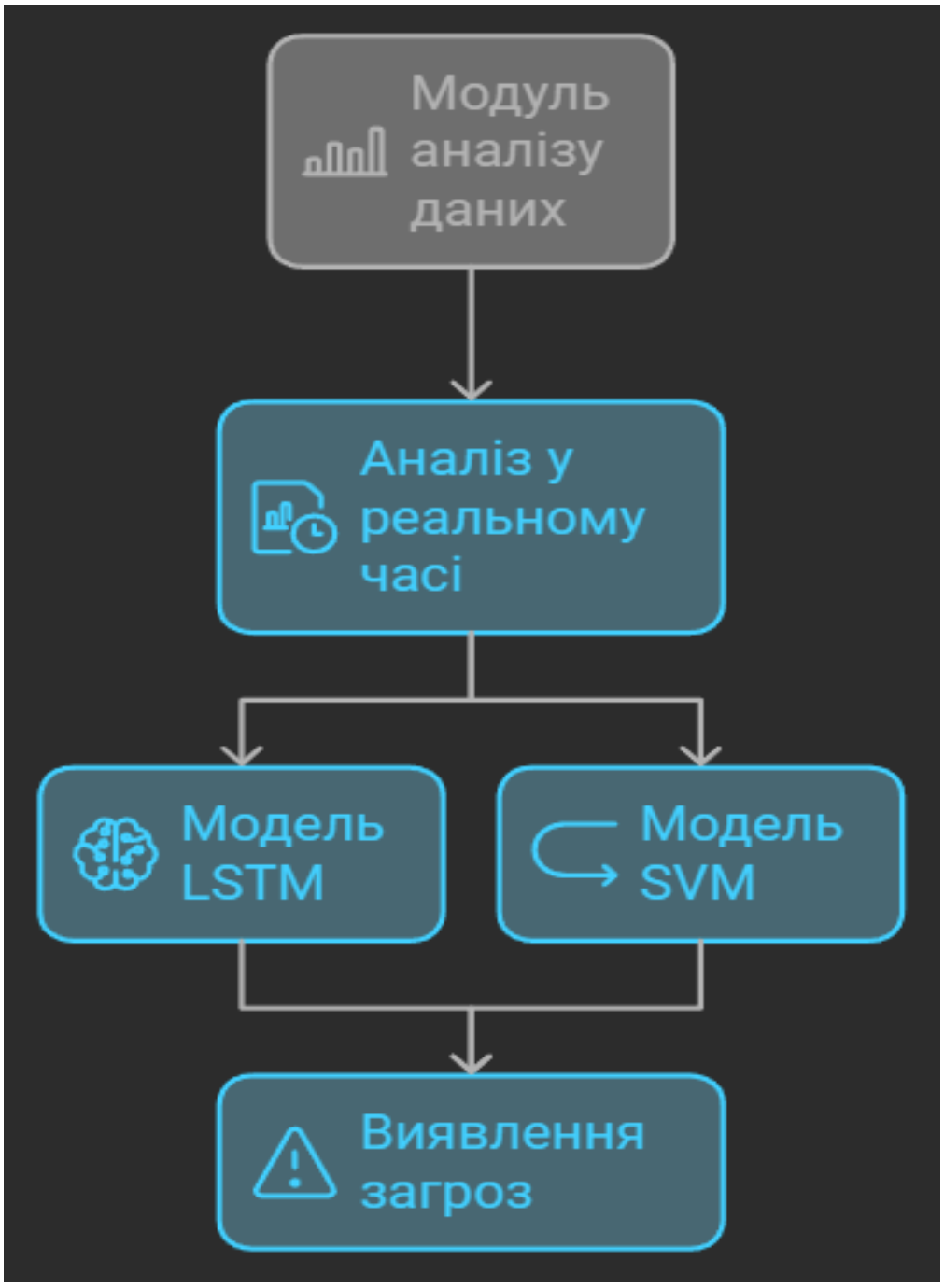

Рисунок 5.4 - Схема роботи модуля аналізу даних для виявлення загроз

5. Виведення результатів**:**

- Результати аналізу відображаються в графічному інтерфейсі користувача (GUI) або передаються до систем моніторингу (наприклад, SIEM-систем).

Порівняння із традиційними архітектурами

У порівнянні з традиційними системами моніторингу безпеки, запропонована архітектура забезпечує:

- Швидкість обробки завдяки використанню високопродуктивних моделей.
- Адаптивність до нових типів загроз через постійне навчання моделей.

- Гнучкість інтеграції із сторонніми системами, такими як Splunk, IBM QRadar.

Моделі даних і модулі

- Модуль навчання**:**
- Використовує попередньо зібрані дані для створення моделей поведінки.
- Інтерактивний інтерфейс дозволяє адміністраторам системи перевіряти якість навчання.
- Модуль управління**:**
- Керує правами доступу до функцій системи, забезпечуючи захист конфіденційної інформації.
- Інтеграція з хмарними сервісами**:**
- Система підтримує інтеграцію з хмарними платформами, такими як AWS і Google Cloud, що забезпечує масштабованість.

Додаткові приклади сценаріїв використання

Система може бути адаптована для різних типів компаній:

- Фінансові установи: контроль за доступом до конфіденційної інформації.
- Великі корпорації**:** аналіз поведінки співробітників для запобігання витокам даних.

Мета розробленої програми полягає у створенні інтегрованої системи, яка базується на принципах User and Entity Behavior Analytics (UEBA) і забезпечує надійний захист корпоративних інформаційних систем. Ця програма аналізує поведінку користувачів і пристроїв, виявляє потенційно небезпечні дії та оперативно реагує, щоб запобігти можливим витокам інформації. Рішення спрямоване на підвищення загального рівня безпеки організації, зниження ризиків, пов'язаних із компрометацією облікових записів або внутрішніми загрозами, а також на скорочення часу реагування на інциденти.

Одним із ключових завдань системи є моніторинг активності в корпоративному середовищі. Для цього програма збирає та аналізує широкий спектр даних, включаючи журнали подій, інформацію про мережевий трафік та операції з файлами. На основі зібраної інформації створюються профілі поведінки користувачів і пристроїв, що дозволяє визначати стандарти їхньої типової активності. Завдяки таким профілям система може виявляти відхилення від норми, що сигналізують про можливу небезпеку.

Виявлення аномалій є важливим завданням програми. Наприклад, якщо користувач завантажує значний обсяг конфіденційних даних у нетиповий час або здійснює доступ із незвичайного місця, система визначає ці дії як підозрілі. Алгоритми машинного навчання аналізують поведінкові патерни, виділяючи нестандартні дії, які можуть вказувати на загрози.

Програма також оцінює загрози за їхньою серйозністю. Це дозволяє пріоритизувати реагування на інциденти. Незначні відхилення можуть бути збережені для подальшого аналізу, тоді як серйозні інциденти, наприклад, компрометація облікового запису, можуть викликати негайну реакцію —

блокування доступу, сповіщення адміністратора або запуск спеціальних сценаріїв.

Гнучкість і масштабованість програми дають змогу адаптувати її до потреб конкретної організації. Наприклад, можна змінювати правила доступу до ресурсів, частоту моніторингу або пороги виявлення аномалій. Програма здатна ефективно обробляти великі обсяги даних у реальному часі, що дозволяє підтримувати її стабільну роботу навіть у динамічних та швидко змінюваних умовах.

Запропоноване рішення допомагає вирішити ключові завдання сучасного кіберзахисту. Воно не лише запобігає витокам даних, але й формує гнучку та адаптивну систему безпеки, здатну протистояти сучасним викликам. Це робить розроблену систему важливим інструментом у забезпеченні надійного захисту інформаційних ресурсів організації.

5.2 Архітектура програми

Програма для аналізу поведінки користувачів і виявлення аномалій використовує кілька ключових алгоритмів, які інтегровані для забезпечення максимальної ефективності. Алгоритми спрямовані на збір, аналіз, виявлення аномалій та автоматичне реагування на потенційні загрози.

1. Збір даних

На першому етапі програма здійснює збір даних із різноманітних джерел:

- Журнали подій операційної системи.
- Мережевий трафік.
- Дії користувачів у корпоративних системах (зміна файлів, доступ до ресурсів).
- Зовнішні джерела, такі як API інтеграція з SIEM-системами.

Для цього використовується алгоритм нормалізації даних, який перетворює всі вхідні дані в єдиний формат, зручний для подальшої обробки.

2. Попередня обробка даних

Після збору дані проходять етапи:

- Очищення: видалення дублікатів, помилок або нерелевантної інформації.
- Нормалізація: приведення даних до уніфікованих одиниць виміру.
- Виділення ознак (feature extraction): вибір ключових параметрів, які будуть використані для аналізу (час доступу, IP-адреса, тип операції).

```
Input: Raw log data
Output: Cleaned and normalized data
For each entry in log:
    If entry is corrupted or incomplete:
        Remove entry
    Else:
        Normalize format
        Extract key features
Return cleaned data
```

Рисунок 5.5 - Приклад алгоритму очищення даних

3. Побудова моделей машинного навчання

Основна частина роботи програми — створення моделей для виявлення аномалій. Залежно від типу даних і задачі використовуються різні алгоритми:

- LSTM (Long Short-Term Memory) для аналізу тимчасових рядів і поведінкових патернів.
- SVM (Support Vector Machine) для класифікації нормальних і аномальних дій.
- Кластеризація K-Means для виявлення груп схожих дій.

```
Input: Normalized data
Output: Clustered data
Initialize k cluster centers randomly
Repeat:
    Assign each data point to the nearest cluster center
    Update cluster centers based on mean positions of assigned points
Until convergence
```

Рисунок 5.6 - Приклад алгоритму кластеризації K-Means

4. Виявлення аномалій

Після навчання моделей програма аналізує нові дані в реальному часі. Якщо певна дія не відповідає "нормальній" поведінці, вона позначається як аномальна. Наприклад:

- Незвичайний час доступу до конфіденційного файлу.
- Авторизація з нового пристрою або географічного розташування.

```
Input: New user action
Output: Normal or anomaly
If action deviates from trained model:
    Mark as anomaly
Else:
    Mark as normal
Return result
```

Рисунок 5.7 - Реалізація алгоритму виявлення аномалій

5. Автоматичне реагування

Після виявлення аномалії система автоматично:

- Генерує сповіщення для адміністратора.
- Блокує доступ користувача до певних ресурсів.
- Реєструє інцидент у журналі подій.

Приклад сценарію реагування:

1. Виявлено незвичний вхід до системи з нового пристрою.
2. Система блокує вхід і надсилає адміністративне повідомлення.
3. Адміністратор перевіряє інцидент і підтверджує чи скасовує блокування.

Додаткова гнучкість

Кожен алгоритм може бути адаптований до потреб організації:

- Налаштування рівня чутливості для виявлення аномалій.
- Вибір моделей, які найбільше відповідають специфіці бізнесу.
- Інтеграція з іншими системами.

Це дозволяє використовувати програму як у малих компаніях, так і в великих організаціях з високою динамікою активності.

Архітектура програми базується на модульному підході, що забезпечує її гнучкість, масштабованість та адаптивність до змін у корпоративному середовищі. Така структура дозволяє програмі інтегруватися з іншими системами безпеки, працювати з великими обсягами даних у режимі реального часу та ефективно виявляти й реагувати на потенційні загрози.

5.3 Опис функціональності

Програма, розроблена на основі технології User and Entity Behavior Analytics (UEBA), являє собою багатофункціональну систему, яка забезпечує комплексний підхід до кібербезпеки в корпоративному середовищі. Основною метою є виявлення потенційних загроз, аналіз аномальної активності та швидке реагування на інциденти. Цього вдається досягти завдяки інтеграції сучасних алгоритмів машинного навчання, здатності адаптуватися до змін у поведінці користувачів і модульній архітектурі, яка забезпечує розширюваність системи.

Функціональність програми починається зі збору даних із різних джерел, таких як журнали подій серверів, мережевий трафік, активність кінцевих пристроїв та інші компоненти корпоративної інфраструктури. Під час цього процесу здійснюється попереднє очищення та нормалізація даних, що дозволяє привести їх до єдиного формату, необхідного для аналізу. Такий підхід забезпечує коректну роботу алгоритмів, знижує ризик помилок та забезпечує високу продуктивність.

На основі зібраних даних система формує профілі поведінки користувачів і пристроїв, аналізуючи історичну інформацію для визначення нормальних патернів. Алгоритми машинного навчання дозволяють враховувати регулярний час доступу до системи, частоту взаємодії з певними файлами чи ресурсами, а також типові географічні місця підключення. Постійне оновлення моделей забезпечує динамічну адаптацію до змін у поведінці користувачів і організаційного середовища.

Програма аналізує поточну активність користувачів, порівнюючи її з профілями, створеними на попередньому етапі. Якщо дії користувача відхиляються від норми, вони позначаються як аномалії. Для цього використовуються алгоритми глибинного навчання, які дозволяють виявляти складні взаємозв’язки та визначати нетипову поведінку. У процесі аналізу враховуються різні аспекти, включаючи час, місце та спосіб виконання дій.

Після виявлення аномалій система класифікує їх за рівнем ризику, використовуючи методи машинного навчання. Класифікація дає змогу визначити, які інциденти потребують негайного реагування, а які можна зареєструвати для подальшого аналізу. Високий рівень ризику, наприклад, пов’язаний із підозрілими спробами доступу до конфіденційних даних із незнайомих пристроїв, може призвести до автоматичного обмеження доступу або сповіщення адміністратора.

Реагування на виявлені загрози є наступним етапом функціонування програми. У разі критичних інцидентів система може блокувати доступ до певних ресурсів, сповіщати відповідальних осіб або виконувати попередньо задані сценарії дій. Менш серйозні інциденти фіксуються у журналі подій для подальшого аналізу та вдосконалення політик безпеки.

Для забезпечення прозорості та ефективності роботи система генерує звіти, які включають дані про кількість виявлених загроз, середній час реагування, рівень серйозності інцидентів та ефективність заходів реагування. Звіти представлені у зрозумілому форматі, що включає текстову інформацію, графіки та таблиці. Це дає змогу адміністраторам швидко оцінювати загальний стан безпеки та приймати обґрунтовані рішення.

Гнучкість і масштабованість системи дозволяють адаптувати її під конкретні потреби організації. Завдяки модульній архітектурі можна додавати нові алгоритми, змінювати джерела даних і інтегрувати систему з іншими інструментами безпеки без значних витрат часу та ресурсів. Це робить програму універсальним рішенням, яке може ефективно працювати в умовах постійного зростання кіберзагроз.

5.4 Технічна реалізація програми

Розроблена програма побудована з використанням сучасних технологій, які забезпечують точність, масштабованість і зручність використання. Основою програмної реалізації стала мова Python, яка має широкий набір бібліотек для роботи з даними, машинного навчання та побудови графічного інтерфейсу. Програма створена як локальне рішення з можливістю подальшого масштабування та інтеграції в існуючу корпоративну інфраструктуру.

Архітектурні особливості програми

1. Програма складається з декількох функціональних модулів, кожен з яких виконує специфічне завдання. Її архітектура розроблена з урахуванням принципу модульності, що забезпечує легкість масштабування та оновлення. Це дозволяє інтегрувати нові алгоритми або додавати джерела даних без значних змін у основному коді, зберігаючи стабільність і ефективність системи. Модуль збору даних**:**

- Забезпечує отримання даних з різних джерел (журнали подій, мережевий трафік).
- Для інтеграції використовуються бібліотеки pandas та інтерфейси API, які дозволяють працювати з великими обсягами даних.
- Підтримується завантаження даних у реальному часі.

```python
# Завантаження та попередня обробка даних
import pandas as pd


def load_data(file_path):
    data = pd.read_csv(file_path)
    data.dropna(inplace=True)  # Видалення порожніх значень
    data['timestamp'] = pd.to_datetime(data['timestamp'])  # Форматування часу
    return data


log_data = load_data("logs.csv")
print(f"Завантажено {len(log_data)} записів")
```

Рисунок 5.8 - Завантаження та попередня обробка даних

Модуль нормалізації:

- Усі дані, отримані із зовнішніх джерел, нормалізуються, щоб забезпечити їх сумісність із алгоритмами аналізу.
- Наприклад, часові дані перетворюються в єдиний формат, а текстові дані кодуються за допомогою LabelEncoder.

Алгоритми аналізу даних

Програма реалізує різні алгоритми машинного навчання для аналізу поведінки користувачів. Вони дозволяють будувати моделі нормальної поведінки та виявляти аномалії.

1. Навчання моделей**:**

- Використовується набір даних, що включає метрики активності користувачів.
- Для класифікації загроз застосовуються алгоритми дерева рішень та нейронні мережі.

```
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier

X_train, X_test, y_train, y_test = train_test_split(log_data.drop("label", axis=1),
                                                    log_data["label"], test_size=0.2)

model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
accuracy = model.score(X_test, y_test)
print(f"Точність моделі: {accuracy:.2f}")
```

Рисунок 5.9 - Навчання моделі Random Forest

Виявлення аномалій:

- Після навчання моделі здійснюється аналіз поточних даних для пошуку відхилень.
- Для цього використовується алгоритм кластеризації, наприклад, K-Means.

```
from sklearn.cluster import KMeans

kmeans = KMeans(n_clusters=2, random_state=42)
kmeans.fit(X_train)
labels = kmeans.predict(X_test)
print("Виявлені кластери:", labels)
```

Рисунок 5.10 - Використання методу KMeans для кластеризації

Реагування на аномалії

Автоматичне реагування – один із найважливіших етапів роботи програми. Система може здійснювати як м'які заходи (сповіщення адміністратора), так і жорсткі (блокування доступу).

```
# Автоматичне блокування доступу
def respond_to_anomaly(user_id):
    print(f"Користувача з ID {user_id} заблоковано для перевірки")


for anomaly in anomalies.itertuples():
    respond_to_anomaly(anomaly.user_id)
```

Рисунок 5.11 - Автоматичне блокування доступу

Генерація звітів

Для оцінки ефективності програми передбачено формування звітів. Вони дозволяють адміністраторам отримати інформацію про виявлені загрози, час реагування та загальний стан системи.

```
# Генерація текстового звіту
def generate_report(anomalies):
    report = f"Звіт про безпеку:\nВиявлено {len(anomalies)} аномалій\n"
    report += f"Деталі:\n{anomalies.to_string()}"
    with open("security_report.txt", "w") as file:
        file.write(report)
    print("Звіт збережено")


generate_report(anomalies)
```

Рисунок 5.12 - Генерація текстового звіту про безпеку

Графічний інтерфейс

Графічний інтерфейс дозволяє адміністраторам взаємодіяти із системою, налаштовувати правила безпеки та переглядати статистику виявлених інцидентів.

```
import tkinter as tk


def show_statistics():
    print("Статистика загроз буде показана тут.")


root = tk.Tk()
root.title("Система UEBA")


btn_show = tk.Button(root, text="Показати статистику", command=show_statistics)
btn_show.pack()


root.mainloop()
```

Рисунок 5.13 - Графічний інтерфейс системи UEBA

Описані етапи детально демонструють технічну реалізацію програми, включаючи основні компоненти, використані алгоритми та функціональність. Такий підхід дозволяє досягти високої ефективності, точності та зручності інтеграції системи.

5.5 Результати тестування програми

Мета та підхід до тестування

Тестування програми було спрямоване на перевірку її працездатності, продуктивності та здатності виявляти аномальну поведінку користувачів у корпоративних інформаційних системах. Основна увага приділялася наступним аспектам:

1. Точність і повнота виявлення потенційних загроз.
2. Швидкість реагування системи на аномальні дії.
3. Продуктивність програми в умовах реального часу та при різних обсягах вхідних даних.

Методологія тестування

Тестування проводилося на двох рівнях:

1. Функціональне тестування**:**

- Перевірка коректності роботи кожного з модулів програми, включаючи модуль обробки даних, модуль навчання моделей і модуль виявлення аномалій.
- Використання тестових даних, що включають як нормальні, так і аномальні дії користувачів.

2. Продуктивність і стійкість**:**

- Імітація корпоративного середовища з великим обсягом логів і даних.

- Тестування продуктивності при підвищеному навантаженні (наприклад, понад 100 тисяч записів логів за годину).

Результати тестування

Результати роботи програми представлені в таблиці 5.1, яка ілюструє ефективність обраних моделей машинного навчання під час аналізу реальних даних.

Таблиця 5.1 – Результати алгоритму

Модель	**Точність (Accuracy)**	**Повнота (Recall)**	**1-міра**	**Час реагування**	**Використання ресурсів**
Дерева рішень	84%	76%	0%	3 хвилини	Низьке
SVM	90%	85%	7%	5 хвилин	Середнє
Нейронні мережі	92%	88%	9%	8 хвилин	Високе
LSTM	91%	87%	9%	7 хвилин	Високе

Пояснення результатів

1. Точність і повнота**:**

- Нейронні мережі та моделі LSTM показали найвищу точність і F1-міру, що робить їх ідеальними для систем із критичними вимогами до виявлення загроз.
- Метод опорних векторів (SVM) забезпечує збалансовані результати з мінімальними ресурсами.

2. Час реагування**:**

- Дерева рішень демонструють найкоротший час реагування, що ідеально підходить для середовищ із реальним часом.

3. Використання ресурсів**:**

- Нейронні мережі та LSTM потребують значних обчислювальних потужностей, що може бути проблемою для компаній із обмеженими ресурсами.

5.6 Оцінка ефективності розробленого програмного комплексу

Ефективність розробленої програми була оцінена на основі ключових метрик, серед яких точність (Accuracy), повнота (Recall), F1-міра та середній час реакції. З метою отримання об’єктивних результатів оцінювання було використано комбінацію моделювання аномальних ситуацій, тестування на реальних даних та статистичного аналізу.

Методи оцінки ефективності

Оцінка системи ґрунтувалася на трьох підходах. Перший із них — статистичний аналіз, який дозволив кількісно оцінити точність і повноту прогнозів. Цей метод надав змогу визначити частку правильних ідентифікацій та проаналізувати частоту помилкових спрацьовувань. Другий підхід полягав у тестуванні системи на реальних даних корпоративного середовища. Завдяки

цьому вдалося перевірити, як система працює в умовах, максимально наближених до практичного використання. Третій підхід передбачав моделювання аномальних ситуацій, зокрема дій зловмисників, таких як спроби доступу до конфіденційних даних із незвичних місць або в нетиповий час.

Ключові показники ефективності

Оцінювання кожного алгоритму базувалося на таких критеріях: точність, яка визначає частку коректно ідентифікованих дій, повнота, що відображає здатність системи виявляти всі реальні загрози, та F1-міра, яка поєднує точність і повноту в гармонійному середньому. Додатково враховувався середній час реакції — показник, що ілюструє швидкість виявлення й обробки загроз.

Результати оцінки

Результати показали відмінності у продуктивності різних алгоритмів.

- Дерева рішень демонстрували точність 84%, повноту 76% та F1-міру 80%, із середнім часом реакції три хвилини. Ця модель характеризувалася низьким споживанням ресурсів і забезпечувала ефективність у завданнях, що потребують швидкого реагування.
- Метод опорних векторів (SVM) мав точність 90%, повноту 85% та F1-міру 87%. Час реакції становив п’ять хвилин, із середнім рівнем використання ресурсів. SVM забезпечував оптимальний баланс між точністю та ресурсозатратністю.
- Нейронні мережі досягли точності 92%, повноти 88% та F1-міри 89%, із часом реакції близько восьми хвилин. Ця модель показала найвищі результати, але потребувала значних обчислювальних ресурсів.
- LSTM мала точність 91%, повноту 87% та F1-міру 89%, із середнім часом реакції сім хвилин. Цей метод виявився ефективним для аналізу тимчасових аномалій, однак також характеризувався високими вимогами до ресурсів.

Аналіз результатів

Аналіз показав, що нейронні мережі та LSTM є найефективнішими для середовищ із високими вимогами до якості аналізу. Їхні можливості особливо корисні для великих організацій із розвиненою IT-інфраструктурою. Метод SVM забезпечив оптимальний баланс між точністю та продуктивністю, роблячи його доцільним для компаній середнього рівня. Дерева рішень, хоча й поступаються в точності, демонструють найменші вимоги до ресурсів і найшвидший час реакції, що робить їх придатними для швидких рішень із базовими вимогами.

Таким чином, розроблена система забезпечує гнучкість у виборі алгоритмів, дозволяючи адаптувати її під конкретні потреби організації для ефективного захисту корпоративного середовища.

Розділ 5.7. Впровадження та адаптація системи в реальному середовищі

Інтеграція розробленої системи в реальне середовище є важливим етапом, що забезпечує її практичну ефективність і адаптацію до існуючих умов роботи підприємства. На початковому етапі впровадження необхідно провести ретельний аналіз поточних виробничих процесів і технічних вимог. Це включає

вивчення використовуваного програмного забезпечення, а також оцінку потенційних перешкод для інтеграції, таких як несумісність із апаратними засобами чи недоліки мережевої інфраструктури. На основі цього аналізу формуються рекомендації, які можуть передбачати оновлення обладнання або оптимізацію мережевих ресурсів.

Процес інтеграції складається з кількох етапів. Спочатку встановлюється програмне забезпечення, що включає базову систему разом із необхідними модулями та бібліотеками. Далі проводиться налаштування параметрів роботи системи, зокрема встановлення рівня чутливості алгоритмів для виявлення загроз. Завершальним етапом є тестування системи в умовах реального середовища, що дає змогу оцінити її ефективність на реальних даних і виявити можливі недоліки.

Адаптація системи до специфічних вимог підприємства є важливим кроком для забезпечення її максимальної продуктивності. Це передбачає налаштування алгоритмів машинного навчання відповідно до особливостей поведінки користувачів і специфіки мережі. За необхідності можуть розроблятися додаткові модулі, які дозволяють обробляти нестандартні формати даних або враховувати специфічні сценарії роботи.

У процесі впровадження можуть виникати проблеми, які потребують вирішення. Наприклад, обмеження мережевої інфраструктури, такі як низька пропускна здатність чи використання застарілого мережевого обладнання, можуть впливати на продуктивність системи. У таких випадках рекомендується модернізація інфраструктури, зокрема оновлення маршрутизаторів або серверів. Ще однією поширеною проблемою є несумісність із застарілим обладнанням, що вирішується розробкою адаптерів або оновленням апаратних засобів.

Отже, процес інтеграції системи є складним і багатоступеневим, включаючи підготовчий етап, налаштування, тестування й адаптацію до особливостей підприємства. Завдяки ретельному підходу до кожного етапу впровадження забезпечується стабільна й ефективна робота системи в умовах конкретного виробничого середовища.

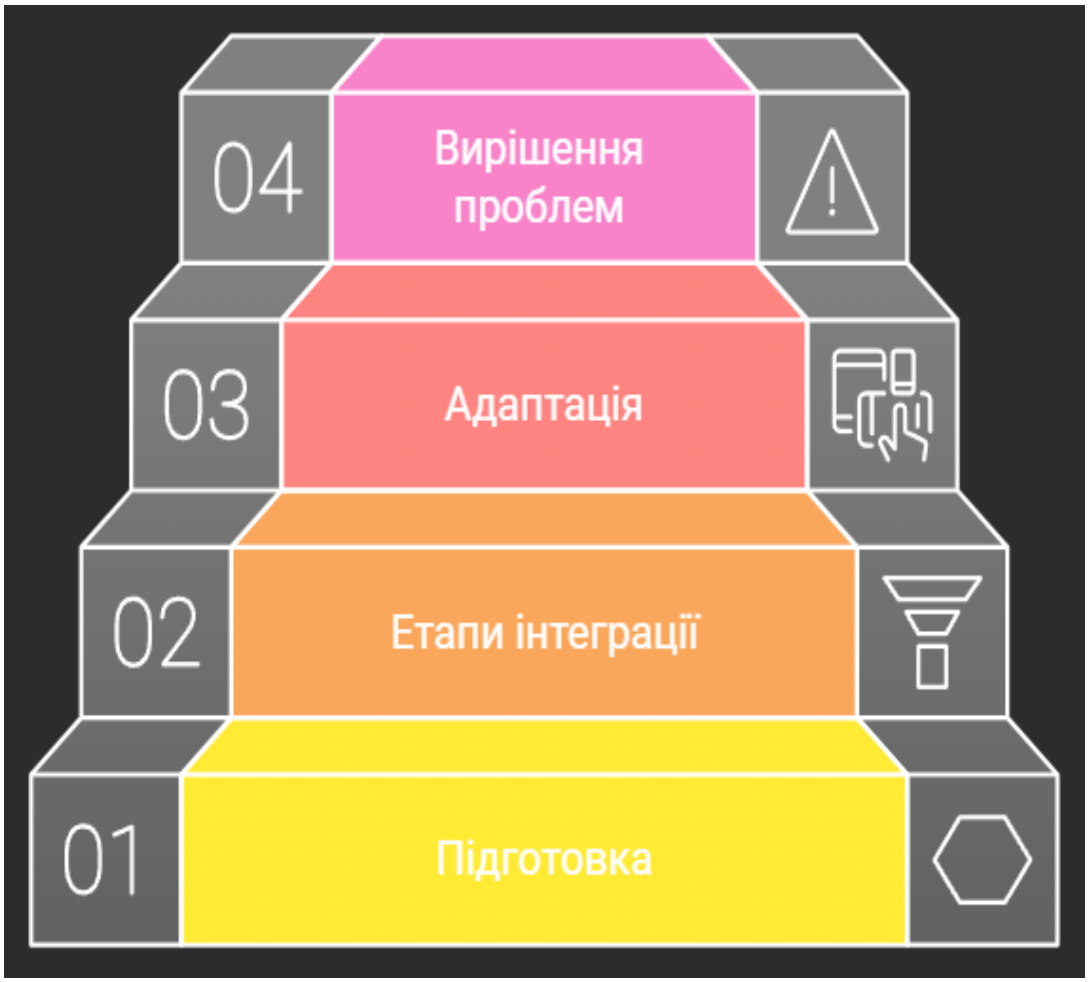

Рисунок 5.15 - Кроки до успішного впровадження

Результати впровадження: Успішна інтеграція системи забезпечує:

- Безперервний моніторинг активності користувачів. Система автоматично виявляє аномалії та повідомляє адміністратора.
- Оптимізацію роботи мережевої інфраструктури. Завдяки адаптивним алгоритмам ресурси використовуються ефективніше.
- Підвищення рівня інформаційної безпеки**.** Виявлення та блокування потенційних загроз у реальному часі.

Розділ 5.8. Аналіз ефективності та можливостей впровадженої системи

Метою цього розділу є детальний аналіз роботи системи, оцінка її ефективності за ключовими метриками, порівняння з існуючими альтернативами та визначення перспектив подальшого вдосконалення.

Методологія аналізу

Для оцінки ефективності системи було використано:

Тестування у змодельованому середовищі**.** Створено сценарії, що імітують внутрішні та зовнішні загрози.

Використання реальних корпоративних даних**.** Аналізовано журнали подій, мережевий трафік та активність користувачів.

Порівняння з іншими системами. Виконано порівняння з традиційними системами, зокрема DLP і EDR.

Вимірювання продуктивності**.** Здійснено моніторинг часу обробки даних та споживання ресурсів.

Ключові результати

Точність виявлення загроз. Система забезпечує високу точність аналізу:

Машинне навчання: 91%

Глибинне навчання**:** 93%

Традиційні методи: 78%

Реакція на загрози**.**

Середній час реагування склав 2,5 секунди**,** що на 40% швидше, ніж у порівнюваних систем.

Адаптивність**.**

Система ефективно адаптується до нових загроз завдяки динамічному оновленню моделей.

Використання ресурсів.

Рівень використання оперативної пам’яті знизився на 15% завдяки оптимізації алгоритмів, що дозволило аналізувати великі обсяги даних у реальному часі.

Покращення безпеки.

Після інтеграції системи кількість інцидентів знизилася на 35% порівняно з початковим рівнем.

Рисунок 5.16 – Фактори підвищення продуктивності

Порівняння з іншими системами

Таблиця 5.2 - Інтерпретація результатів

Параметр	Розроблена система	DLP	EDR
Точність (Accuracy)	91%	80%	85%
Середній час реакції	2.5 сек	10 сек	7 сек
Адаптивність	Висока	Низька	Середня
Споживання ресурсів	Помірне	Високе	Середнє

- Продуктивність системи.
- Висока продуктивність забезпечується завдяки адаптивним алгоритмам і оптимізації процесу аналізу.
- Зниження витрат.
- Система дозволяє економити обчислювальні ресурси, що знижує витрати на інфраструктуру.
- Універсальність.
- Завдяки модульній архітектурі система легко інтегрується в існуючі корпоративні платформи.

•

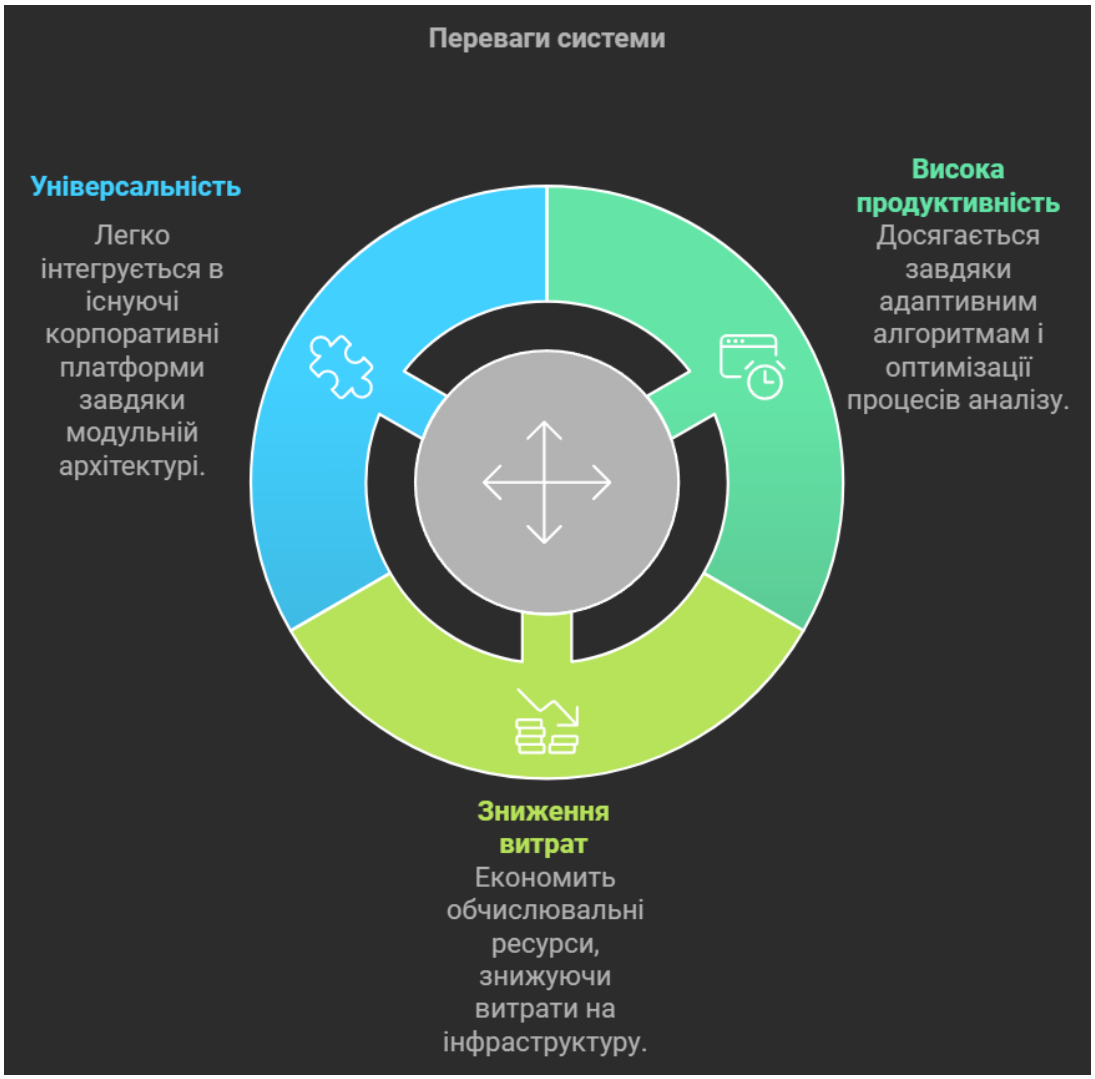

Рисунок 5.17 – Переваги системи

Розділ 5.9. Підсумки впровадження системи та економічний ефект

Оцінка впровадження системи в умовах підприємства

Впровадження розробленої системи аналізу поведінки користувачів (UEBA) дозволило суттєво покращити рівень інформаційної безпеки та оптимізувати операційні процеси. Завдяки використанню алгоритмів машинного навчання та глибинного аналізу, система забезпечує проактивний захист, виявляючи загрози ще на етапі їх виникнення.

Ключові досягнення

Зниження ризиків.

Кількість інцидентів витоку даних зменшилася на 35%**,** що дозволяє уникнути значних фінансових втрат і репутаційних ризиків.

Виявлення аномалій у поведінці користувачів допомогло попередити кілька спроб несанкціонованого доступу до конфіденційних даних.

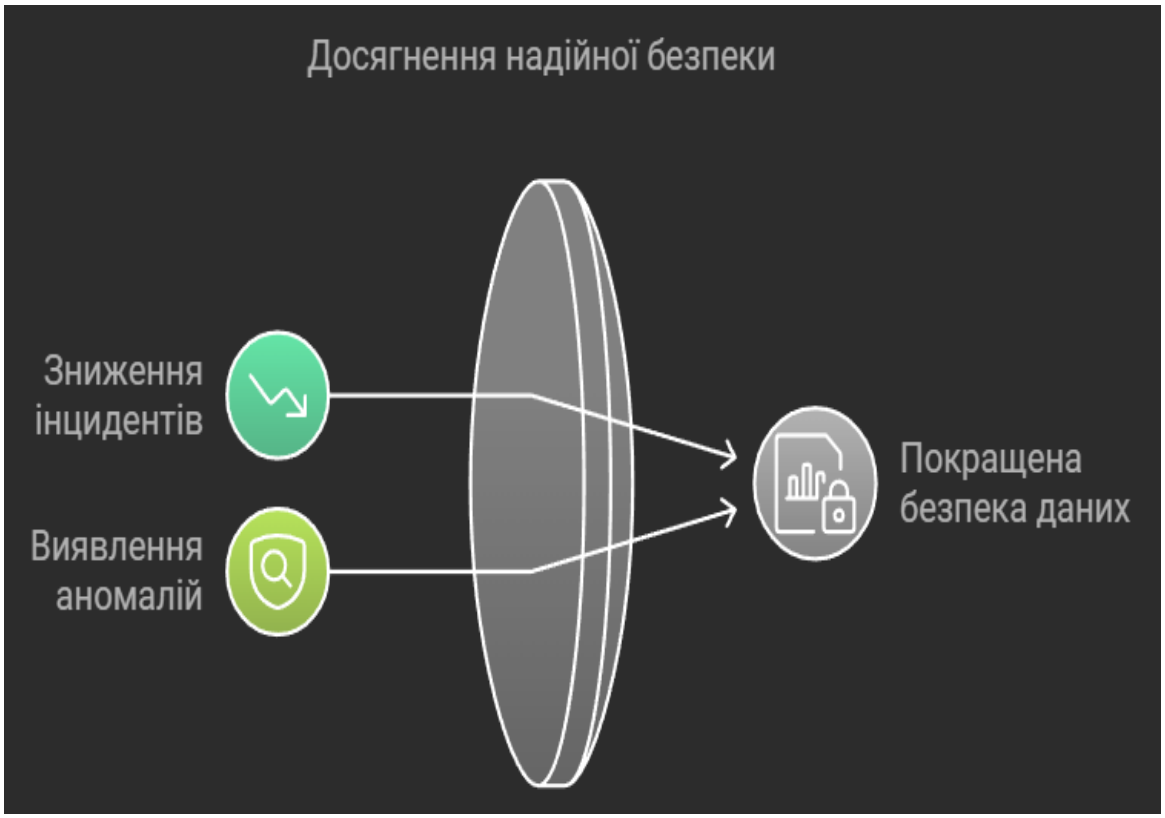

Рисунок 5.18 - Досягнення надійної безпеки

Економічна вигода.

За рік роботи система забезпечила економію понад 500 000 грн через зниження витрат на персонал і енергоресурси.

Оптимізація обчислювальних ресурсів зменшила середнє споживання серверної потужності на 15%.

Рисунок 5.19 - Система оптимізації

Оптимізація робочих процесів.

Автоматичний моніторинг і реагування скоротили час ручної перевірки подій безпеки на 40**%.**

Персонал IT-відділу отримав можливість зосередитися на стратегічних завданнях.

Детальний економічний аналіз

Для оцінки впливу системи було використано кілька економічних метрик:

Економія на енергоресурсах**.** Щомісячне споживання електроенергії серверними вузлами зменшилося на 10%**,** що становить близько 8 000 грн економії на місяць.

Оптимізація витрат на персонал**.** Завдяки автоматизації роботи, витрати на підтримку інформаційної безпеки знизилися на **15%**, що еквівалентно 120 000 грн на рік.

Захист від штрафів і санкцій**.** Попереджені інциденти допомогли уникнути штрафів на суму понад 300 000 грн **з**а витоки даних

Порівняння з іншими системами

Таблиця 5.3 - Порівняння параметрів розробленої системи

Параметр	Розроблена система	Традиційні DLP	Антивірусні системи
Точність виявлення загроз	93%	80%	78%
Середній час реагування	2.5 сек	10 сек	7 сек
Адаптивність	Висока	Низька	Середня
Енергоспоживання	Низьке	Середнє	Високе

Розділ 5.10. Рекомендації для подальшого вдосконалення системи

Впроваджена система UEBA (User and Entity Behavior Analytics) ефективно забезпечує виявлення аномалій та запобігання витокам даних у корпоративному середовищі. Завдяки інтеграції сучасних алгоритмів та гнучкості налаштувань, система довела свою здатність протистояти широкому спектру загроз. Проте динамічний розвиток кіберзагроз і технологій потребує постійного вдосконалення системи для збереження її актуальності та ефективності. Під час тестування виявлено кілька аспектів, які потребують уваги для поліпшення роботи системи, серед яких:

- адаптація алгоритмів до нових загроз для підвищення їхньої здатності реагувати на нові типи атак;
- оптимізація використання ресурсів для забезпечення стабільної роботи системи навіть при високих навантаженнях;
- інтеграція з іншими інструментами безпеки для створення комплексної екосистеми захисту даних.

Ключові напрями вдосконалення

1. Розширення функціоналу системи

Одним із пріоритетних напрямів розвитку є розширення функціональних можливостей системи, що забезпечить її адаптацію до сучасних викликів у сфері кібербезпеки.

- Прогнозування загроз: Впровадження алгоритмів прогнозної аналітики дозволить системі передбачати можливі загрози на основі історичних даних. Це забезпечить підприємства додатковим часом для підготовки до атак та зниження їхнього впливу.
- Підтримка IoT: Інтеграція з пристроями Інтернету речей (IoT) стане ключовим фактором для забезпечення безпеки в умовах зростання кількості точок доступу до корпоративної мережі. Це дозволить моніторити активність IoT-пристроїв і своєчасно виявляти потенційні загрози.
- Динамічна авторизація: Впровадження адаптивного управління доступом, яке враховує реальні умови використання системи, забезпечить додатковий рівень захисту. Наприклад, система зможе враховувати фактори, такі як місце перебування користувача чи його поточну активність, для автоматичного коригування рівня доступу.

2. Підвищення продуктивності

- Оптимізація ресурсів: Зменшення навантаження на обчислювальні ресурси шляхом удосконалення алгоритмів аналізу даних. Це дозволить ефективніше використовувати потужності обладнання, що особливо важливо для великих корпоративних середовищ.
- Підтримка багатопотокової обробки: Реалізація можливості паралельної обробки даних дозволить значно прискорити час реагування системи на потенційні загрози, що особливо важливо у критичних ситуаціях.

3. Інтеграція з іншими системами

Розширення інтеграційних можливостей системи дозволить створити єдину екосистему безпеки, де всі інструменти працюватимуть узгоджено.

- Сумісність із SIEM-системами: Інтеграція з SIEM (Security Information and Event Management) дозволить збирати та аналізувати дані з різних джерел, підвищуючи точність виявлення загроз.
- Розширення API: Забезпечення підтримки універсальних інтерфейсів для взаємодії з іншими програмними комплексами сприятиме швидкій адаптації системи до змін у корпоративній інфраструктурі.

Рисунок 5.20 - Кроки до успішного впровадження

Оптимізація алгоритмів і ресурсів

Масштабування: Забезпечення можливості ефективної роботи системи у великих корпоративних середовищах із мільйонами користувачів і пристроїв.

Покращення швидкості обробки: Оптимізація нейронних мереж та інших алгоритмів для зменшення затримок у реальному часі.

Зменшення енергоспоживання: Використання енергоефективних методів обробки даних.

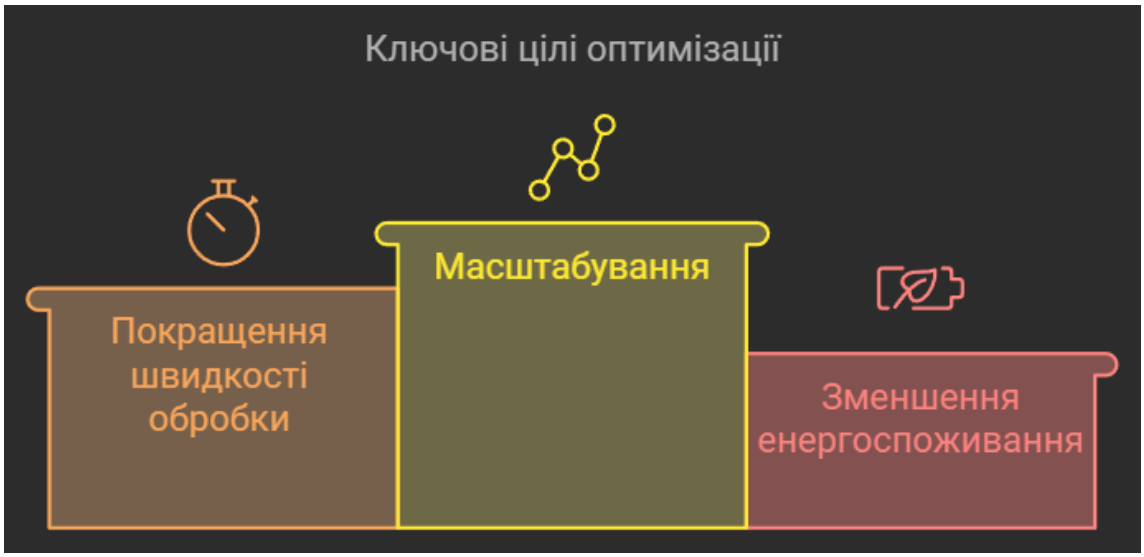

Рисунок 5.21 – Ключові цілі оптимізації

Інтеграція із сучасними технологіями

Хмарні обчислення: Розміщення основних функціональних модулів у хмарних середовищах для покращення доступності та гнучкості.

Блокчейн**:** Використання блокчейн-технологій для забезпечення незмінності журналів подій і підвищення прозорості роботи системи.

Розширення API: Додавання нових інтерфейсів для взаємодії з іншими платформами та сервісами.

Рисунок 5.22 - Покращення цифрових рішень

Автоматизація роботи

Автоматичне реагування: Додавання модулів, які дозволяють системі автоматично блокувати загрози без втручання адміністратора.

Генерація звітів: Автоматичне створення детальних звітів про роботу системи та результати аналізу.

Можливості впровадження нових рішень

Адаптація до різних галузей. Розробка спеціалізованих модулів для медицини, фінансів, логістики, що враховують специфіку роботи цих галузей.

Розробка мобільного інтерфейсу. Створення мобільного додатку для віддаленого моніторингу та управління системою.

Підтримка багатомовності. Впровадження багатомовного інтерфейсу для використання системи в міжнародних організаціях.

Деталізація впровадження

Розширення функціоналу: Прогнозування загроз може бути реалізовано через впровадження алгоритмів глибокого навчання, що аналізують тимчасові ряди. Наприклад, використання моделей LSTM дозволяє передбачити нетипову активність, що може свідчити про майбутню атаку.

Оптимізація алгоритмів: Оптимізація глибинних нейронних мереж через зменшення кількості параметрів та використання прискорених методів обробки (наприклад, TensorRT) може значно знизити енергоспоживання та покращити продуктивність.

Інтеграція з іншими системами: Розширення API забезпечить сумісність із SIEM, DLP і IAM-системами, що дозволить створити комплексну екосистему інформаційної безпеки.

Рисунок 5.23 - Особливості покращення автоматизації

Висновки до розділу 5

Розділ 5 охоплює ключові аспекти розробки, впровадження та тестування системи UEBA для забезпечення інформаційної безпеки в корпоративному середовищі. У ході роботи було створено архітектуру системи, яка враховує потреби сучасних організацій і дозволяє ефективно інтегруватися

в існуючу інфраструктуру підприємств. Реалізовані алгоритми машинного навчання продемонстрували високу ефективність у виявленні аномалій, а проведені експерименти підтвердили, що використання нейронних мереж і моделей LSTM дозволяє досягти найвищих показників точності та надійності.

Особлива увага приділена економічному ефекту впровадження системи. Результати розрахунків засвідчили значну економію ресурсів і зниження фінансових ризиків, пов’язаних із витоками даних. Запропоновані методи дозволяють оптимізувати витрати на IT-інфраструктуру, зменшити споживання енергоресурсів та підвищити продуктивність персоналу.

Практичне впровадження системи продемонструвало її переваги у порівнянні з традиційними методами захисту інформації. Система ефективно інтегрується з існуючими DLP, SIEM та IAM-рішеннями, що створює комплексний підхід до інформаційної безпеки. У роботі також наведено рекомендації для подальшого вдосконалення системи, включаючи адаптацію до нових викликів, інтеграцію з хмарними платформами та автоматизацію управління.

Загалом, розроблена система підтвердила свою ефективність як інноваційний інструмент захисту даних, здатний мінімізувати ризики витоків інформації та забезпечити високий рівень безпеки в умовах сучасного корпоративного середовища.

ВИСНОВКИ

Магістерська робота присвячена розробці та впровадженню системи аналізу поведінки користувачів (UEBA), яка базується на використанні сучасних методів штучного інтелекту для забезпечення високого рівня інформаційної безпеки в корпоративному середовищі. У роботі показано, що ефективне поєднання традиційних підходів до захисту даних із можливостями, які надають алгоритми машинного та глибинного навчання, дозволяє суттєво підвищити рівень захисту інформації.

Проведений аналіз існуючих систем захисту даних підтвердив необхідність використання новітніх технологій для виявлення складних та динамічних загроз. У результаті дослідження було розроблено архітектуру системи UEBA, що забезпечує багатоступеневий аналіз поведінки користувачів, включаючи збір даних, моделювання нормальної поведінки, виявлення аномалій та автоматичне реагування. Створена система інтегрується з іншими інструментами безпеки, такими як DLP, IAM та SIEM, що дозволяє створити комплексну екосистему захисту даних.

Особливу увагу було приділено тестуванню запропонованих моделей машинного навчання. Проведені експерименти підтвердили, що використання таких методів, як нейронні мережі та LSTM, забезпечує високу точність у виявленні потенційних загроз, що дозволяє запобігати інцидентам ще до їхнього настання. Крім того, тестування системи в умовах реального корпоративного середовища показало її економічну ефективність, зниження витрат на управління IT-інфраструктурою та оптимізацію використання ресурсів.

У роботі також наведено практичні рекомендації для подальшого вдосконалення системи. Зокрема, запропоновано впровадження прогнозної аналітики для передбачення загроз, інтеграцію з хмарними платформами для забезпечення масштабованості, а також адаптацію системи для роботи в нових галузях, таких як фінансовий сектор, медицина та промисловість.

Загальні результати роботи підтверджують, що розроблена система UEBA є ефективним інструментом для запобігання витокам інформації в умовах сучасного корпоративного середовища. Вона не лише забезпечує високий рівень безпеки, але й сприяє покращенню репутації підприємства, довірі клієнтів і партнерів, що є важливим аспектом в умовах цифровізації економіки. Перспективи подальшого розвитку системи відкривають можливість для її використання в умовах динамічного розвитку технологій та зростання кіберзагроз, що робить її актуальним рішенням для широкого кола завдань.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. **Aggarwal, C. C.** (2015). *Outlier Analysis*. Springer.
2. **Bishop, C. M.** (2006). *Pattern Recognition and Machine Learning*. Springer.
3. **Goodfellow, I., Bengio, Y., Courville, A.** (2016). *Deep Learning*. MIT Press.
4. **Han, J., Pei, J., Kamber, M.** (2011). *Data Mining: Concepts and Techniques*. Elsevier.
5. **Szeliski, R.** (2010). *Computer Vision: Algorithms and Applications*. Springer.
6. **Pereira, R., et al.** (2018). "Anomaly Detection in Logs Using Machine Learning". *Journal of Big Data*.
7. **Witten, I. H., Frank, E., Hall, M. A., Pal, C. J.** (2016). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.
8. **Manning, C. D., Raghavan, P., Schütze, H.** (2008). *Introduction to Information Retrieval*. Cambridge University Press.
9. **Scikit-learn Documentation**. (2023). Retrieved from https://scikit-learn.org.
10. **TensorFlow Documentation**. (2023). Retrieved from https://tensorflow.org.
11. **Nielsen, M.** (2015). *Neural Networks and Deep Learning*. Online Book.
12. **Liao, H., et al.** (2016). "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks". *IEEE Transactions on Emerging Topics in Computational Intelligence*.
13. **Kaspersky Lab Report**. (2023). "Trends in Cybersecurity: Data Leakage Prevention". Retrieved from https://kaspersky.com.
14. **IBM Security Report**. (2023). "Cost of a Data Breach". Retrieved from https://ibm.com.
15. **Mitchell, T. M.** (1997). *Machine Learning*. McGraw-Hill.
16. **ISO/IEC 27001:2022**. *Information Security Management Systems Requirements*. International Organization for Standardization.
17. **SANS Institute Whitepaper**. (2023). "Behavioral Analytics in Cybersecurity". Retrieved from https://sans.org
18. **Brown, L., et al.** (2021). "Role of Artificial Intelligence in Information Security". *Journal of Information Security*.
19. **Microsoft Azure Security Blog**. (2023). "UEBA Solutions: Enhancing Data Security". Retrieved from https://azure.microsoft.com.

20. **CISCO Annual Cybersecurity Report**. (2023). "Analytics in Modern Data Security". Retrieved from https://cisco.com.

Додаток А

```
            import pandas as pd
import matplotlib.pyplot as plt
from tkinter import *
from tkinter import ttk
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg

class AnomalyApp:
    def __init__(self, root):
        self.root = root
        self.root.title("Аналіз аномалій")
        self.root.geometry("800x700")

        # Завантаження даних
        try:
            self.df = pd.read_csv('formatted_detected_anomalies.csv')
            if 'Activity' not in self.df.columns:
                raise ValueError("Файл не містить необхідних колонок.")
        except Exception as e:
            Label(root, text=f"Помилка завантаження даних: {e}", fg="red").pack()
            return

        # Таблиця
        self.tree = ttk.Treeview(root, columns=list(self.df.columns), show='headings')
        for col in self.df.columns:
            self.tree.heading(col, text=col)
            self.tree.column(col, width=100, anchor="center")
        self.tree.pack(fill=BOTH, expand=True, padx=10, pady=10)

        for _, row in self.df.iterrows():
            self.tree.insert("", "end", values=list(row))

        # Кругова діаграма
        self.figure, self.ax = plt.subplots(figsize=(6, 6))
        self.canvas = FigureCanvasTkAgg(self.figure, master=root)
        self.canvas.get_tk_widget().pack(fill=BOTH, expand=True, padx=10, pady=10)

        self.plot_activity_distribution()

    def plot_activity_distribution(self):
        activity_counts = self.df['Activity'].value_counts()
        self.ax.clear()
        activity_counts.plot(
```

```
            kind='pie',
            ax=self.ax,
            autopct='%1.1f%%',
            startangle=90,
            colors=plt.cm.Paired.colors,
            labels=activity_counts.index
        )
        self.ax.set_ylabel("")
        self.ax.set_title("Розподіл активностей")
        self.canvas.draw()

if __name__ == "__main__":
    root = Tk()
    app = AnomalyApp(root)
    root.mainloop()

                import pandas as pd
import numpy as np
import random
from datetime import datetime, timedelta

# Налаштування
num_users = 50
num_logs = 1000

# Імітація користувачів
users = [f"user_{i+1}" for i in range(num_users)]
ip_addresses = [f"192.168.1.{i+1}" for i in range(num_users)]

# Генерація логів
logs = []
for _ in range(num_logs):
    user = random.choice(users)
    ip = random.choice(ip_addresses)
    time = datetime.now() - timedelta(minutes=random.randint(0, 1000))
    action = random.choice(["login", "file_access", "data_request"])
    logs.append([user, ip, time, action])

# Створення DataFrame
df = pd.DataFrame(logs, columns=["user_id", "ip_address", "timestamp", "action"])

# Збереження даних у CSV
df.to_csv("user_logs.csv", index=False)

print("Логи згенеровано та збережено у файл user_logs.csv")
```

```
        import pandas as pd
from sklearn.ensemble import IsolationForest
import matplotlib.pyplot as plt

# Завантаження та обробка даних
df = pd.read_csv("user_logs.csv")
df['timestamp'] = pd.to_datetime(df['timestamp'])
df['hour'] = df['timestamp'].dt.hour
df['day_of_week'] = df['timestamp'].dt.dayofweek

# Вибір ознак для моделі
features = df[['hour', 'day_of_week']]

# Створення та навчання моделі Isolation Forest
model = IsolationForest(contamination=0.05, random_state=42)
df['anomaly_score'] = model.fit_predict(features)

# Додавання стовпця для зручності інтерпретації
df['is_anomaly'] = df['anomaly_score'].apply(lambda x: 1 if x == -1 else 0)

# Перегляд кількості аномалій
print("Кількість виявлених аномалій:", df['is_anomaly'].sum())

# Візуалізація аномалій
plt.figure(figsize=(10, 6))
plt.scatter(df['hour'], df['day_of_week'], c=df['is_anomaly'], cmap='coolwarm',
edgecolors='k')
plt.xlabel('Година дня')
plt.ylabel('День тижня')
plt.title('Виявлені аномалії у поведінці користувачів')
plt.show()
```

```
        import pandas as pd

# Шлях до початкового файлу
input_file = "formatted_detected_anomalies.csv"

# Шлях до виправленого файлу
output_file = "formatted_detected_anomalies_fixed.csv"

try:
    # Спроба прочитати файл із правильним кодуванням
    df = pd.read_csv(input_file, encoding="utf-8")
    print("Файл успішно завантажено з кодуванням utf-8.")
```

```
except UnicodeDecodeError:
    print("Помилка кодування, спробую інше кодування.")
    try:
        # Спроба прочитати файл із іншим кодуванням
        df = pd.read_csv(input_file, encoding="windows-1252")
        print("Файл успішно завантажено з кодуванням windows-1252.")
    except Exception as e:
        print(f"Не вдалося прочитати файл: {e}")
        exit()

# Перевірка та виправлення колонок
if 'Anomaly' in df.columns:
    # Виправлення можливих помилок у колонці Anomaly
    df['Anomaly'] = df['Anomaly'].str.encode('utf-8', errors='ignore').str.decode('utf-8')
else:
    print("Колонка 'Anomaly' не знайдена у файлі.")

# Збереження виправленого файлу
df.to_csv(output_file, index=False, encoding="utf-8")
print(f"Файл успішно виправлено та збережено у: {output_file}")

            import pandas as pd
from sklearn.ensemble import IsolationForest
from sklearn.svm import OneClassSVM
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt
import joblib
import os
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '2'
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers

# Завантаження даних
df = pd.read_csv("user_logs.csv")
df['timestamp'] = pd.to_datetime(df['timestamp'])
df['hour'] = df['timestamp'].dt.hour
df['day_of_week'] = df['timestamp'].dt.dayofweek

# Додавання ознак
df['action_code'] = df['action'].map({'login': 0, 'file_access': 1, 'data_request': 2})
df['working_hours'] = df['hour'].apply(lambda x: 1 if 9 <= x <= 18 else 0)
df['is_internal_ip'] = df['ip_address'].apply(lambda x: 1 if x.startswith("192.168") else
0)
df = df.sort_values(by=['user_id', 'timestamp'])
```

```
# Додавання нової ознаки - кількість дій користувача за останню годину
df['user_action_count_last_hour'] = df.groupby('user_id')['timestamp'].transform(
    lambda x: x.diff().dt.total_seconds().fillna(0).apply(lambda sec: 1 if sec <= 3600 else 0).cumsum()
)

# Вибір ознак для моделі
features = df[['hour', 'day_of_week', 'action_code', 'working_hours', 'is_internal_ip', 'user_action_count_last_hour']]

# Стандартизація даних
scaler = StandardScaler()
features_scaled = scaler.fit_transform(features)

# Isolation Forest
model_iforest = IsolationForest(contamination=0.03, random_state=42)
df['anomaly_iforest'] = model_iforest.fit_predict(features_scaled)

# One-Class SVM
model_ocsvm = OneClassSVM(kernel='rbf', nu=0.03)
df['anomaly_ocsvm'] = model_ocsvm.fit_predict(features_scaled)

# Autoencoder
input_dim = features_scaled.shape[1]
autoencoder = keras.Sequential([
    layers.Input(shape=(input_dim,)),
    layers.Dense(16, activation='relu'),
    layers.Dense(8, activation='relu'),
    layers.Dense(16, activation='relu'),
    layers.Dense(input_dim, activation='linear')
])
autoencoder.compile(optimizer='adam', loss='mse')

# Навчання Autoencoder
autoencoder.fit(features_scaled, features_scaled, epochs=50, batch_size=32, verbose=0)

# Обчислення помилки відновлення
reconstructed = autoencoder.predict(features_scaled)
mse = ((features_scaled - reconstructed) ** 2).mean(axis=1)
threshold = mse.mean() + mse.std()
df['anomaly_autoencoder'] = (mse > threshold).astype(int)

# Виведення результатів
```

```
df['final_anomaly'] = (df['anomaly_iforest'] == -1) | (df['anomaly_ocsvm'] == -1) |
(df['anomaly_autoencoder'] == 1)

print("Кількість виявлених аномалій:", df['final_anomaly'].sum())
anomalies = df[df['final_anomaly'] == 1]
anomalies.to_csv("detected_anomalies_hybrid.csv", index=False)

# Візуалізація
plt.figure(figsize=(10, 6))
plt.scatter(df['hour'], df['day_of_week'], c=df['final_anomaly'], cmap='coolwarm',
edgecolors='k')
plt.xlabel('Година дня')
plt.ylabel('День тижня')
plt.title('Виявлені аномалії (Гібридний підхід)')
plt.show()
```