Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Факультет комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)
Кафедра кібербезпеки
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр
(освітній рівень)

на тему: "Методи виявлення аномалій у великих даних для прогнозування кіберзагроз"

Виконав: студент (ка)
Спеціальності:
125 «Кібербезпека»
(шифр і назва напряму підготовки, спеціальності)
Гладчук Максим Віталійович
підпис (прізвище та ініціали)

Керівник Стадник М. А.
підпис (прізвище та ініціали)

Нормоконтроль Тимощук Д. І.
підпис (прізвище та ініціали)

Завідувач кафедри Загородна Н.В.
підпис (прізвище та ініціали)

Рецензент
підпис (прізвище та ініціали)

м. Тернопіль – 2024

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра кібербезпеки
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Загородна Н.В.
(підпис) (прізвище та ініціали)
«__» _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Магістр
(назва освітнього ступеня)

за спеціальністю 125 Кібербезпека та захист інформації
(шифр і назва спеціальності)

Студенту Гладчуку Максиму Віталійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи
Методи виявлення аномалій у великих даних для прогнозування кіберзагроз

Керівник роботи Стадник Марія Андріївна к. т. н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

2. Термін подання студентом завершеної роботи 20.12.2024

3. Вихідні дані до роботи Наукові публікації щодо методиів виявлення аномалій у великих даних для прогнозування кіберзагроз

4. Зміст роботи (перелік питань, які потрібно розробити)
Проаналізувати властивості великих даних які ускладнюють виявлення аномалій.
Проаналізувати які методи найкраще виявляють аномалії у великих даних.
Проаналізувати як практично впровадити, оцінити та оптимізувати методи виявлення аномалій у реальних умовах корпоративних мереж?
Охорона праці та безпека в надзвичайних ситуаціях
Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1 Титульний слайд. 2 Вступ та актуальність. 3 Мета та завдання. 4 Методи штучного інтелекту для виявлення аномалій у великих даних. 5 Архітектура для збору даних. 6 Огляд датасету. 7 Застосування автоенкодингу. 8 Попередня обробка даних. 9 Блок схеми алгоритму k-means. 10 Блок схема алгоритму SVM. 11 Блок схема алгоритму isolative forest. 12 Результати оцінки якостей моделей. 13 Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Осухівська Г.М., к.т.н., доцент		
Безпека в надзвичайних ситуаціях	Клепчик В.М., проректор з адміністративно-господарської роботи та будівництва		

7. Дата видачі завдання 23.09.2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	23.09 – 25.09	Виконано
2.	Підбір джерел для аналізу великих даних	25.09 – 05.10	
3.	Опрацювання джерел в галузі дослідження	10.10 – 20.10	
4.	Оформлення розділу «Теоретичні основи виявлення аномалій у великих даних»	01.11 – 05.11	
5.	Оформлення розділу «Методи виявлення аномалій у великих даних для прогнозування кіберзагроз»	06.11 – 11.11	
6.	Оформлення розділу «Практичні аспекти застосування методів виявлення аномалій у великих даних»	12.11-20.11	
7.	Виконання завдання до підрозділу «Охорона праці та Безпека в надзвичайних ситуаціях»	21.11 – 25.11	
8.	Оформлення кваліфікаційної роботи	23.11 – 05.12	
9.	Нормоконтроль	06.12 – 11.12	
10.	Перевірка на плагіат	11.12 – 12.12	
11.	Попередній захист кваліфікаційної роботи	17.12 – 18.12	
12.	Захист кваліфікаційної роботи	23.12.2024	

Студент _____ Гладчук М.В
(підпис) (прізвище та ініціали)

Керівник роботи _____ Стадник М.А.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Методи виявлення аномалій у великих даних для прогнозування кіберзагроз // ОР «Магістр» // Гладчук Максим Віталійович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра кібербезпеки, група СБм-61 // Тернопіль, 2024 // С. 104, рис. – 15, табл. – 4 , додат. – 2.

Ключові слова: виявлення аномалій, кібербезпека, великі дані, машинне навчання, глибинне навчання.

У кваліфікаційній роботі вирішується завдання ефективного виявлення аномалій у великих масивах даних, що є критично важливим для підвищення рівня кібербезпеки сучасних інформаційних систем. У роботі розглянуто основні особливості великих даних, наведено переваги та недоліки традиційних і сучасних методів виявлення аномалій, включаючи кластеризаційні підходи та глибинні нейронні мережі.

Детально проаналізовано механізми оцінки зрілості та ефективності систем виявлення, а також методи оптимізації моделей для підвищення точності й надійності прогнозування загроз. Представлено систему виявлення аномалій на основі машинного та глибинного навчання, а також проведено порівняльний аналіз запропонованого підходу за критеріями точності, швидкодії та застосовності в умовах реальних корпоративних мереж.

ABSTRACT

Big data anomaly detection methods for cyber threat prediction // Thesis of educational level "Master"// Maksym Hladchuk // Ternopil Ivan Puluj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Cybersecurity, group СБм-61 // Ternopil, 2024 // P. 104, tables 4, drawings 15, added. – 2.

Keywords: anomaly detection, cybersecurity, big data, machine learning, deep learning.

The qualification work addresses the task of effective anomaly detection in large-scale data sets, which is critical for enhancing the cybersecurity of modern information systems. The study examines the key features of big data and presents the advantages and disadvantages of both traditional and advanced anomaly detection methods, including clustering approaches and deep neural networks.

A detailed analysis of methods for assessing the maturity and effectiveness of detection systems is provided, as well as approaches for model optimization to improve the accuracy and reliability of threat prediction. The paper introduces an anomaly detection system based on machine learning and deep learning techniques and includes a comparative analysis of the proposed approach in terms of accuracy, performance, and applicability within real corporate network environments.

ЗМІСТ

ВСТУП

Актуальність теми. У сучасних умовах стрімкого розширення цифрового простору та зростання обсягів даних, що надходять із різноманітних джерел – веб-серверів, соціальних мереж, мобільних пристроїв, систем «Інтернету речей», – проблема ефективного виявлення аномалій стає критично важливою. Традиційні методи, що спираються на статичні сигнатури або фіксовані правила, втрачають дієвість проти дедалі витонченіших кібератак. Це обумовлює необхідність застосування гнучких, інтелектуальних підходів, здатних автоматично ідентифікувати нетипові патерни поведінки та прогнозувати потенційні загрози. Особливої актуальності набуває інтеграція кластеризаційних методів та технологій глибинного навчання, що дозволяє підвищити точність і надійність систем кібербезпеки.

Мета і задачі дослідження. Метою кваліфікаційної роботи є розробка комплексного підходу до виявлення аномалій у великих масивах даних, що поєднує методи кластеризації з глибинними нейронними мережами та автоенкодерами. Досягнення поставленої мети передбачає визначення найбільш перспективних алгоритмів машинного навчання для роботи з великими обсягами інформації в умовах динамічно змінюваного середовища, розроблення стратегій нормалізації та інженерії ознак для підвищення якості вхідних даних, реалізацію гібридних підходів, що інтегрують традиційні методи кластеризації (k-середніх) із глибинними автоенкодерами та нейронними мережами, упровадження механізмів оцінки й валідації моделей задля мінімізації хибних спрацювань, а також формулювання рекомендацій щодо практичного використання одержаних результатів у реальних системах кібербезпеки.

Об’єкт дослідження. Об’єктом дослідження є складні та динамічні інформаційні середовища, що обробляють великі обсяги даних, стаючи потенційним середовищем для появи нетипових патернів, пов’язаних із кібератаками.

Предмет дослідження. Предметом дослідження є методи та алгоритми поєднання кластеризаційних підходів із глибинними нейронними мережами,

спрямовані на виявлення аномалій у даних та підвищення ефективності систем кібербезпеки.

Наукова новизна одержаних результатів кваліфікаційної роботи. Наукова новизна полягає у формуванні інтегрованого підходу до виявлення аномалій, який поєднує традиційні методи кластеризації з глибинними нейронними мережами й автоенкодерами. Запропоноване рішення забезпечує гнучкість, точність і надійність виявлення нетипових патернів, що дає змогу оперативно адаптуватися до нових та невідомих загроз.

Практичне значення одержаних результатів. Одержані результати мають вагоме практичне значення для фахівців з кібербезпеки та адміністраторів інформаційних систем. Запропонований комплексний підхід дозволить своєчасно ідентифікувати приховані патерни шкідливої активності, знижуючи ризики витоку даних, мінімізуючи наслідки потенційних кібератак та підвищуючи загальну стійкість цифрової інфраструктури. Застосування розроблених методів у реальних умовах сприятиме побудові більш адаптивних та надійних систем захисту.

Апробація результатів магістерської роботи. Основні результати проведених досліджень обговорювались на: XII науково-технічна конференція «Інформаційні моделі, системи та технології» (м.Тернопіль, Україна).

Публікації. Основні результати кваліфікаційної роботи опубліковано у працях конференції (див. Додаток А).

РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ВИЯВЛЕННЯ АНОМАЛІЙ У ВЕЛИКИХ ДАНИХ

1.1 Основні концепції великих даних

Великі дані являють собою масиви інформації, що характеризуються високими показниками обсягу, швидкості обробки, різноманіттям і рівнем достовірності, які відомі як чотири V (Volume, Velocity, Variety, Veracity). Обсяг (Volume) означає, що дані генеруються у величезних кількостях із різноманітних джерел, таких як соціальні мережі, сенсори, транзакції тощо. Важливо враховувати, що обсяг даних не лише створює виклики для їхнього зберігання, але й вимагає ефективних методів обробки та аналізу для виявлення цінної інформації. Швидкість (Velocity) вказує на те, що дані генеруються та обробляються в режимі реального часу або з мінімальними затримками, що критично для виявлення аномалій, адже затримка в аналізі може призвести до пропуску загроз[1]. Різноманіття (Variety) охоплює широкий спектр типів даних, таких як структуровані, напівструктуровані та неструктуровані дані, що вимагає використання різних підходів для їхньої обробки та аналізу. Достовірність (Veracity) стосується надійності та якості даних, оскільки не всі зібрані дані є точними, і саме цей аспект є важливим для мінімізації помилкових спрацьовувань у процесі виявлення загроз.

Великі дані відіграють ключову роль у контексті кібербезпеки, оскільки їхні властивості безпосередньо впливають на здатність виявляти аномалії та прогнозувати кіберзагрози. Кіберзагрози генерують велику кількість сигналів та подій, які можуть вказувати на підготовку або виконання атаки. Висока швидкість обробки дозволяє системам швидко реагувати на потенційні загрози, одночасно аналізуючи різноманітні дані з різних джерел, таких як журнали подій, трафік мережі чи поведінкові патерни користувачів. Наприклад, зростання кількості нетипових запитів до серверів або відхилення від звичайних патернів у трафіку можуть сигналізувати про потенційні загрози. Однак для ефективного виявлення

таких аномалій необхідно ретельно враховувати фактор достовірності даних, щоб уникнути помилкових спрацювань та пропуску реальних загроз[2].

Одним із найпоширеніших прикладів використання великих даних у сфері кібербезпеки є моніторинг мережі в реальному часі для виявлення атак типу "відмова в обслуговуванні" (DDoS). Такі атаки генерують величезну кількість трафіку, і аналіз цього трафіку за допомогою технологій великих даних дозволяє виявити аномалії у поведінці користувачів або систем. Наприклад, компанії, що використовують аналітичні платформи, такі як Splunk або ELK Stack, можуть в реальному часі моніторити трафік і фіксувати відхилення від нормальної поведінки, що сигналізує про атаку[3]. Інший приклад – аналіз поведінкових патернів користувачів для виявлення нетипової активності, яка може бути індикатором компрометації облікових записів.

Технології, що використовуються для обробки великих даних, такі як Hadoop та Apache Spark, є критично важливими для забезпечення ефективної обробки й аналізу великих масивів даних. Hadoop дозволяє зберігати та обробляти великі обсяги неструктурованих даних, розподіляючи їх між кількома вузлами для підвищення ефективності. Spark, своєю чергою, забезпечує швидкий аналіз даних завдяки можливості обробляти їх у пам'яті, що робить його надзвичайно ефективним для завдань, що потребують обробки в реальному часі, таких як виявлення аномалій у мережевому трафіку. Без використання таких технологій аналіз великих даних стає практично неможливим, оскільки традиційні методи обробки не здатні впоратися з такою кількістю інформації в обмежений час.

Однак, незважаючи на важливість великих даних у контексті кібербезпеки, існує низка викликів, що ускладнюють їхнє використання. Одним із найбільших викликів є зберігання даних, оскільки їхній обсяг постійно зростає, що вимагає значних обчислювальних ресурсів. Ще однією проблемою є обробка великої кількості "шуму" — непотрібних або недостовірних даних, що можуть заплутувати системи виявлення загроз.

Це особливо важливо в контексті виявлення аномалій, оскільки неправильно відфільтровані дані можуть призвести до пропуску суттєвих загроз або, навпаки, до великої кількості хибнопозитивних результатів.

1.2 Аномалії у великих даних

Аномалії у великих даних — це критичні відхилення в інформаційних наборах, які значно відрізняються від очікуваних або нормальних патернів. В контексті великих даних, виявлення аномалій є складним завданням через великий обсяг інформації, різноманітність джерел та швидкість обробки. Оскільки дані постійно збільшуються, навіть незначні аномалії можуть бути індикаторами серйозних кіберзагроз, таких як зломи, витоки інформації або атаки зловмисників. Наприклад, під час аналізу трафіку в корпоративній мережі можуть виникнути аномальні зміни в активності, що вказують на DDoS-атаку або спробу компрометації системи[4]. Виявлення таких відхилень є критичним для запобігання потенційним загрозам.

Аномалії у великих даних класифікуються на кілька категорій залежно від їх природи та прояву:

1. Статистичні аномалії — це відхилення від математичних середніх значень. Наприклад, у фінансовій системі, різке збільшення кількості транзакцій за короткий період може сигналізувати про шахрайську діяльність.

2. Темпоральні аномалії — виникають, коли події відбуваються не в відповідності до звичайних часових патернів. Наприклад, у 2019 році компанія X виявила аномальну активність на серверах у нічний час, що було індикатором несанкціонованого доступу[5].

3. Аномалії поведінкових патернів — відхилення від нормальної поведінки користувачів або систем. Наприклад, коли користувач, який зазвичай входить у систему з одного регіону, раптово здійснює вхід з іншого континенту, це може свідчити про спробу викрадення облікових даних.

Кожен тип аномалії вимагає окремого підходу до виявлення та аналізу. Хоча темпоральні та поведінкові аномалії можуть перетинатися, різниця полягає у природі їхньої появи: темпоральні відображають часові зміни, тоді як поведінкові фокусуються на дії користувачів або систем.

Методи виявлення аномалій у великих даних поділяються на кілька категорій:

1. Алгоритми машинного навчання:

- Методи без нагляду: Один з найбільш ефективних методів — це ізоляційні ліси, які будують дерева для ізоляції кожної точки даних. Перевага цього методу полягає у його здатності працювати з великими обсягами інформації, швидко виявляючи аномалії.
- Кластеризація (K-means): Цей метод групує подібні дані в кластери, а будь-які точки, що знаходяться далеко від своїх кластерів, вважаються аномальними.

2. Статистичні методи:

- Регресія: Використовується для прогнозування нормальної поведінки на основі попередніх даних. Відхилення від регресійної моделі свідчать про наявність аномалій.
- Аналіз часових рядів: Використовується для виявлення темпоральних аномалій шляхом порівняння поточних значень з історичними даними.

3. Методи штучного інтелекту:

- Глибоке навчання: Рекурентні нейронні мережі (RNN) виявляють складні патерни в послідовностях даних, що дозволяє виявляти аномалії, які традиційні методи пропускають. Наприклад, використання RNN дозволило одній з фінансових компаній знизити кількість хибних спрацьовувань на 30% при аналізі транзакцій.

Кожен метод має свої переваги і недоліки. Наприклад, ізоляційні ліси ефективні для швидкого аналізу великих масивів даних, але можуть пропускати складні аномалії, тоді як глибоке навчання забезпечує високу точність, проте потребує значних обчислювальних ресурсів[6].

Процес виявлення аномалій у великих даних стикається з низкою викликів:

1. Об'єм даних: Великі масиви даних потребують потужних обчислювальних ресурсів та високої масштабованості для їх аналізу. Технології, такі як Apache Hadoop і Spark, дозволяють обробляти великі обсяги даних у розподілених середовищах.

2. Швидкість обробки: Реальний час обробки критично важливий для кібербезпеки, коли затримки можуть призвести до катастрофічних наслідків.

Швидкісні системи обробки даних, такі як Storm або Flink, дозволяють миттєво реагувати на загрози.

3. Висока вимірність: Дані можуть мати тисячі змінних, що ускладнює виявлення аномалій. Методи зниження розмірності, такі як PCA (аналіз головних компонент), використовуються для зменшення кількості змінних без втрати ключової інформації.

4. Шум у даних: Наявність великої кількості неінформативних або некоректних даних ускладнює аналіз і може призвести до хибних спрацьовувань. Використання алгоритмів фільтрації допомагає мінімізувати вплив шуму.

Виявлення аномалій є критичним елементом кібербезпеки. Наприклад, виявлення незвичних патернів у мережевому трафіку дозволяє компаніям запобігти DDoS-атакам, які можуть вивести з ладу інфраструктуру[7]. У 2020 році компанія Y змогла виявити витік даних завдяки ідентифікації аномального зростання обсягу переданих файлів на зовнішні IP-адреси[8]. Завдяки цьому вдалося запобігти втраті конфіденційної інформації.

Сучасні дослідження у сфері виявлення аномалій спрямовані на покращення точності методів і скорочення часу реагування на загрози. Наприклад, методи глибокого навчання, зокрема рекурентні нейронні мережі та автокодери, покращують здатність аналізувати складні патерни у великих даних. Хмарні обчислення, такі як AWS або Google Cloud, також відіграють важливу роль, забезпечуючи масштабовані платформи для аналізу в режимі реального часу.

Використання комбінованих підходів, таких як гібридні моделі машинного навчання та штучного інтелекту, відкриває нові можливості для підвищення ефективності виявлення аномалій у різних галузях, включно з фінансами, охороною здоров’я та промисловістю[9].

Виявлення аномалій у великих даних є невід'ємним компонентом кібербезпеки, оскільки воно дозволяє ідентифікувати та запобігати потенційним загрозам. Основні виклики, такі як масштабованість і висока вимірність даних, вимагають постійного вдосконалення методів виявлення та використання нових технологій, зокрема глибокого навчання та хмарних обчислень. Перспективи цієї

галузі пов'язані з розвитком автоматизованих систем прогнозування загроз, які дозволять суттєво зменшити час реагування та підвищити точність аналізу.

1.3 Кіберзагрози

Кіберзагрози в контексті великих даних являють собою сукупність загроз безпеці даних, що можуть виникати в процесі їх збору, зберігання та обробки[10]. Великі дані стають об’єктом атак через свою масштабність та значення для бізнесу, урядових установ та критичних інфраструктур. Основними цілями кіберзлочинців є конфіденційна інформація, аналітичні дані та інтелектуальна власність, яка міститься в системах великих даних. Завдяки їхній структурованості та наявності сенситивної інформації, кіберзлочинці можуть використовувати ці дані для фінансових злочинів, шантажу або викрадання ідентифікаційних даних[11]. Також великі дані є стратегічною метою через високу залежність бізнесу та державних установ від їх аналітики, що забезпечує прогнозування ринкових трендів, поведінки споживачів та інших важливих показників.

Кіберзагрози для великих даних можна класифікувати на кілька основних типів:

1. Проблеми з приватністю даних: надійність систем захисту великих масивів інформації часто недостатньою, що відкриває шлях до масштабних витоків відомостей, серед яких персональні дані, фінансова звітність чи інтелектуальна власність. Прикладом стала атака на Equifax у 2017 році, коли зловмисникам вдалося одержати доступ до персональної інформації понад 147 мільйонів осіб, включно з номерами соціального страхування та кредитними історіями[12].

2. Загрози для достовірності даних: навмисне викривлення або внесення шкідливих змін до великих обсягів даних може призвести до спотворених аналітичних висновків. Це, у свою чергу, негативно впливає на ухвалення управлінських рішень, оскільки керівництво базуватиметься на неточній або навмисно підробленій інформації, що є критичною проблемою для бізнесу та державних інституцій.

Наприклад, атака на електронні системи виборів або системи прогнозування попиту може суттєво вплинути на результати прогнозів або прийняття рішень.

3. Атаки на доступність даних: Втрата доступу до даних через атаки на інфраструктуру великих даних, зокрема DDoS-атаки або програми-викупники, може паралізувати бізнес-процеси[13]. У 2020 році одна з найбільших атак програмою-викупником відбулася проти корпорації Garmin, яка втратила доступ до своїх систем на кілька днів, що призвело до значних фінансових втрат.

4. Методи соціальної інженерії: Кіберзлочинці використовують людський фактор для доступу до конфіденційних даних. Соціальна інженерія може включати фішингові атаки, у яких шахраї отримують доступ до великих баз даних через обман працівників. Наприклад, фішингова атака на Uber у 2016 році дала доступ до особистих даних 57 мільйонів користувачів і водіїв.

Загрози для конфіденційності, цілісності та доступності даних:

1. Конфіденційність: Загрози для конфіденційності великих даних включають витоки інформації через недоліки в системах захисту, хмарні вразливості, або зламані канали зв'язку між хмарними платформами. Уразливі API можуть дозволити несанкціонованим користувачам отримати доступ до сенситивних даних, наприклад, фінансових транзакцій чи даних про клієнтів.

2. Цілісність: Порушення цілісності великих даних відбувається через атаки типу SQL-ін'єкцій, коли дані баз даних змінюються або видаляються зловмисниками. Це може вплинути на аналітичні висновки або порушити ланцюги постачання, як це сталося під час атак на ритейлерів у 2013 році, коли були зламані дані про транзакції мільйонів покупців[14].

3. Доступність: Атаки типу DDoS або ransomware можуть призвести до того, що організації втрачають доступ до своїх даних на значний час, що паралізує роботу критичних систем[15]. Наприклад, атака програми-викупника на компанію Colonial Pipeline у 2021 році спричинила зупинку роботи паливопроводів і вплинула на енергопостачання великого регіону.

Методи атак:

1. SQL-ін'єкції: Це один з найпоширеніших методів атаки, який використовується для впливу на великі бази даних. Атакуючі вводять шкідливий

SQL-код у запити до бази даних, отримуючи доступ до сенситивних даних[16]. Це часто відбувається через недостатній захист користувацьких форм на вебсайтах.

2. DDoS-атаки: Мета цих атак — перевантажити інфраструктуру великих даних, що призводить до недоступності сервісів. Атаки можуть бути спрямовані на сервіси, які використовуються для збору чи обробки великих даних, що робить систему недоступною для користувачів[17].

3. Атаки через API: Вразливості в API можуть використовуватися для отримання несанкціонованого доступу до великих обсягів даних. Наприклад, уразливості API призвели до витоку даних з LinkedIn у 2021 році, що дозволило шахраям отримати доступ до профілів мільйонів користувачів.

Великі дані є особливо вразливими через свою цінність для бізнесу та економіки. Втрата даних може спричинити значні фінансові збитки, втрату довіри клієнтів та юридичні наслідки. Наприклад, у випадку Equifax, компанія зіткнулася з багатомільйонними штрафами та позовами після витоку даних, що значно вплинуло на її репутацію і фінансову стабільність[18]. Інший приклад — Uber, яка після кібератаки втратила довіру користувачів і змушена була виплатити штрафи на сотні мільйонів доларів.

Кіберзагрози постійно еволюціонують разом з розвитком технологій великих даних. Зокрема, кіберзлочинці починають використовувати штучний інтелект та машинне навчання для автоматизації атак та обману захисних систем. Нові вектори атак включають використання блокчейн-технологій для прихованого обміну даними між зловмисниками та обхід традиційних систем захисту[19]. Шкідливі програми стають все більш складними, а їхні автори використовують хмарні сервіси для приховування своєї активності.

У контексті великих даних особливо важливим є дотримання міжнародних стандартів та нормативних актів. Стандарти, такі як ISO/IEC 27001, визначають вимоги до управління інформаційною безпекою, а закони на зразок GDPR і CCPA зобов'язують організації відповідально підходити до захисту персональних даних[20]. Компанії, що порушують ці норми, можуть бути піддані значним штрафам, як це сталося з Facebook, яка отримала штраф у розмірі 5 мільярдів доларів через порушення конфіденційності даних.

Очікується, що у майбутньому кіберзагрози, пов'язані з великими даними, будуть лише посилюватися. Зокрема, через зростаючий обсяг даних та складність їх обробки виникатимуть нові вектори атак, що будуть націлені на порушення безпеки великих даних. Очікується, що зловмисники активно використовуватимуть штучний інтелект для автоматизованих атак на аналітичні системи та інфраструктуру хмар.

1.3 Методи виявлення аномалій

Концепція великих даних охоплює сукупність методів і технологій для обробки, зберігання та аналізу масивних і різнорідних інформаційних потоків, що перевищують можливості традиційних систем обробки даних. Характеризуючи великі дані, зазвичай використовують модель 3V: обсяг (volume), швидкість (velocity) і різноманітність (variety)[21]. Однак, на сьогоднішній день до цієї моделі додають ще кілька ключових характеристик, таких як достовірність (veracity) та цінність (value), що дозволяє більш глибоко охарактеризувати виклики, пов'язані з обробкою даних.

Реальні кейси застосування методів великих даних демонструють їхню високу практичну цінність. Наприклад, компанія Netflix використовує методи кластеризації для аналізу споживчих вподобань, що дозволяє їй розробляти персоналізовані рекомендації для кожного користувача[22]. Інший приклад — Amazon, який застосовує алгоритми машинного навчання для виявлення аномалій у фінансових операціях і попередження шахрайства. Ці приклади підкреслюють важливість та універсальність методів великих даних у сучасних підприємствах.

Методи виявлення аномалій у великих даних поділяються на кілька категорій: статистичні методи, методи кластеризації, машинне навчання та методи на основі глибокого навчання[23].

Статистичні методи передбачають використання математичних моделей для виявлення відхилень у поведінці даних. Наприклад, методи на основі гіпотез статистичної значущості дозволяють виявляти значні відхилення від середнього значення в наборах даних[24]. Такий підхід застосовувався у великих банках для

моніторингу фінансових операцій та виявлення аномальних транзакцій. Однак основним недоліком статистичних методів є їхня обмежена здатність працювати з високорозмірними та різнорідними даними.

Методи кластеризації дозволяють групувати дані на основі схожості між ними. Алгоритм k-середніх, широко використовуваний у сфері маркетингу, дозволяє сегментувати клієнтів на основі їхніх поведінкових характеристик. Однак цей алгоритм має обмеження, наприклад, він погано працює з кластерами складних форм[25]. У таких випадках краще використовувати метод DBSCAN, який ефективно визначає кластери різної щільності. Практичний приклад — компанія Uber, яка використовує DBSCAN для аналізу транспортних потоків і оптимізації маршрутів.

Методи машинного навчання, такі як Isolation Forest, дозволяють автоматично виявляти аномалії на основі структурних особливостей даних. Наприклад, у фінансовій сфері алгоритм Isolation Forest використовується для ідентифікації підозрілих операцій у реальному часі. Цей метод особливо ефективний для роботи з великими даними, оскільки його обчислювальні витрати залишаються лінійними щодо обсягу даних, що відрізняє його від інших методів, таких як LOF, який вимагає більше обчислювальних ресурсів і погано масштабується на великих наборах даних[26].

Глибоке навчання — це сучасний підхід до виявлення аномалій, що використовує нейронні мережі для аналізу складних структур даних. Однак слід зазначити, що такі моделі є ресурсомісткими, оскільки для їхнього тренування потрібні обчислювальні потужності рівня десятків чи навіть сотень GFlops. Наприклад, Google використовує глибокі нейронні мережі для виявлення аномалій у своїх дата-центрах, що дозволяє знижувати енергоспоживання та запобігати поломкам обладнання[27]. Попри ефективність, глибоке навчання має й слабкі сторони, такі як схильність до вразливостей, пов'язаних із атаками на нейронні мережі (наприклад, атаками типу adversarial), що вимагає додаткових заходів безпеки.

Методи виявлення аномалій є важливими інструментами для виявлення незвичайних або підозрілих патернів у великих обсягах даних. Кожен з цих методів

має свої особливості, переваги та недоліки, які варто розглянути для вибору найбільш підходящого підходу в конкретних умовах. Таблиця 1.1 містить огляд основних методів виявлення аномалій, а також їх характеристик.

Таблиця 1.1. Методи виявлення аномалій.

Метод	Опис	Переваги	Недоліки	Застосування
Статистичні методи	Виявлення аномалій на основі статистичних моделей, таких як Z-оцінка чи границі	Простота реалізації; зрозумілість	Чутливість до змін у даних	Фінансові транзакції
Кластеризація	Використання алгоритмів кластеризації для виявлення об'єктів, що не належать до кластерів	Гнучкість; можливість роботи з неструктурованими даними	Потребує налаштування параметрів; може бути повільною	Виявлення шахрайства в електронній комерції
Машинне навчання	Використання алгоритмів, що навчаються на основі історичних даних для виявлення аномалій	Можливість автоматичного навчання; висока точність	Вимагає великих обчислювальних ресурсів	Обробка великих даних
Глибоке навчання	Використання нейронних мереж для виявлення складних патернів у даних	Висока точність; здатність виявляти складні аномалії	Складність налаштування; ризик переобучення	Відеоспостереження

Продовження таблиці 1.1

Логічні правила	Використання правил для виявлення аномалій на основі експертних знань	Зрозумілість; простота реалізації	Залежність від експертного знання	Системи моніторингу

Ця таблиця надає огляд основних методів виявлення аномалій, акцентуючи увагу на їхніх характеристиках, перевагах і недоліках. Наприклад, статистичні методи є простими у реалізації, але можуть не витримувати коливань у даних, тоді як машинне навчання, хоча й дуже потужне, вимагає значних обчислювальних ресурсів. Знання цих особливостей дозволяє вибрати найкращий підхід залежно від специфіки задачі.

Оцінка ефективності методів виявлення аномалій здійснюється на основі метрик, таких як точність, повнота, F1-міра та AUC-ROC[28]. Наприклад, у фінансових системах точність методу особливо важлива, оскільки хибні позитиви можуть призводити до зайвих витрат на додаткові перевірки. У випадку зі складними системами охорони здоров'я повнота методу є критично важливою, оскільки пропуск навіть однієї аномалії може призвести до катастрофічних наслідків[29]. Таким чином, вибір метрик залежить від конкретних потреб галузі.

Один із ключових викликів у виявленні аномалій у великих даних — це "прокляття розмірності". Високорозмірні дані можуть погіршувати ефективність методів машинного навчання. Наприклад, в компанії Facebook проблема надлишкової кількості параметрів у моделі для виявлення аномалій була вирішена шляхом впровадження алгоритмів зниження розмірності, таких як PCA (аналіз головних компонент) [30]. Це дозволило суттєво покращити точність виявлення та скоротити час обробки даних.

Виявлення аномалій стикається з низкою викликів і проблем, які можуть вплинути на ефективність методів. Розуміння цих викликів є важливим для розробки адаптивних рішень, здатних адекватно реагувати на зміни в даних. Таблиця нижче містить основні виклики та проблеми, з якими стикаються організації під час виявлення аномалій.

Таблиця 1.2. Виклики та проблеми.

Виклик/Проблема	Опис	Вплив на ефективність методу	Потенційні рішення
Прокляття розмірності	Зростання кількості змінних у даних ускладнює виявлення аномалій	Зниження точності	Використання методів редукції розмірності
Збалансованість класів	Наявність переважно нормальних даних у поєднанні з рідкісними аномаліями	Невірні висновки	Застосування методів, що працюють з незбалансованими даними
Динамічні зміни в даних	Зміни в патернах даних з часом, які вимагають адаптації алгоритмів	Зниження продуктивності	Регулярне оновлення моделей
Вимірювання ефективності	Відсутність чітких метрик для оцінки ефективності методів виявлення аномалій	Ускладнене порівняння	Розробка конкретних метрик для оцінки
Непередбачувані аномалії	Виявлення нових типів аномалій, які не були враховані під час навчання	Зниження ефективності	Використання адаптивних алгоритмів

Таблиця містить ключові виклики та проблеми, які можуть виникати під час виявлення аномалій. Наприклад, прокляття розмірності ускладнює ідентифікацію патернів, а динамічні зміни в даних можуть вимагати постійного вдосконалення моделей. Знання цих проблем дозволяє фахівцям розробляти стратегії, що зменшують їхній негативний вплив на процес виявлення аномалій.

У майбутньому варто очікувати зростання популярності федерованого навчання, яке дозволяє тренувати моделі без необхідності передавати дані між

серверами, що підвищує безпеку й конфіденційність[31]. Також активно досліджується можливість використання квантових обчислень для поліпшення ефективності виявлення аномалій. Наприклад, IBM наразі тестує квантові алгоритми для аналізу великих даних, що може значно знизити час обчислень і підвищити точність прогнозування кіберзагроз[32].

Сфера виявлення аномалій постійно розвивається, впроваджуючи нові технології та підходи, які обіцяють покращити результати. Розуміння майбутніх напрямків розвитку є важливим для фахівців, які прагнуть залишатися на передовій технологій. Таблиця 1.3 надає огляд актуальних напрямків розвитку в цій галузі.

Таблиця 1.3. Майбутні напрямки розвитку.

Напрямок розвитку	Опис	Потенційний вплив на виявлення аномалій	Приклади використання
Федероване навчання	Розподілене навчання моделей без централізації даних	Підвищення конфіденційності даних	Банківські системи
Квантові обчислення	Використання квантових алгоритмів для підвищення швидкості та ефективності виявлення	Зменшення часу обробки великих обсягів даних	Прогнозування аномалій у фінансових ринках
Інтеграція з AI	Поєднання алгоритмів виявлення аномалій із штучним інтелектом для покращення точності	Підвищення ефективності та точності	Системи кібербезпеки
Використання інтерпретованих моделей	Розробка моделей, які легко пояснюються та дозволяють зрозуміти причини аномалій	Підвищення довіри до результатів виявлення аномалій	Системи для аудиту та контролю
Адаптивні алгоритми	Алгоритми, які автоматично підлаштовуються під зміни у даних	Поліпшення стійкості до змін	Застосування в реальному часі

Ця таблиця демонструє майбутні напрямки розвитку в області виявлення аномалій, включаючи федероване навчання та квантові обчислення. Наприклад, федероване навчання може суттєво покращити конфіденційність, тоді як квантові обчислення можуть підвищити швидкість обробки даних. Ці новітні технології можуть стати вирішальними в підвищенні ефективності методів виявлення аномалій у найближчому майбутньому[33].

При порівнянні методів виявлення аномалій можна зазначити, що Isolation Forest краще підходить для великих наборів даних завдяки своїй лінійній масштабованості. LOF, навпаки, демонструє вищу точність при роботі з меншими наборами даних і особливо ефективний для виявлення локальних аномалій[34]. Проте в реальних сценаріях, коли важлива обробка даних у реальному часі, метод Isolation Forest має перевагу через свої нижчі обчислювальні витрати.

Таким чином, використання методів виявлення аномалій у великих даних дозволяє не лише попереджати кіберзагрози, але й оптимізувати бізнес-процеси в різних галузях. Однак важливо розуміти обмеження кожного методу та враховувати специфіку конкретної задачі при виборі підходу. Майбутній розвиток технологій, таких як федероване навчання та квантові обчислення, значно змінить підходи до аналізу великих даних і забезпечить нові можливості для підвищення ефективності виявлення аномалій[35].

1.4 Виклики та проблеми виявлення аномалій у кібербезпеці

Масштабування систем для обробки великих обсягів даних у реальному часі є одним із головних викликів виявлення аномалій у кібербезпеці. Великі обсяги інформації вимагають потужної інфраструктури для їх ефективної обробки. Для забезпечення аналізу постійно зростаючих потоків даних традиційні методи, що використовуються для менших масивів, стають непридатними. Наприклад, компанії типу Facebook або Google стикаються з проблемою масштабування систем виявлення кіберзагроз через значні обсяги трафіку, які їхні платформи обробляють щодня. Їхні класичні системи безпеки не здатні забезпечити ефективний аналіз в

умовах таких обсягів даних, що вимагає розробки нових рішень для обробки і виявлення аномалій на нових масштабах.

Однією з найбільших перешкод при аналізі великих обсягів даних є високий рівень шуму. Значна частина трафіку, що аналізується, не має реальної загрози та лише створює додаткове навантаження на системи виявлення аномалій. Наприклад, хибний трафік, спричинений нормальними операціями мережі, часто маскує справжні загрози. Це ускладнює роботу аналітиків, які повинні відфільтровувати цей шум, щоб отримати точні результати. Зменшення шуму вимагає впровадження спеціальних методів фільтрації, але їхня реалізація часто є ресурсно затратною і складною через різноманітність та непередбачуваність самих даних.

Системи виявлення аномалій постійно стикаються з проблемою помилкових спрацьовувань, коли нормальні операції системи класифікуються як потенційні загрози. Це призводить до великої кількості хибних тривог, що негативно впливає на ефективність роботи аналітиків та кібербезпеки загалом. Наприклад, у 2020 році в одній з найбільших фінансових установ США системи виявлення спрацювали помилково, що спричинило значні фінансові втрати через затримки в обробці транзакцій[36]. Подібні випадки призводять до втрат продуктивності та значних фінансових витрат через надмірну кількість перевірок та розслідувань.

Кіберзагрози постійно змінюються, що робить виявлення аномалій складним процесом. Те, що вважалося аномалією вчора, може стати нормою завтра. Моделі виявлення не завжди здатні адаптуватися до таких змін, особливо якщо вони не оновлюються вчасно або є статичними. Наприклад, сучасні фішингові атаки та віруси швидко змінюють свою поведінку, тому системи, що використовують застарілі моделі, часто не встигають виявити нові загрози. Ця еволюція вимагає динамічних і постійно оновлюваних моделей виявлення, впровадження яких часто вимагає значних ресурсів та часу.

Для ефективного навчання моделей виявлення аномалій необхідна велика кількість якісних даних. Проте організації часто не мають достатньої кількості даних з реальних атак або не бажають ділитися такими даними через конфіденційність чи репутаційні ризики. Це негативно впливає на точність моделей, оскільки без великої кількості релевантних даних системи можуть

працювати неефективно. Наприклад, багато компаній відмовляються надавати дані про свої кіберінциденти для досліджень, що ускладнює розвиток більш точних та ефективних систем виявлення.

Дані для виявлення кіберзагроз можуть надходити з різних джерел, таких як мережеві логи, події безпеки або трафік з різних пристроїв. Ці дані часто мають різну структуру, що створює проблеми для аналізу та інтеграції. Проблема непослідовності даних є серйозним викликом для систем виявлення аномалій, оскільки інтероперабельність різних типів даних вимагає складних процесів нормалізації та обробки. Наприклад, мережеві логи можуть мати зовсім іншу структуру порівняно з журналами подій системи безпеки, що ускладнює їх одночасний аналіз.

Багато наявних інструментів для виявлення аномалій не здатні ефективно обробляти великі обсяги даних або працювати з новітніми кіберзагрозами. Наприклад, інструменти на основі сигнатур втрачають актуальність, оскільки не можуть виявляти нові або модифіковані атаки, які не відповідають відомим шаблонам. Технології, що використовуються великими підприємствами на кшталт антивірусних рішень, дедалі більше не відповідають сучасним вимогам через свою обмежену функціональність у виявленні нових загроз.

Оперативність виявлення загроз є критичною, особливо коли йдеться про великі потоки даних у реальному часі. Проте затримка у процесі аналізу даних може зробити систему безпеки менш ефективною. Наприклад, атака на Target у 2013 році стала можливим через те, що система не встигла вчасно проаналізувати дані, що призвело до компрометації мільйонів кредитних карт.

Це підтверджує важливість швидкості обробки даних у системах виявлення аномалій.

Незважаючи на автоматизацію багатьох процесів, роль людини у виявленні аномалій залишається вирішальною. Аналітики повинні вміти інтерпретувати результати автоматизованих систем і приймати правильні рішення на їх основі. Відсутність кваліфікованих фахівців або їхні помилки можуть призвести до неправильного тлумачення результатів і критичних промахів у кіберзахисті. Згідно

зі статистикою, понад 90% інцидентів у кібербезпеці виникають через людський фактор, що підкреслює важливість підготовки персоналу.

Впровадження систем виявлення аномалій потребує значних фінансових вкладень. Наприклад, розробка власних рішень для великих корпорацій, таких як Microsoft, вимагає не лише значних витрат на ліцензії та інфраструктуру, а й постійної підтримки та оновлення систем. Багато компаній не готові інвестувати такі ресурси, що створює додаткові бар'єри для впровадження сучасних рішень у кібербезпеці.

Впровадження ефективних систем виявлення аномалій є критично важливим елементом у забезпеченні кібербезпеки. Проте існуючі виклики, такі як непослідовність даних, обмеження традиційних інструментів, затримки в обробці інформації та недостатня підготовка персоналу, ставлять під загрозу здатність організацій адекватно реагувати на нові кіберзагрози. Для подолання цих труднощів необхідно зосередити зусилля на інтеграції сучасних технологій, таких як машинне навчання та штучний інтелект, які можуть покращити процес виявлення аномалій, прискорити аналіз даних та зменшити залежність від людського фактора.

Крім того, для забезпечення надійності та ефективності систем виявлення аномалій організації повинні інвестувати у навчання та розвиток своїх працівників. Залучення фахівців, які можуть інтерпретувати дані та приймати обґрунтовані рішення, є запорукою успішного кіберзахисту. Оскільки кіберзагрози постійно еволюціонують, компаніям слід переосмислити свої підходи до захисту даних і впровадити комплексні стратегії, що включають технологічні, організаційні та людські аспекти для адаптації до нових викликів.

РОЗДІЛ 2 МЕТОДИ ВИЯВЛЕННЯ АНОМАЛІЙ У ВЕЛИКИХ ДАНИХ ДЛЯ ПРОГНОЗУВАННЯ КІБЕРЗАГРОЗ

2.1 Машинне навчання та його роль у виявленні аномалій

Машинне навчання (ML) є потужною методологією, яка дозволяє комп'ютерним системам вчитися на основі даних і знаходити закономірності без явного програмування кожного кроку. У контексті виявлення аномалій, ML дозволяє виявляти відхилення у великих наборах даних, що може вказувати на потенційні кіберзагрози. Це відбувається завдяки здатності ML обробляти великі масиви інформації, ідентифікувати закономірності та відмінності, які можуть бути непомітними для традиційних методів[37]. Наприклад, у банківській сфері використання ML дозволило знизити кількість шахрайських операцій на 40%, що демонструє ефективність цього підходу.

Машинне навчання можна поділити на три основні категорії: наглядове, ненаглядове навчання та навчання з підкріпленням. Наглядове навчання використовує мічені дані, де система навчається розпізнавати зразки та зв’язки між входом і виходом на основі попередніх даних. Цей підхід ефективний для виявлення відомих загроз, однак обмежується залежністю від точних навчальних даних. Ненаглядове навчання, яке зазвичай використовується для виявлення аномалій, не потребує мічених даних і дозволяє виявляти нові або невідомі загрози. Наприклад, в умовах кіберзахисту ненаглядове навчання здатне виявляти відхилення в мережевому трафіку, що вказує на нові вразливості. Нарешті, навчання з підкріпленням полягає в тому, що алгоритми самонавчаються через взаємодію з навколишнім середовищем, що може бути ефективним у складних динамічних умовах[38].

Для задач виявлення аномалій широко використовуються такі алгоритми:

1. K-means clustering: Цей алгоритм поділяє дані на кластери, групуючи схожі зразки разом. Аномалії визначаються як ті точки, які знаходяться далеко від центрів кластерів. Проте одним з недоліків є те, що цей метод є чутливим до вибору

кількості кластерів та не завжди добре справляється з нерівномірно розподіленими даними.

2. Isolation Forest: Цей алгоритм ізолює зразки, які відрізняються від решти. Він ефективний для виявлення рідкісних подій або відхилень у даних. Його перевагою є висока швидкість роботи навіть з великими наборами даних, проте він може бути менш точним у випадках, коли аномалії складні для ізоляції.

3. One-class SVM: Алгоритм будує модель, яка описує нормальні дані, і будь-які відхилення від цієї моделі вважаються аномаліями. Цей підхід ефективний для задач виявлення рідкісних подій, але може вимагати великих обчислювальних ресурсів для роботи з великими даними. Недоліком є необхідність ретельного налаштування параметрів, що може призводити до перенавчання.

Однією з основних проблем при використанні ML для виявлення аномалій є неповнота або нерівномірність даних. Якщо навчальні дані є недостатньо репрезентативними, система може неправильно ідентифікувати нормальні або аномальні зразки. Перенавчання (overfitting) також є серйозною проблемою, коли модель надто добре навчається на тренувальних даних, що призводить до низької здатності розпізнавати нові аномалії. Наприклад, у реальних умовах перенавчені моделі можуть неправильно оцінювати ризики, що може призвести до пропуску кібератак. Окрім того, масштабованість є важливою проблемою для великих даних. Моделі ML можуть потребувати значних обчислювальних ресурсів, що може впливати на продуктивність систем у реальному часі.

Машинне навчання має ряд переваг порівняно з традиційними методами виявлення аномалій, такими як статистичний аналіз чи евристичні методи. Наприклад, традиційні методи обмежені в обробці великих наборів даних і часто потребують заздалегідь визначених правил для визначення аномалій. Машинне навчання, з іншого боку, дозволяє автоматично адаптуватися до нових загроз і змін у поведінці системи.

Крім того, традиційні методи часто вимагають ручної роботи для налаштування правил, тоді як ML може постійно самонавчатися і вдосконалюватися.

У реальних умовах ML використовується для виявлення аномалій у різних галузях. Наприклад, у банківській сфері системи машинного навчання застосовуються для аналізу транзакцій з метою виявлення шахрайства. Алгоритми ненаглядового навчання можуть швидко ідентифікувати підозрілі транзакції, які відхиляються від звичайних шаблонів[39]. У сфері кібербезпеки мережеві оператори використовують ML для аналізу мережевого трафіку в реальному часі, дозволяючи автоматично виявляти відхилення, що можуть вказувати на кібератаки.

Точність і чутливість є важливими показниками для оцінки ефективності алгоритмів ML. Точність визначає, наскільки добре модель ідентифікує реальні аномалії, тоді як чутливість визначає здатність системи виявляти всі можливі аномалії. Баланс між цими показниками критичний: якщо система надто чутлива, вона може генерувати велику кількість хибних спрацьовувань, що перевантажує ресурси компанії. Якщо ж система буде занадто орієнтована на точність, це може призвести до пропуску важливих загроз.

Використання машинного навчання для виявлення аномалій пов'язане з певними ризиками. Наприклад, великий відсоток хибних тривог може призвести до надмірних витрат на обробку подій і зниження продуктивності системи. Крім того, моделі ML потребують регулярного оновлення, оскільки кіберзагрози постійно змінюються, і старі моделі можуть втрачати актуальність. Регулярне тестування та оновлення моделей є критичними для збереження їхньої ефективності.

Машинне навчання є потужним інструментом для виявлення аномалій у великих даних, проте його ефективність залежить від якості даних і правильного налаштування моделей. Організаціям варто регулярно тестувати та оновлювати свої моделі, а також аналізувати хибні тривоги для підвищення ефективності систем.

2.2 Метод кластеризації для виявлення аномалій

Кластеризація — це метод машинного навчання, який використовується для групування наборів даних на основі їх схожості. У контексті великих даних цей підхід дозволяє поділити дані на кілька підгруп (кластерів), де об'єкти в межах одного кластеру мають більше спільних характеристик між собою, ніж з об'єктами

в інших кластерах. Це допомагає виявляти аномалії як елементи, які значно відрізняються від загальних патернів у даних. Існують різні типи кластеризації, серед яких варто відзначити ієрархічну кластеризацію (де кластери утворюються шляхом об'єднання або поділу на підгрупи) і роздільну кластеризацію (де кластери формуються одночасно шляхом розділення набору даних)[40].

Математичний апарат кластеризації базується на розрахунку відстані між точками даних. Наприклад, метод k-середніх кластеризації використовує формулу евклідової відстані для обчислення схожості між точками. Цей алгоритм намагається мінімізувати суму квадратів відстаней між кожною точкою та її відповідним центроїдом. Алгоритм k-середніх формулюється наступним рівнянням для обчислення центроїда μ_j кластеру j (2.1):

$$\mu_j = \frac{1}{|C_j|} \sum_{x_j \epsilon C_j} x_i \tag{2.1}$$

де C_j — це набір точок в кластері j, а x_j — координати точки даних. Інший популярний метод, DBSCAN (Density-Based Spatial Clustering of Applications with Noise), класифікує точки на основі щільності їх розташування, використовуючи поняття «ядра щільності», і виявляє кластери, що відповідають щільним регіонам даних.

Кластеризація є потужним інструментом для виявлення аномалій, оскільки дозволяє виділяти патерни в даних і визначати об'єкти, які не належать до жодного кластеру або віддалені від центроїдів кластерів. Наприклад, у мережевому моніторингу, якщо з'являються точки даних, які не відповідають жодному кластеру, вони можуть бути класифіковані як потенційні загрози або незвичайні активності[41]. Цей підхід дозволяє автоматично виявляти аномалії без необхідності визначати жорсткі правила або шаблони.

Метод кластеризації здатен виявляти різноманітні типи аномалій, включаючи відхилення в мережевому трафіку, нестандартні транзакції в банківській сфері, або незвичні шаблони поведінки користувачів. Наприклад, у фінансових установах можна використовувати кластеризацію для відстеження транзакцій, що не

відповідають нормальним патернам поведінки клієнтів, що може свідчити про шахрайські операції.

Робота з великими обсягами даних потребує спеціальних алгоритмів, які можуть ефективно обробляти мільйони або мільярди записів. Одним із найпоширеніших методів є k-середніх, який дозволяє швидко і ефективно обробляти великі набори даних, хоча варіанти Mini-Batch k-середніх використовуються для прискорення процесу[42]. DBSCAN, навпаки, є ефективним у виявленні аномалій у великих наборах, оскільки не потребує заздалегідь визначеної кількості кластерів і добре працює з нерівномірно розподіленими даними.

Для оцінки якості кластеризації використовуються кілька метрик. Однією з найпоширеніших є індекс Сілуета, який показує, наскільки близько знаходяться об'єкти до свого кластера порівняно з іншими кластерами. Іншою метрикою є індекс Дейвіса-Болдіна, який оцінює відношення між відстанями всередині кластерів та відстанями між кластерами[43]. Ці метрики допомагають визначити, чи правильно сформовані кластери і наскільки добре аномалії відокремлені від нормальних даних.

Однією з основних проблем кластеризації є вибір оптимальної кількості кластерів, що є критично важливим для досягнення точності. Також виникають проблеми з високою обчислювальною складністю та масштабуванням, особливо при роботі з великими даними. Наприклад, метод k-середніх погано працює з нерівномірно розподіленими даними і може породжувати «шум», який заважає правильному групуванню точок.

У фінансовій сфері кластеризація використовується для виявлення шахрайських транзакцій, які відрізняються від нормальної поведінки користувачів. У кібербезпеці кластеризація застосовується для виявлення аномальної мережевої активності, що може свідчити про вторгнення або інші кіберзагрози. Наприклад, компанії використовують кластеризацію для моніторингу мережевого трафіку і визначення відхилень, що вказують на загрози.

Для реалізації кластеризації у великих даних використовуються платформи, такі як Apache Spark та Hadoop, які здатні ефективно обробляти великі обсяги

інформації. Також використовуються мови програмування Python і R з бібліотеками для машинного навчання, такими як Scikit-learn, TensorFlow та PyTorch, які підтримують реалізацію алгоритмів кластеризації для великих наборів даних.

Метод кластеризації є ефективним підходом для виявлення аномалій у великих даних завдяки здатності автоматично групувати дані та виявляти відхилення. Однак він має свої обмеження, такі як проблеми з вибором кількості кластерів та обчислювальні витрати при роботі з великими наборами даних. Альтернативними підходами можуть бути методи на основі графів або нейронних мереж для виявлення більш складних аномалій.

2.3 Метод побудови регресійних моделей для аналізу аномалій

Регресійні моделі — це статистичний інструмент для аналізу залежностей між змінними, що широко використовується для прогнозування. Основна мета регресії — знайти математичний зв'язок між залежною змінною (цільовою) та однією чи кількома незалежними змінними (факторами). Лінійна регресія є фундаментальним видом регресійних моделей, що використовує рівняння виду (2.2):

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \epsilon \qquad (2.2)$$

де y — залежна змінна, $x_1, x_2, \ldots, x_n$ — незалежні змінні, β_0 — вільний член, $\beta_1, \beta_2, \ldots, \beta_n$— коефіцієнти регресії, а ϵ — випадкова похибка. Лінійна регресія базується на припущенні, що залежність між змінними є лінійною. З її допомогою можна прогнозувати поведінку залежної змінної на основі історичних даних, що є критичним для аналізу аномалій у великих даних.

Для побудови регресійної моделі спочатку необхідно підготувати навчальні дані, що містять залежну змінну та пояснювальні фактори[44]. Коефіцієнти моделі обчислюються через мінімізацію середньоквадратичної помилки (MSE), яка визначається як (2.3):

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.3}$$

де y_i — фактичне значення залежної змінної, $\hat{y}_i$— прогнозоване значення моделі, n — кількість спостережень.

Регресійні моделі використовуються не лише для прогнозування, але й для виявлення аномалій у даних. При побудові моделі на основі історичних даних регресія створює очікуваний діапазон значень для залежної змінної. Відхилення фактичних результатів від прогнозованих значень можуть вказувати на наявність аномалій. Наприклад, у випадку прогнозування навантаження на сервер, якщо фактичне навантаження суттєво перевищує або не відповідає прогнозованому діапазону, це може сигналізувати про можливу кіберзагрозу, таку як DDoS-атака.

Аномалії можуть бути виявлені через аналіз залишків (помилок прогнозування). Якщо залишки значно відрізняються від нуля, це може свідчити про наявність непередбачуваних факторів у даних або неправомірну поведінку системи. Регресійні моделі дозволяють визначати такі відхилення через встановлення порогів допустимих значень залишків.

Великий обсяг даних створює значні виклики для регресійних моделей, оскільки великий набір факторів може призвести до перенавчання та низької продуктивності моделі. Для великих даних використовують спеціальні варіанти регресійних алгоритмів, такі як лінійна регресія з регуляризацією. Метод Lasso (Least Absolute Shrinkage and Selection Operator) мінімізує помилку через штрафування великих коефіцієнтів (2.4):

$$\text{Lasso: } \min_{\beta}\left(\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda\sum_{j=1}^{p}|\beta_j|\right) \tag{2.4}$$

де λ — параметр регуляризації, який контролює ступінь штрафування. Цей підхід допомагає зменшити вплив незначних факторів і зменшити ризик перенавчання.

Крім лінійної регресії, існує кілька інших типів регресійних моделей, які можуть використовуватися для виявлення аномалій у великих даних:

1. Нелінійна регресія — використовується, коли залежність між змінними не є лінійною. Такі моделі здатні краще адаптуватися до складніших даних.

2. Багатофакторна регресія — аналізує вплив кількох факторів одночасно, що важливо для комплексних систем.

3. Логістична регресія — використовується для бінарних випадків, де аномалія може бути визначена як «так» або «ні».

4. Регресія на основі дерева рішень — дозволяє моделювати складні взаємозв'язки між змінними та легко інтерпретується.

Кожен із цих методів має свої переваги і використовується залежно від специфіки завдання. Для кращого розуміння в таблиці 2.1 відображені порівняння регресійних моделей.

Таблиця 2.1. Порівняння різних типів регресійних моделей для аналізу аномалій у великих даних.

Тип регресії	Характеристики	Переваги	Недоліки
Лінійна регресія	Пряма залежність між змінними	Простота та швидкість	Погана адаптація до нелінійних даних
Нелінійна регресія	Нелінійна залежність	Адаптація до складних систем	Ускладненість обчислень
Логістична регресія	Бінарний результат	Чітка інтерпретація результатів	Обмежена тільки для бінарних випадків
Регресія на основі дерева	Моделювання складних взаємозв'язків	Гнучкість та інтерпретація	Можливість перенавчання

Навчання регресійних моделей виконується за допомогою алгоритмів оптимізації. Один з найпоширеніших — метод найменших квадратів, що мінімізує суму квадратів помилок. Для великих даних використовують градієнтний спуск, який ітеративно шукає мінімум функції помилки. Градієнтний спуск дозволяє

обробляти великі обсяги даних за коротший час, але потребує вибору правильного кроку навчання, щоб уникнути застрявання в локальних мінімумах.

Для оцінки якості регресійних моделей використовуються кілька метрик. Середньоквадратична помилка (MSE) визначає середню величину відхилень прогнозованих значень від фактичних. Середня абсолютна похибка (MAE) оцінює середнє абсолютне відхилення, а коефіцієнт детермінації (R^2) показує, наскільки добре модель пояснює змінність залежної змінної. Високі значення R^2 свідчать про високу точність моделі, що є критичним для виявлення аномалій.

Регресійні моделі активно використовуються у сфері кібербезпеки. Один із прикладів — прогнозування DDoS-атак через аналіз мережевого трафіку. Регресія може використовуватися для моделювання нормального трафіку, а відхилення від прогнозованих значень можуть сигналізувати про аномальні атаки. Ідентифікація підозрілої активності базується на тому, що модель виявляє нетипові піки навантаження або неочікувані зниження активності.

Регресійні моделі мають свої обмеження. Вони погано працюють з нелінійними процесами та можуть бути піддані зміщенню через мультиколінеарність, коли незалежні змінні сильно корельовані між собою. Крім того, великі дані можуть включати багато шуму, що ускладнює навчання моделі та призводить до неправильних висновків.

Одним із шляхів покращення регресійних моделей є використання гібридних підходів, таких як поєднання регресії з нейронними мережами. Це дозволяє враховувати складніші взаємозв’язки у великих даних. Крім того, методи кластеризації можуть бути використані для попередньої обробки даних перед регресійним аналізом, що зменшує шум і покращує якість прогнозів.

2.4 Методи на основі глибокого навчання для виявлення аномалій

Глибоке навчання є підгалуззю машинного навчання, яка використовує багатошарові нейронні мережі для обробки великих масивів даних. На відміну від традиційних методів машинного навчання, що залежать від ручного виділення ознак, глибоке навчання автоматично будує ієрархії ознак, що дозволяє знаходити

складні закономірності в даних. Важливим аспектом глибокого навчання є його здатність працювати з великими обсягами даних, що робить його ефективним у сфері кібербезпеки для виявлення кіберзагроз. Оскільки сучасні кіберсистеми генерують значні обсяги інформації, глибоке навчання дозволяє виявляти аномалії, які можуть бути ознаками загроз, що традиційні методи часто пропускають.

У великих даних аномалії можуть бути різних типів, кожен з яких може свідчити про потенційні кіберзагрози. Основні типи аномалій включають:

1. Точкові аномалії — поодинокі дані, що відрізняються від загального набору, наприклад, підозрілий доступ до мережі.
2. Контекстні аномалії — дані, що є нормальними в одному контексті, але аномальними в іншому, наприклад, звичні транзакції, що відбуваються в незвичайний час.
3. Колективні аномалії — набір точок даних, які є нормальними окремо, але разом формують аномальну поведінку, наприклад, послідовність дій, характерна для атак на мережі.

Глибоке навчання ефективно виявляє ці типи аномалій, аналізуючи складні зв’язки між елементами даних і дозволяючи виявляти приховані закономірності.

Автоенкодери є одним із ключових методів для виявлення аномалій. Вони використовують нейронні мережі для стиснення даних у компактне представлення (кодування), після чого намагаються відновити ці дані. Основний принцип дії автоенкодерів полягає у здатності відновлювати нормальні дані з високою точністю, тоді як аномалії відновлюються з більшими похибками. Функція втрат, яка порівнює відновлені дані з оригінальними, дозволяє оцінювати, наскільки добре модель відновлює нормальні дані і чи є певна точка даних аномальною.

LSTM нейронні мережі підходять для роботи з часовими рядами завдяки здатності обробляти залежності у часі. Це робить їх ефективними для виявлення аномалій, що мають тимчасові закономірності, наприклад, послідовність мережевих запитів. Завдяки механізму збереження контексту даних, LSTM дозволяє аналізувати тривалі часові взаємозв'язки, що є критичним у виявленні відхилень у поведінці систем або користувачів.

GANs — це метод, що використовує дві нейронні мережі: генератор і дискримінатор. Генератор створює «нормальні» дані, а дискримінатор навчається розрізняти реальні дані від створених. GANs ефективно виявляють аномалії, порівнюючи згенеровані нормальні дані з реальними спостереженнями. Цей метод дозволяє знаходити відхилення, які складно визначити іншими методами, що робить його потужним інструментом для виявлення нових кіберзагроз.

Навчання моделей глибокого навчання для виявлення аномалій потребує підбору якісних даних. Існують два підходи: з наглядом та без нагляду. Методи з наглядом передбачають навчання на маркованих даних, де відомі приклади нормальних і аномальних точок. Проте через недостатню кількість аномальних прикладів зазвичай використовують безнаглядові методи. Проблеми, що виникають під час навчання, включають класовий дисбаланс (велика кількість нормальних даних порівняно з аномаліями) та нерівномірність даних, що ускладнює побудову ефективної моделі.

Серед переваг методів глибокого навчання є здатність обробляти великі обсяги даних, ефективність у виявленні складних закономірностей і гнучкість у застосуванні до різних типів даних. Однак вони мають і обмеження: висока обчислювальна складність, необхідність великих обсягів даних для навчання, складність інтерпретації результатів і ризик надмірного навчання, що може призвести до неточних висновків.

Ефективність моделей глибокого навчання оцінюють за такими метриками, як precision (точність), recall (повнота), F1-score (збалансована міра) та AUC-ROC (площа під кривою)[45]. Precision показує частку правильно ідентифікованих аномалій серед усіх передбачених, recall — частку справжніх аномалій серед усіх виявлених, а F1-score допомагає збалансувати ці показники. AUC-ROC дозволяє оцінити загальну ефективність моделі при різних порогах. Ці метрики допомагають точно оцінити якість моделей у контексті реальних застосувань.

Глибоке навчання широко використовується для виявлення кіберзагроз. Наприклад, виявлення аномальних мережевих активностей, що можуть свідчити про спроби злому, використовується для захисту інформаційних систем. У банківській сфері глибоке навчання допомагає виявляти підозрілі транзакції, що

можуть бути ознаками шахрайства. Також ефективними є методи для детекції невідомих шкідливих програм через аналіз аномалій у поведінці систем.

Попри досягнення глибокого навчання, існують виклики, пов'язані з високими вимогами до обчислювальних ресурсів, складністю інтерпретації результатів та необхідністю вирішення проблем класового дисбалансу. Майбутні дослідження спрямовані на оптимізацію моделей для зниження їхньої складності, поліпшення інтерпретації та розробку нових архітектур, які ефективніше працюватимуть з невеликими наборами даних.

2.5 Використання нейронних мереж для прогнозування кіберзагроз

Нейронні мережі стали одним із найефективніших інструментів у кібербезпеці для виявлення аномалій і прогнозування загроз. Вони забезпечують високу точність та адаптивність в умовах швидко змінюваних кіберзагроз. Традиційні методи часто не справляються з динамікою та складністю новітніх атак, оскільки мають обмежені можливості аналізу великих обсягів даних і виявлення прихованих закономірностей. На відміну від них, нейронні мережі, особливо з використанням глибокого навчання, можуть розпізнавати складні патерни в даних, що дозволяє ефективніше передбачати потенційні атаки. Завдяки здатності обробляти великий потік інформації і адаптуватися до нових умов, вони стали важливим елементом сучасних систем кіберзахисту.

Існує кілька архітектур нейронних мереж, що використовуються для виявлення кіберзагроз. Згорткові нейронні мережі (CNN) широко застосовуються для аналізу патернів у мережевих даних. Вони ефективно обробляють структуру даних, виявляючи закономірності, що можуть вказувати на аномальну активність. Рекурентні нейронні мережі (RNN), включаючи їхні варіанти, такі як довготривала короткочасна пам'ять (LSTM), використовуються для аналізу послідовностей подій, що є критичним для моніторингу лог-файлів атак. LSTM здатні зберігати інформацію про попередні події, що дозволяє виявляти тривалі та складні загрози. Поєднання різних типів архітектур дозволяє підвищити точність прогнозування кіберзагроз.

Оскільки кібербезпека вимагає аналізу великих обсягів даних, нейронні мережі добре справляються з цією задачею завдяки здатності працювати з різними форматами даних і знаходити аномалії у реальному часі. У процесі обробки великих даних важливим є те, що нейронні мережі здатні виділяти сигнали серед "шуму" та знаходити закономірності, що можуть свідчити про загрозу. Великий обсяг даних, зокрема лог-файли систем, мережевий трафік і метадані файлів, є критично важливим для тренування моделей, оскільки чим більше різноманітних прикладів атак мережа отримує під час навчання, тим точніше вона виявляє нові загрози.

Нейронні мережі навчаються на базових патернах нормальної поведінки і можуть виявляти відхилення, що вказують на потенційні загрози. Наприклад, підозрілі запити до серверів, неприродні зміни у мережевому трафіку, дії, характерні для шкідливого програмного забезпечення, можуть бути розпізнані як аномалії. Нейронні мережі здатні аналізувати величезні обсяги даних у реальному часі, виявляючи приховані загрози, які могли б залишитися непоміченими за допомогою традиційних методів.

Для прогнозування кіберзагроз використовуються різні методи навчання нейронних мереж: супервізоване, напівсупервізоване та безсупервізоване навчання. У випадку супервізованого навчання моделі тренуються на даних з відомими мітками атак, що дозволяє класифікувати нові загрози. Безсупервізоване навчання використовується для виявлення нових типів загроз, коли немає явних міток, що вказують на аномальні дані. Це дозволяє мережі автоматично класифікувати нові зразки поведінки як загрози або норму.

Збір даних для навчання нейронних мереж у кібербезпеці включає різні джерела, такі як логи систем, мережевий трафік, файли і метадані. Важливим етапом є підготовка даних: очищення від "шуму", нормалізація та трансформація їх у зручний для аналізу формат. Неправильна підготовка даних може призвести до виникнення хибних позитивних або негативних спрацьовувань, що знижує ефективність моделі. Тому важливо забезпечити якісну попередню обробку даних перед навчанням моделей.

Ефективність нейронних мереж для виявлення кіберзагроз оцінюється за допомогою метрик, таких як точність, відчутливість (recall), специфічність та F-

міра. Важливо знаходити баланс між виявленням загроз і мінімізацією хибних спрацювань, особливо в умовах великих обсягів даних. Модель повинна виявляти найбільшу кількість потенційних загроз, при цьому зберігаючи низький рівень хибних сигналів, що є критично важливим для ефективної роботи систем кіберзахисту.

Нейронні мережі в кібербезпеці стикаються з рядом проблем, серед яких перенавчання (overfitting), коли модель занадто точно відтворює патерни навчальних даних і не здатна адаптуватися до нових загроз. Також є проблема швидкого розвитку нових видів атак, через що моделі потребують постійного оновлення. Крім того, складність і неоднорідність великих даних ускладнюють тренування та тестування моделей, що потребує додаткових ресурсів і часу.

Великі компанії, такі як банки та дата-центри, активно використовують нейронні мережі для захисту своїх систем. Наприклад, банки застосовують нейронні мережі для аналізу транзакцій у реальному часі, виявляючи підозрілі операції та захищаючи від шахрайства. Великі дата-центри використовують ці технології для моніторингу мережевої активності і виявлення аномалій, що можуть свідчити про підготовку до атаки.

У майбутньому нейронні мережі можуть використовуватися у поєднанні з іншими технологіями, такими як штучний інтелект, для підвищення точності прогнозування загроз. Розробка нових архітектур і методів навчання дозволить ще ефективніше боротися з кіберзагрозами, зокрема у поєднанні з іншими інноваційними технологіями, що дозволить створювати системи з проактивним виявленням загроз.

РОЗДІЛ 3 ПРАКТИЧНІ АСПЕКТИ ЗАСТОСУВАННЯ МЕТОДІВ ВИЯВЛЕННЯ АНОМАЛІЙ У ВЕЛИКИХ ДАНИХ

3.1 Огляд наявних інструментів для обробки великих даних

Сховища даних та бази даних є фундаментальними компонентами сучасних систем обробки великих даних, забезпечуючи ефективне зберігання, управління та доступ до величезних обсягів інформації. Розподілена файлова система Hadoop HDFS виступає однією з ключових технологій у цій сфері, забезпечуючи високу доступність та надійність даних шляхом розподілу їх на численні вузли кластеру, що дозволяє горизонтальне масштабування та стійкість до відмов (див. рисунок 3.1).

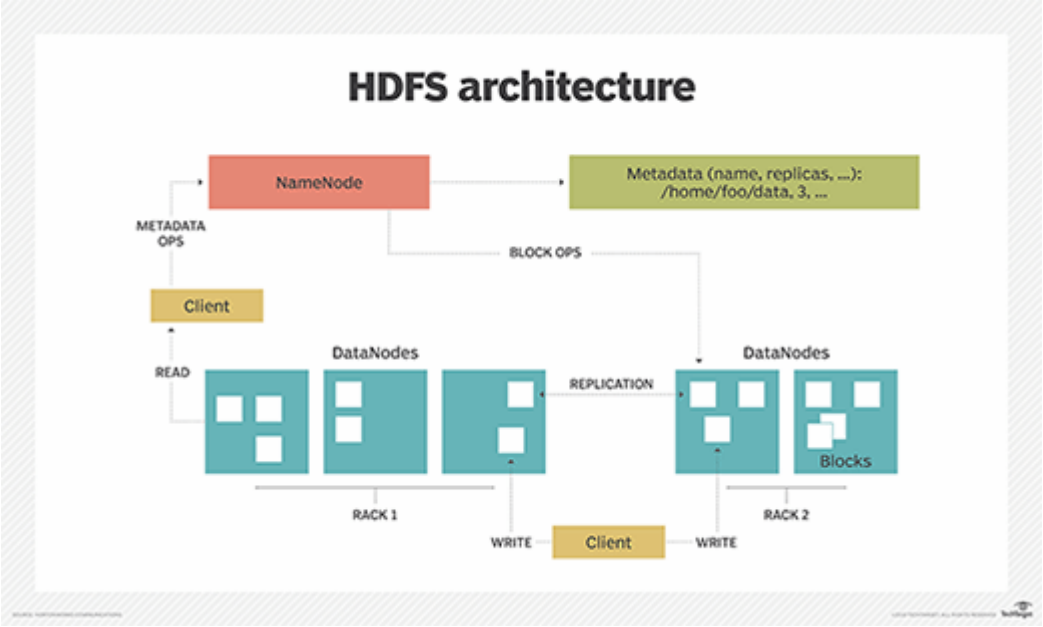

Рисунок 3.1 – Архітектура HDFS

Ця система оптимізована для обробки великих обсягів даних шляхом паралельного виконання завдань, що значно підвищує продуктивність аналізу. У контексті NoSQL баз даних, такі рішення як MongoDB, Cassandra та HBase надають гнучкість у моделюванні даних, що особливо важливо для нереляційних структур, що виникають у великих даних. MongoDB забезпечує документно-орієнтоване зберігання, що дозволяє швидко адаптуватися до змінних вимог даних, тоді як

Cassandra пропонує високу доступність та розподілену архітектуру, що підтримує масштабування без втрати продуктивності. HBase, побудована на основі Hadoop, забезпечує можливості для реалізації випадково доступних записів у великомасштабних таблицях, що є необхідним для обробки великих потоків даних (див. рисунок 3.2).

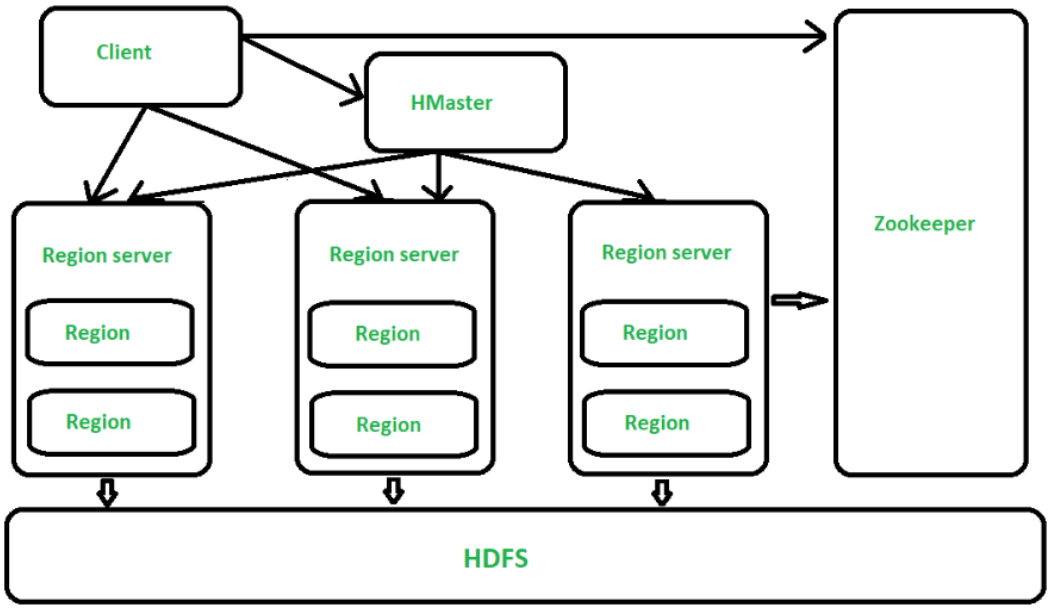

Рисунок 3.2 – Архітектура HBase

Ці технології спільно забезпечують основу для побудови гнучких, масштабованих та ефективних систем зберігання даних, здатних підтримувати складні аналітичні завдання та великі обсяги інформації, що генеруються в сучасних бізнес-середовищах. Вибір конкретного рішення залежить від специфічних вимог до структури даних, швидкості доступу, масштабованості та стійкості до відмов, що робить розуміння їхніх характеристик критично важливим для оптимальної організації систем обробки великих даних.

Фреймворки для обробки даних відіграють ключову роль у сучасній інформаційній екосистемі, забезпечуючи ефективні механізми для обробки великих обсягів структурованих та неструктурованих даних. Одним із найпопулярніших рішень є Apache Spark, який відзначається своєю здатністю виконувати обробку даних у пам'яті, що суттєво підвищує швидкість аналізу порівняно з традиційними

дисковими методами. Spark підтримує різноманітні парадигми обробки, включаючи пакетну обробку, потокову обробку та інтерактивний аналіз, що робить його універсальним інструментом для різних типів задач. Крім того, Spark надає широкий набір бібліотек для машинного навчання, графових обчислень та обробки потокових даних, що дозволяє здійснювати комплексний аналіз даних без необхідності інтеграції сторонніх компонентів.

Іншим значущим фреймворком є Apache Flink, який спеціалізується на обробці потокових даних у реальному часі з високою пропускною здатністю та низькою затримкою. Flink забезпечує точність обчислень та підтримку складних станів обробки, що робить його ідеальним для застосувань, де критично важлива своєчасність та точність даних, таких як фінансові транзакції або моніторинг промислових процесів. Архітектура Flink дозволяє масштабувати обробку горизонтально, забезпечуючи високу доступність та стійкість до відмов, що є важливими аспектами для корпоративних середовищ

Hadoop MapReduce залишається фундаментальним компонентом екосистеми великих даних, забезпечуючи масштабовану пакетну обробку великих наборів даних через розподілені обчислення (див. рисунок 3.3).

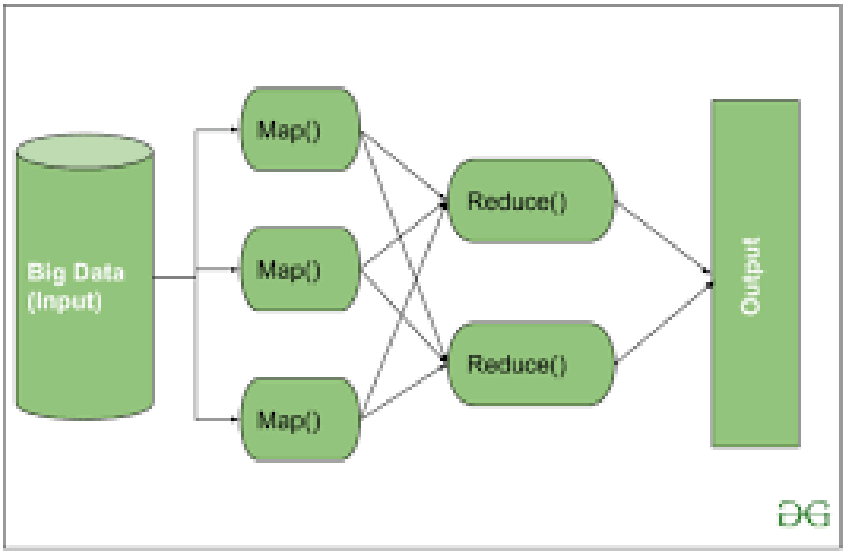

Рисунок 3.3 – Архітектура Hadoop MapReduce

Хоча MapReduce може бути менш ефективним у порівнянні з більш сучасними фреймворками, такими як Spark, він все ще широко використовується

завдяки своїй стабільності та інтеграції з іншими компонентами Hadoop, такими як HDFS для зберігання даних. MapReduce оптимізований для завдань, що вимагають високої надійності та повторюваності, забезпечуючи автоматичне відновлення після збоїв вузлів.

Інтеграція цих фреймворків з іншими технологіями, такими як системи управління базами даних, інструменти для обробки даних у реальному часі та платформи для машинного навчання, дозволяє створювати потужні та гнучкі рішення для аналізу великих даних. Важливими аспектами при виборі відповідного фреймворку є вимоги до продуктивності, тип даних, що обробляються, масштабованість рішення та специфічні бізнес-потреби. Таким чином, сучасні фреймворки для обробки даних надають широкі можливості для ефективного управління та аналізу великих обсягів інформації, сприяючи прийняттю обґрунтованих рішень у різних сферах діяльності.

Інструменти для завантаження та інтеграції даних відіграють ключову роль у процесі обробки великих даних, забезпечуючи ефективне перенесення, об'єднання та підготовку даних з різноманітних джерел для подальшого аналізу. Одним із таких інструментів є Apache Kafka, яка забезпечує високопродуктивну платформу для обробки потокових даних, дозволяючи створювати надійні та масштабовані потоки інформації в реальному часі (див. рисунок 3.4).

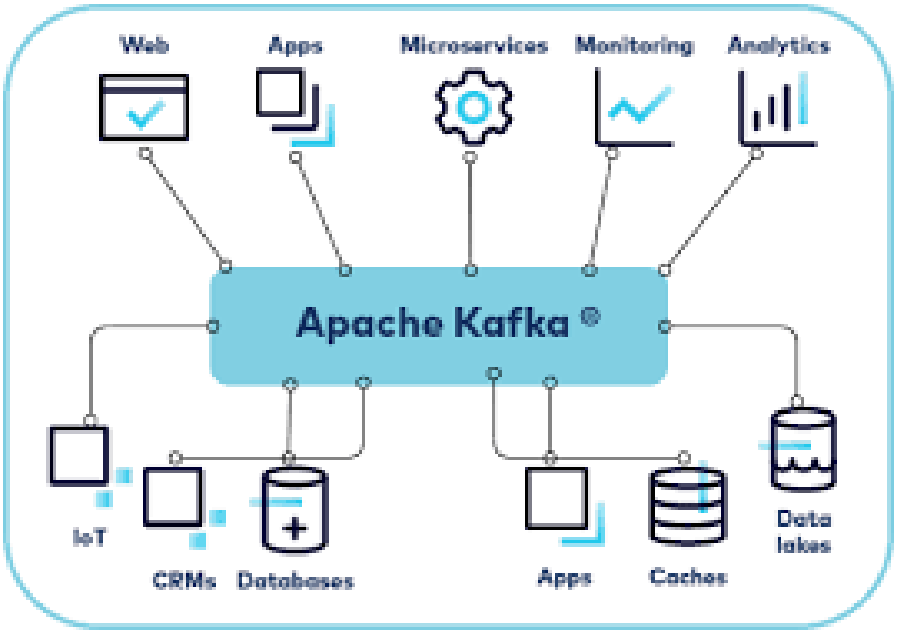

Рисунок 3.4 – Архітектура Apache Kafka

Kafka дозволяє ефективно управляти великими обсягами даних, забезпечуючи високу пропускну здатність та низьку затримку, що є критично важливим для застосунків, які вимагають моментальної обробки інформації.

Ще одним важливим інструментом є Apache NiFi, який призначений для автоматизації потоків даних між різними системами (див. рисунок 3.5).

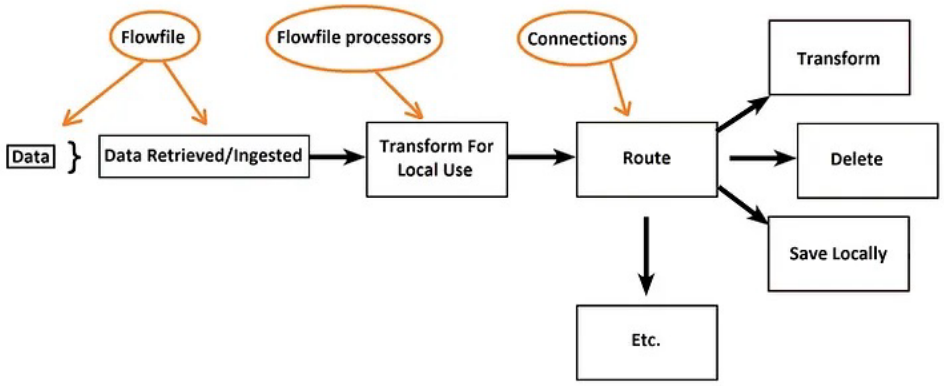

Рисунок 3.5 – Архітектура Apache NiFi

NiFi підтримує візуальне налаштування процесів інтеграції, що дозволяє користувачам легко створювати, керувати та моніторити складні пайплайни даних без необхідності глибоких знань програмування. Він забезпечує широкі можливості для трансформації, маршрутизації та обробки даних, підтримуючи різноманітні формати та протоколи передачі інформації.

Інтеграція даних може також здійснюватися за допомогою інструментів ETL (Extract, Transform, Load), таких як Talend та Informatica. Talend є відкритим інструментом, який пропонує гнучкі можливості для інтеграції та перетворення даних, підтримуючи численні підключення до різних джерел даних та дозволяючи автоматизувати складні процеси обробки інформації. Informatica, з іншого боку, надає комплексну платформу для управління даними, включаючи функції очищення, збагачення та синхронізації даних, що забезпечує високу якість та узгодженість інформації в рамках організаційних процесів.

Крім того, існують хмарні сервіси, такі як Amazon Kinesis та Google Cloud Dataflow, які пропонують інтегровані рішення для завантаження та обробки даних у хмарних середовищах. Ці сервіси забезпечують масштабованість та гнучкість, дозволяючи організаціям швидко адаптуватися до змін у обсягах даних та вимогах до обробки без необхідності вкладати значні ресурси в інфраструктуру.

Загалом, вибір інструментів для завантаження та інтеграції даних залежить від конкретних вимог проекту, включаючи обсяги даних, необхідність обробки в реальному часі, сумісність з існуючими системами та вимоги до масштабованості. Використання сучасних рішень у цій сфері дозволяє забезпечити ефективну та надійну інтеграцію даних, що є фундаментальним аспектом успішного аналізу великих даних та прийняття обґрунтованих бізнес-рішень.

Інструменти ETL (Extract, Transform, Load) відіграють ключову роль у процесі обробки великих даних, забезпечуючи ефективне витягування, перетворення та завантаження даних з різноманітних джерел до цільових систем зберігання або аналітики. Витягування передбачає збір даних з різних джерел, таких як реляційні бази даних, файли, веб-сервіси та інші системи, забезпечуючи їх інтеграцію в єдине середовище. Перетворення включає очищення даних, нормалізацію, агрегування, а також виконання складних обчислювальних операцій, необхідних для приведення даних до потрібного формату та структури, що забезпечує їхню готовність для подальшого аналізу. Завантаження здійснюється шляхом перенесення оброблених даних до цільових сховищ, таких як сховища даних або аналітичні платформи, де вони можуть бути використані для прийняття бізнес-рішень або проведення наукових досліджень.

Інструменти ETL, такі як Talend та Informatica, забезпечують автоматизацію цих процесів, що сприяє підвищенню ефективності та зменшенню ризиків, пов'язаних з ручною обробкою даних. Вони підтримують широкий спектр джерел даних та цільових систем, забезпечуючи гнучкість та масштабованість, необхідні для роботи з великими обсягами інформації. Крім того, сучасні ETL-платформи часто включають можливості для управління метаданими, моніторингу процесів та забезпечення безпеки даних, що є важливими аспектами у середовищі великих даних.

Значення ETL-інструментів полягає в їх здатності забезпечувати високоякісну підготовку даних для аналізу, що є фундаментальним для ефективного використання великих даних у різних галузях. Вони сприяють інтеграції даних з різних джерел, забезпечують їхню консистентність та точність, що є критично важливими для отримання надійних аналітичних результатів. Таким чином, інструменти ETL виступають невід'ємною складовою сучасних систем обробки великих даних, забезпечуючи необхідну інфраструктуру для їхньої ефективної обробки та аналізу.

Хмарні сервіси для обробки великих даних стали критично важливими компонентами сучасної інформаційної інфраструктури, забезпечуючи масштабованість, гнучкість та ефективність у роботі з великими обсягами даних. Одним із провідних постачальників у цій сфері є Amazon Web Services (AWS), яка пропонує різноманітні інструменти, такі як Amazon EMR, що дозволяє здійснювати розподілену обробку даних за допомогою фреймворків Hadoop та Spark. Крім того, сервіс Amazon Redshift надає можливості для аналітичних запитів та зберігання даних у колонковому форматі, що оптимізує швидкість доступу та обробки великих масивів інформації.

Google Cloud Platform (GCP) також займає важливе місце на ринку хмарних послуг для великих даних, пропонуючи рішення, як-от BigQuery, який є високопродуктивним аналітичним складом даних, здатним обробляти терабайти та петабайти даних з низькою затримкою. BigQuery підтримує стандартні SQL-запити та інтегрується з іншими сервісами GCP, що забезпечує гнучкість у виконанні складних аналітичних завдань. Крім того, Google Cloud Dataflow надає можливості для обробки потокових та пакетних даних, забезпечуючи ефективну обробку в реальному часі.

Microsoft Azure також пропонує комплексні рішення для обробки великих даних, серед яких виділяється Azure HDInsight, що надає керовані кластерні рішення для Hadoop, Spark, та інших фреймворків. Azure також інтегрує ці сервіси з іншими своїми продуктами, такими як Azure Data Lake Storage, який забезпечує масштабоване зберігання даних, та Azure Synapse Analytics, який об'єднує аналітичні можливості та управління даними в єдину платформу. Це дозволяє

організаціям ефективно аналізувати великі обсяги даних, виконувати складні запити та отримувати цінні інсайти для прийняття рішень.

Інші хмарні платформи, такі як IBM Cloud та Oracle Cloud, також пропонують свої рішення для обробки великих даних, включаючи можливості для інтеграції з існуючими бізнес-процесами та забезпечення високої доступності та безпеки даних. Наприклад, IBM Cloud використовує рішення на основі Hadoop та Spark, а також пропонує інструменти для машинного навчання та штучного інтелекту, що дозволяє проводити глибоку аналітику та прогнозування на основі великих даних. Oracle Cloud, у свою чергу, фокусується на оптимізації баз даних та аналітичних процесів, забезпечуючи високу продуктивність та масштабованість для корпоративних клієнтів.

Усі ці хмарні сервіси спільно сприяють розвитку екосистеми великих даних, дозволяючи організаціям ефективно зберігати, обробляти та аналізувати дані незалежно від їхнього розміру та складності. Використання хмарних рішень забезпечує гнучкість у виборі необхідних ресурсів, дозволяє скоротити витрати на інфраструктуру та підвищує швидкість впровадження аналітичних проектів. Таким чином, хмарні сервіси стають невід’ємною частиною стратегії управління великими даними, підтримуючи інновації та конкурентоспроможність організацій у різних галузях економіки.

Інструменти для аналітики та бізнес-інтелекту відіграють ключову роль у перетворенні великих обсягів даних на корисну інформацію, що підтримує прийняття рішень у підприємствах. Вони забезпечують можливість збору, обробки, візуалізації та аналізу даних з різних джерел, сприяючи глибшому розумінню бізнес-процесів та виявленню нових можливостей для розвитку.

Одним із провідних інструментів у цій сфері є Tableau, який відзначається високою гнучкістю та інтуїтивністю у створенні візуалізацій. Завдяки потужним можливостям інтеграції з різними джерелами даних, Tableau дозволяє користувачам швидко створювати інтерактивні дашборди та звіти, що полегшує аналіз складних даних та виявлення тенденцій. Це сприяє підвищенню оперативності прийняття рішень та підсилює аналітичні здібності організації.

Power BI від Microsoft також займає значне місце серед інструментів бізнес-інтелекту, пропонуючи широкі можливості для інтеграції з іншими продуктами Microsoft, такими як Excel та Azure. Power BI характеризується високою продуктивністю у обробці великих обсягів даних та підтримкою розширених функцій аналітики, включаючи машинне навчання та штучний інтелект. Це дозволяє користувачам не лише створювати звіти та візуалізації, а й проводити глибокий аналіз даних для прогнозування майбутніх тенденцій.

Іншим важливим інструментом є Qlik Sense, який надає можливість асоціативного аналізу даних, що дозволяє користувачам вільно досліджувати інформацію без необхідності заздалегідь визначених шляхів аналізу. Цей підхід сприяє виявленню прихованих зв’язків між різними наборами даних, що може призвести до відкриття нових інсайтів та покращення бізнес-стратегій.

Domo представляє собою комплексну платформу для бізнес-інтелекту, яка об'єднує в собі функції збору, обробки та візуалізації даних у реальному часі. Це забезпечує безперервний доступ до актуальної інформації, що є критично важливим для оперативного управління та швидкого реагування на зміни ринкових умов. Domo також підтримує інтеграцію з численними сторонніми додатками та сервісами, що розширює її функціональні можливості.

Інструменти для аналітики та бізнес-інтелекту також включають SAP BusinessObjects, який відомий своєю потужною системою звітності та аналітики. SAP BusinessObjects дозволяє створювати детальні звіти та аналізувати дані з високою точністю, що сприяє оптимізації бізнес-процесів та покращенню загальної ефективності організації.

Використання цих інструментів дозволяє підприємствам ефективно управляти своїми даними, проводити глибокий аналіз та приймати обґрунтовані рішення на основі достовірної інформації. Це, в свою чергу, сприяє підвищенню конкурентоспроможності та забезпеченню сталого розвитку в умовах швидкозмінного ринку.

Платформи машинного навчання та штучного інтелекту відіграють ключову роль у сучасній обробці великих даних, забезпечуючи інструменти та середовища для розробки, тренування та впровадження моделей, здатних аналізувати та

інтерпретувати складні набори даних. Однією з провідних платформ є TensorFlow, що розроблена компанією Google та забезпечує гнучку та масштабовану інфраструктуру для створення глибоких нейронних мереж, підтримуючи як навчання на великих наборах даних, так і їхнє розгортання в різноманітних середовищах. Apache Mahout, інтегрований з екосистемою Hadoop, пропонує бібліотеки для реалізації масштабованих алгоритмів машинного навчання, що дозволяє ефективно обробляти великі обсяги даних розподіленим способом. Платформи, такі як PyTorch, надають високорівневі інтерфейси для розробки моделей машинного навчання, сприяючи швидкому прототипуванню та експериментуванню завдяки своїй динамічній обчислювальній графіці.

Сучасні платформи інтегрують інструменти для автоматизації процесів навчання моделей, оптимізації гіперпараметрів та управління життєвим циклом моделей, що значно підвищує ефективність роботи аналітиків даних та дослідників. Наприклад, платформи як Microsoft Azure Machine Learning та Google AI Platform забезпечують хмарну інфраструктуру, що дозволяє масштабувати обчислювальні ресурси відповідно до потреб проекту, а також інтегруються з іншими сервісами для зберігання, обробки та візуалізації даних. Важливою характеристикою сучасних платформ є підтримка розподіленого обчислення, що дозволяє обробляти великі обсяги даних паралельно, знижуючи час тренування моделей та підвищуючи їх точність.

Крім того, платформи машинного навчання активно використовують методи автоматизованого машинного навчання (AutoML), що спрощують процес створення моделей шляхом автоматичного вибору та налаштування оптимальних алгоритмів та параметрів. Це дозволяє залучати фахівців з різними рівнями підготовки до розробки ефективних моделей, знижуючи бар'єри для впровадження машинного навчання в різні галузі. Інтеграція з інструментами для обробки природної мови, комп’ютерного зору та інших спеціалізованих областей розширює можливості платформ, дозволяючи вирішувати широкий спектр завдань, від аналізу текстових даних до розпізнавання образів та прогнозування поведінки користувачів.

Таким чином, платформи машинного навчання та штучного інтелекту забезпечують комплексні рішення для ефективної обробки та аналізу великих

даних, сприяючи розвитку інноваційних технологій та їхньому впровадженню в різні сфери діяльності. Їхня здатність адаптуватися до різноманітних вимог проектів, підтримка масштабованості та інтеграція з іншими інструментами обробки даних роблять їх незамінними компонентами сучасних систем обробки великих даних.

Інструменти для управління та оркестрації даних відіграють ключову роль у забезпеченні ефективного та узгодженого процесу обробки великих обсягів інформації. Одним із провідних рішень у цій сфері є Apache Airflow, який дозволяє автоматизувати та планувати складні робочі процеси шляхом визначення залежностей між завданнями у вигляді оркестрованих графів. Ця система забезпечує високу гнучкість завдяки можливості інтеграції з різноманітними джерелами даних та підтримці різних мов програмування, що сприяє адаптації до специфічних вимог проектів. Airflow також надає потужні інструменти для моніторингу та управління виконанням завдань, що дозволяє виявляти та усувати потенційні проблеми в режимі реального часу.

Іншим значущим інструментом є Luigi, розроблений компанією Spotify, який спеціалізується на створенні складних пайплайнів даних. Цей інструмент підтримує організацію завдань у вигляді залежностей та забезпечує контроль над їх виконанням, що дозволяє ефективно керувати потоками даних та забезпечувати їхню надійність. Luigi відзначається простотою використання та можливістю інтеграції з різними системами зберігання та обробки даних, що робить його придатним для широкого спектру застосувань.

Крім того, сучасні платформи, такі як Kubernetes, активно використовуються для оркестрації контейнеризованих додатків, включаючи сервіси обробки даних. Завдяки автоматичному масштабуванню, розподілу навантаження та забезпеченню високої доступності, Kubernetes дозволяє ефективно управляти ресурсами та забезпечувати безперебійну роботу систем обробки великих даних.

Важливим аспектом управління та оркестрації даних є також забезпечення безпеки та відповідності нормативним вимогам. Інструменти забезпечують механізми автентифікації, авторизації та шифрування даних, що гарантує захист конфіденційної інформації під час її обробки та передачі між різними

компонентами системи. Це особливо актуально у середовищах, де обробка даних здійснюється у хмарних інфраструктурах, де контроль доступу та захист даних мають вирішальне значення.

Таким чином, інструменти для управління та оркестрації даних забезпечують структурований та автоматизований підхід до обробки великих обсягів інформації, сприяють підвищенню ефективності операцій, забезпечують масштабованість та надійність систем, а також гарантують безпеку та відповідність стандартам. Вибір конкретного інструменту залежить від специфічних потреб проекту, існуючої інфраструктури та вимог до інтеграції з іншими компонентами системи.

3.2 Вибір алгоритмів для виявлення аномалій у кіберзагрозах

На рисунку 3.6 зображено алгоритм Isolation Forest.

Рисунок 3.6 – Алгоритм Isolation Forest

Алгоритм Isolation Forest характеризується ефективним підходом до виявлення аномалій шляхом ізоляції спостережень у високорозмірному просторі даних. Процес розпочинається зі збору та завантаження даних, після чого здійснюється їх попередня обробка, що включає очищення та нормалізацію, забезпечуючи готовність даних до подальшої обробки. Наступним етапом є визначення параметрів моделі, таких як кількість дерев у лісі (`n_estimators`) та максимальна глибина кожного дерева (`max_depth`), що впливає на здатність алгоритму до детекції аномалій та загальну продуктивність моделі.

Після налаштування параметрів починається процес побудови дерев рішень. Для кожного дерева випадково вибирається підмножина ознак даних, після чого створюється дерево рішень, яке служить для ізоляції окремих спостережень. Кожне дерево генерується шляхом послідовного розділення даних на вузли, де аномалії, як правило, ізолюються на менших глибинах дерева порівняно з нормальними спостереженнями. Це зумовлено тим, що аномалії, маючи рідкісні або відхилені характеристики, швидше виділяються у процесі випадкових розділень.

Після побудови всіх дерев у лісі проводиться обчислення середньої глибини ізоляції для кожного зразка даних. Ця середня глибина нормалізується для подальшого аналізу, що дозволяє оцінити, наскільки легко кожне спостереження було ізольовано у всіх деревах. Чим менша нормалізована середня глибина ізоляції, тим більша ймовірність того, що спостереження є аномалією, оскільки воно було ізольовано на ранніх етапах побудови дерев.

Наступним кроком є визначення порогового значення для ідентифікації аномалій. Це поріг встановлюється таким чином, щоб відокремити спостереження з низькою середньою глибиною ізоляції від решти даних. Спостереження, значення яких нижчі за встановлений поріг, маркуються як аномалії. Цей підхід дозволяє ефективно виявляти нетипові зразки, навіть у великих наборах даних з високою розмірністю.

Заключним етапом алгоритму є завершення процесу обробки даних та представлення результатів виявлення аномалій. Всі аномальні спостереження,

визначені на основі порогового значення, відображаються для подальшого аналізу або прийняття заходів щодо забезпечення безпеки інформаційних систем. Таким чином, алгоритм Isolation Forest забезпечує структурований та автоматизований підхід до виявлення аномалій, використовуючи механізм ізоляції для ефективного та масштабованого аналізу великих обсягів даних.

Загалом, алгоритм Isolation Forest відзначається високою продуктивністю при обробці великих та складних наборів даних, забезпечуючи швидке та точне виявлення аномалій без необхідності попереднього визначення структури даних або попередньої нормалізації. Його здатність до автоматичної адаптації до різних характеристик даних робить його цінним інструментом у сфері кібербезпеки та інших галузях, де виявлення аномалій є критично важливим для забезпечення цілісності та безпеки інформаційних систем.

На рисунку 3.7 зображено алгоритм One-Class Support Vector Machines (One-Class SVM).

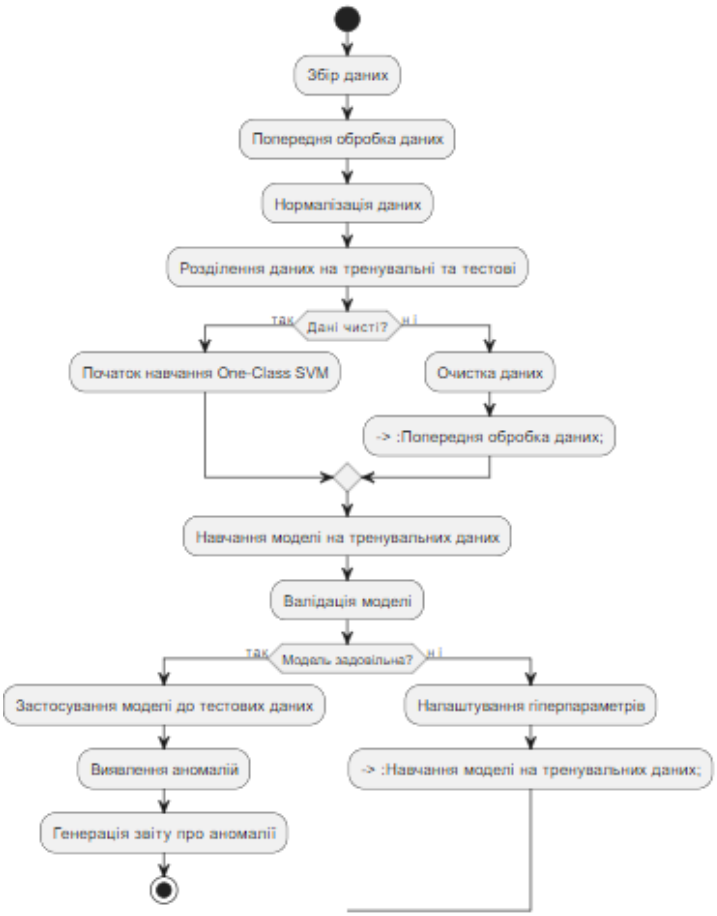

Рисунок 3.7 – Алгоритм One-Class Support Vector Machines (One-Class SVM)

Діаграма активності, що ілюструє алгоритм One-Class Support Vector Machines (One-Class SVM), відображає послідовність етапів, необхідних для ефективного виявлення аномалій у наборі даних. Процес починається зі збору даних, де агрегуються всі релевантні інформаційні ресурси, необхідні для подальшого аналізу. Після збору даних здійснюється їх попередня обробка, що включає очищення від шумів, видалення пропущених значень та вирішення проблеми несумісних форматів, що забезпечує високу якість даних для навчання моделі. Наступним кроком є нормалізація даних, яка полягає у приведенні всіх ознак до єдиного масштабу, що дозволяє уникнути домінування окремих ознак при навчанні моделі та підвищує її здатність до узагальнення.

Далі відбувається розділення даних на тренувальні та тестові набори, що є критичним для оцінки продуктивності моделі на невідомих даних. У разі, якщо дані не є достатньо чистими, процес повертається до етапу попередньої обробки, забезпечуючи таким чином ітеративний підхід до покращення якості даних. Після успішного розділення даних починається навчання моделі One-Class SVM на тренувальних даних. Цей етап включає оптимізацію параметрів моделі з метою максимально точного визначення меж нормальної поведінки у багатовимірному просторі ознак.

Після навчання модель піддається валідації, де її ефективність оцінюється на тестових даних. Якщо модель демонструє задовільні результати, переходить до застосування її до нових даних для виявлення аномалій. У випадку, якщо результати валідації не відповідають очікуванням, здійснюється налаштування гіперпараметрів моделі, після чого модель перенавчається на тренувальних даних. Циклічний процес налаштування та перенавчання продовжується до досягнення необхідного рівня точності.

Фінальним етапом є застосування моделі до нових даних, де вона прогнозує, чи належать вони до нормальної поведінки, чи є аномалією. Виявлені аномалії документуються у звіті, який слугує основою для подальшого аналізу або прийняття відповідних заходів. Таким чином, алгоритм One-Class SVM, зображений на діаграмі активності, забезпечує структурований та систематизований підхід до виявлення аномалій, поєднуючи методи обробки даних,

навчання моделі та її валідації для досягнення високої точності та надійності у виявленні кіберзагроз.

Процес починається зі збору даних, після чого здійснюється їх попередня обробка та нормалізація, що забезпечує придатність даних для навчання моделі. Далі дані розділяються на тренувальні та тестові набори, після чого модель One-Class SVM навчається на чистих тренувальних даних, встановлюючи межі нормальної поведінки. Після навчання модель проходить валідацію на тестових даних, де оцінюється її здатність правильно класифікувати нові зразки. Якщо модель демонструє задовільні результати, вона застосовується до нових даних для виявлення аномалій. У разі незадовільної роботи моделі здійснюється налаштування гіперпараметрів та повторне навчання, що дозволяє підвищити точність моделі. Завершальним етапом є генерація звіту про виявлені аномалії, що забезпечує подальший аналіз та реагування на можливі кіберзагрози. Таким чином, діаграма активності відображає послідовний та ітеративний характер алгоритму One-Class SVM, спрямований на забезпечення високої ефективності та надійності у виявленні аномальних патернів у даних.

На рисунку 3.8 зображено алгоритм Autoencoders.

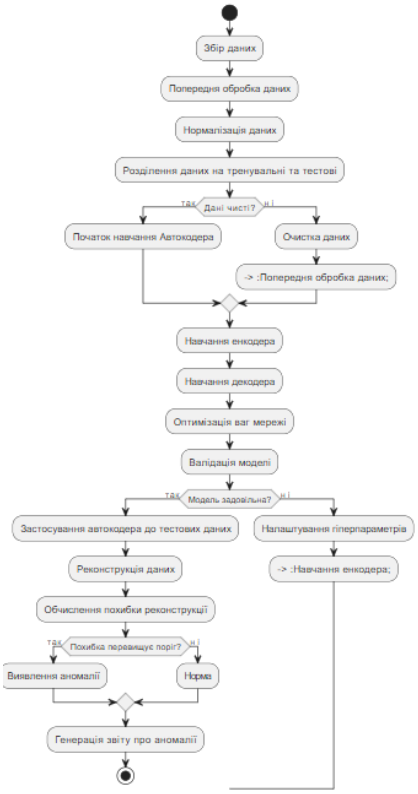

Рисунок 3.8 – Алгоритм Autoencoders

Діаграма активності, що відображає алгоритм Автокодерів, детально ілюструє послідовність етапів, необхідних для ефективного виявлення аномалій у наборі даних. Процес починається зі збору даних, де агрегуються всі релевантні інформаційні ресурси, необхідні для подальшого аналізу. Зібрані дані піддаються попередній обробці, яка включає очищення від шумів, видалення пропущених значень та вирішення проблем несумісних форматів. Цей етап забезпечує високу якість даних, що є критично важливим для подальшого навчання моделі. Після очищення даних здійснюється їх нормалізація, тобто приведення всіх ознак до єдиного масштабу. Це необхідно для уникнення домінування окремих ознак під час навчання моделі та покращення її здатності до узагальнення.

Наступним кроком є розділення даних на тренувальні та тестові набори. Це дозволяє окремо використовувати частину даних для навчання моделі та іншу для її валідації, що є ключовим для оцінки продуктивності моделі на невідомих даних. У разі, якщо дані не відповідають необхідним вимогам якості, процес повертається до етапу попередньої обробки, забезпечуючи ітеративний підхід до покращення якості даних. Після успішного розділення даних починається навчання автокодера, яке включає тренування енкодера та декодера. Енкодер зменшує розмірність вхідних даних, створюючи компактне представлення, тоді як декодер намагається відновити вихідні дані з цього зменшеного представлення. Оптимізація ваг мережі здійснюється з метою мінімізації похибки реконструкції, що дозволяє моделі ефективно захоплювати основні патерни даних.

Після завершення навчання модель піддається валідації, де оцінюється її здатність точно відтворювати дані на тестовому наборі. Якщо модель демонструє задовільні результати, вона застосовується до тестових даних для виявлення аномалій. Процес включає реконструкцію тестових даних за допомогою автокодера та обчислення похибки реконструкції для кожного зразка. Аномалії визначаються як ті точки, для яких похибка реконструкції перевищує встановлений поріг, що вказує на їхню відмінність від нормальних патернів даних. Виявлені аномалії

документуються у звіті, який служить основою для подальшого аналізу або прийняття відповідних заходів щодо забезпечення інформаційної безпеки.

У випадку, якщо результати валідації не відповідають очікуванням, здійснюється налаштування гіперпараметрів моделі, після чого процес навчання повторюється з метою досягнення необхідного рівня точності. Циклічний характер цього процесу дозволяє поступово покращувати модель шляхом оптимізації параметрів та повторного навчання, що забезпечує високий рівень точності та надійності у виявленні аномальних патернів у даних.

Таким чином, алгоритм Автокодерів, відображений на діаграмі активності, забезпечує структурований та систематизований підхід до виявлення аномалій. Він поєднує методи обробки даних, навчання нейронних мереж та їхньої валідації для досягнення високої ефективності та надійності. Послідовність етапів, від збору даних до генерації звіту про аномалії, демонструє інтегровану систему, яка дозволяє ефективно ідентифікувати відхилення від нормальних патернів, що є критично важливим для забезпечення кібербезпеки та захисту інформаційних ресурсів.

На рисунку 3.9 зображено алгоритм K-Means Clustering (Кластеризація методом k-середніх).

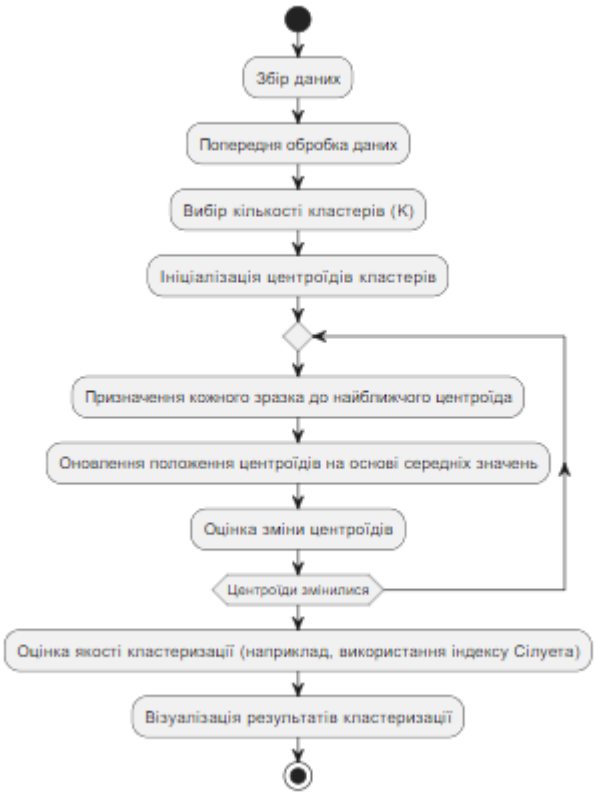

Рисунок 3.9 – Алгоритм Autoencoders K-Means Clustering

Процес починається зі збору даних, де агрегуються всі релевантні інформаційні ресурси, необхідні для подальшого аналізу. Після збору даних здійснюється їх попередня обробка, яка включає очищення від пропущених значень, видалення шуму та приведення даних до єдиного формату, що забезпечує високу якість і однорідність інформації для подальшої обробки.

Наступним етапом є вибір кількості кластерів (K), що може здійснюватися на основі попередніх знань про дані або за допомогою методів визначення оптимальної кількості кластерів, таких як метод ліктя. Після визначення параметра K відбувається ініціалізація центроїдів кластерів, що може виконуватись випадковим вибором точок з набору даних або використанням більш складних методів, наприклад, алгоритму k-means++ для покращення початкової конфігурації центроїдів.

Після ініціалізації центроїдів здійснюється призначення кожного зразка до найближчого центроїда на основі обраної метрики відстані, зазвичай Евклідової. Це призводить до формування початкових кластерів, де кожен кластер містить

зразки, найближчі до відповідного центроїда. Після призначення зразків до кластерів проводиться оновлення положення центроїдів шляхом обчислення середнього значення ознак усіх зразків у кожному кластері, що дозволяє визначити нові центри мас кластерів.

Цей процес призначення та оновлення повторюється ітеративно до тих пір, поки зміни в положенні центроїдів не стануть мінімальними або не буде досягнуто максимальної кількості ітерацій, що свідчить про досягнення стану конвергенції. Під час кожної ітерації оцінюється зміна центроїдів, що дозволяє визначити, чи слід продовжувати процес кластеризації.

Після завершення ітеративного процесу проводиться оцінка якості кластеризації, наприклад, за допомогою індексу Сілуета, який вимірює, наскільки добре кожен зразок вписується у свій кластер порівняно з іншими кластерами. Ця оцінка дозволяє визначити ефективність розподілу зразків між кластерами та ідентифікувати можливі проблеми з перегрупуванням.

Заключним етапом є візуалізація результатів кластеризації, яка дозволяє наочно оцінити структуру кластерів та взаємне розташування зразків у просторі ознак. Візуалізація сприяє інтерпретації отриманих кластерів та їх відповідності очікуванням або доменним знанням, що є важливим для подальшого аналізу та прийняття рішень на основі отриманих результатів.

3.3 Розробка моделі для прогнозування кіберзагроз на основі аномальних даних

Розробка моделі для прогнозування кіберзагроз на основі аномальних даних передбачає кілька послідовних етапів, кожен з яких спрямований на забезпечення точності, надійності та ефективності системи. Початковим кроком є завантаження та попередня обробка даних, що є критично важливими для подальшого аналізу. Дані, що містять різні ознаки мережевого трафіку або поведінкові патерни, завантажуються за допомогою бібліотеки `pandas`, після чого здійснюється їх нормалізація для уніфікації масштабів ознак. Це досягається шляхом обчислення середнього значення та стандартного відхилення для кожної ознаки, після чого

кожне значення масштабуються за формулою (X_new - mean) / std_replaced, де std_replaced запобігає діленню на нуль, замінюючи стандартне відхилення, рівне нулю, на одиницю.

Після нормалізації дані розділяються на тренувальний та тестовий набори за допомогою власної реалізації функції train_test_split, яка випадковим чином перемішує індекси записів та розподіляє їх відповідно до заданого співвідношення. Це забезпечує репрезентативність обох наборів для подальшого навчання та оцінки моделі. Наступним етапом є реалізація алгоритму кластеризації методом k-середніх (K-Means) без використання зовнішніх бібліотек. Ініціалізація центроїдів здійснюється випадковим вибором точок з тренувального набору даних, після чого кожен зразок асоціюється з найближчим центроїдом на основі евклідової відстані. Позиції центроїдів оновлюються шляхом обчислення середнього значення ознак усіх зразків, що належать до відповідного кластера. Цей процес повторюється до досягнення конвергенції, коли зміни у положенні центроїдів стають незначними або досягається максимальна кількість ітерацій.

Після завершення навчання модель піддається оцінці на тестових даних. Для кожного зразка обчислюється відстань до найближчого центроїда, після чого встановлюється поріг для виявлення аномалій, наприклад, на основі 95-го перцентиля розподілу відстаней у тренувальному наборі даних. Записи, що мають відстань, що перевищує цей поріг, класифікуються як аномалії, що можуть свідчити про потенційні кіберзагрози. Якщо у тестових даних присутні мітки класів, здійснюється обчислення метрик точності, precision, recall та F1-score, а також формування матриці плутанини, яка надає детальний аналіз результатів класифікації.

Для забезпечення зручності використання моделі створюється графічний інтерфейс користувача (GUI) за допомогою бібліотеки Tkinter. Інтерфейс включає кнопки для завантаження даних, їх попередньої обробки, навчання моделі, оцінки результатів та візуалізації. Наприклад, завантаження даних реалізується за допомогою функції, що відкриває діалогове вікно для вибору файлу, після чого дані завантажуються у pandas DataFrame та відображаються у текстовій зоні логів для відстеження процесу. Попередня обробка активується кнопкою, яка запускає

відповідну функцію для нормалізації та розділення даних, після чого користувач може перейти до навчання моделі.

Навчання моделі здійснюється у фоновому режимі за допомогою модуля threading, що запобігає заморожуванню GUI під час тривалих обчислень. Наприклад, функція train_model створює окремий потік для виконання алгоритму K-Means, в той час як головний потік залишається відкритим для взаємодії з користувачем. Під час навчання у текстовій зоні логів відображаються повідомлення про прогрес, такі як призначення кластерів, оновлення центроїдів та досягнення конвергенції. Це дозволяє користувачу відстежувати етапи виконання моделі в режимі реального часу.

Після завершення навчання здійснюється оцінка моделі на тестових даних. Результати оцінки відображаються у вигляді текстових повідомлень, що включають обчислені метрики класифікації та матрицю плутанини, яка формується за допомогою власної реалізації функції create_confusion_matrix. Візуалізація результатів здійснюється за допомогою бібліотек matplotlib та seaborn, де матриця плутанини представлена у вигляді теплової карти, що дозволяє візуально оцінити точність класифікації. Додатково може бути побудована діаграма розподілу прогнозів аномалій, що надає інтуїтивне розуміння ефективності моделі у виявленні потенційних загроз.

Усі етапи процесу розробки супроводжуються детальним логуванням, що забезпечує прозорість та можливість виявлення та виправлення помилок на ранніх стадіях. Наприклад, у випадку виникнення помилок при завантаженні даних або під час навчання моделі, відповідні повідомлення відображаються у текстовій зоні логів, що дозволяє користувачу швидко ідентифікувати та усунути проблему.

Таким чином, розробка застосунку для прогнозування кіберзагроз на основі аномальних даних включає комплексний підхід, що охоплює завантаження, обробку та аналіз даних, навчання та оцінку моделі кластеризації, а також створення зручного інтерфейсу для взаємодії з користувачем. Цей підхід забезпечує високий рівень точності та надійності у виявленні аномалій, що потенційно можуть представляти кіберзагрози, сприяючи тим самим підвищенню рівня безпеки мережевих систем.

3.4 Тестування і оцінка ефективності моделей виявлення аномалій

Для тестування моделі виявлення аномалій було зібрано набір даних, що представлено на рисунку 3.10.

	A	B	C	D	E
1	Feature1,Feature2,Label				
2	52.483570765056164,50.4853877467402,0.0				
3	49.308678494144075,54.84322495266444,0.0				
4	53.23844269050346,46.48973453061324,0.0				
5	57.61514928204013,48.36168926701116,0.0				
6	48.82923312638332,48.03945923433921,0.0				
7	48.8293152152541,42.68242525933941,0.0				
8	57.896064077536955,51.48060138532288,0.0				
9	53.837173645764544,51.30527636089945,0.0				
10	47.65262807032524,50.025567283212304,0.0				
11	52.712800217929825,48.82706433312426,0.0				
12	47.68291153593769,42.923146289747926,0.0				
13	47.671351232148716,47.89677338617321,0.0				
14	51.20981135783017,48.286427417366156,0.0				
15	40.43359877671101,45.9886136538919,0.0				
16	41.375410837434835,49.19357144166995,0.0				
17	47.188562353795135,52.02025428407269,0.0				
18	44.93584439832788,59.43092950605265,0.0				
19	51.57123666297637,50.872889064159196,0.0				
20	45.45987962239394,51.28775195361382,0.0				
21	42.93848149332354,49.62777042116917,0.0				
22	57.32824384460777,40.40614392350479,0.0				
23	48.87111849756732,49.867430622753915,0.0				
24	50.337641023439616,50.30115104970513,0.0				
25	42.87625906893272,62.316210562426434,0.0				
26	47.27808637737409,49.03819517609439,0.0				
27	50.554612948549334,51.50773671166806,0.0				

Рисунок 3.10 – Набір даних

Синтетичний набір даних, представлений у форматі CSV-файлу, призначений для моделювання та аналізу поведінки мережевого трафіку з метою виявлення потенційних кіберзагроз. Цей набір містить три основні параметри, які відображають різні аспекти діяльності в мережі: Feature1, Feature2 та Label.

Feature1 і Feature2 представляють собою числові показники, що відображають характеристики мережевого трафіку. Наприклад, Feature1 може відповідати кількості запитів до певного сервера за певний проміжок часу, що відображає активність користувачів або системних процесів. Feature2 може

відображати обсяг переданих даних або кількість спроб доступу до ресурсів мережі, що також є критичними параметрами для оцінки безпеки мережі

Мітка Label використовується для класифікації записів у два категорії: нормальні (значення 0) та аномальні (значення 1). Нормальні записи відповідають типовій та очікуваній активності в мережі, що не викликає підозр з точки зору безпеки. Аномальні записи, навпаки, відображають нетипові або підозрілі патерни поведінки, які можуть свідчити про потенційні кіберзагрози, такі як спроби несанкціонованого доступу, розповсюдження шкідливого програмного забезпечення або інші форми атак.

Набір даних складається з 200 * 10^6 записів, де 85% з них класифіковані як нормальні, а 15% — як аномальні. Нормальні дані групуються в два різні кластери, що дозволяє моделі навчитися розпізнавати різні типи нормальної поведінки в мережі. Аномальні дані розташовані далеко від цих кластерів, що імітує ситуації, коли зловмисні дії мають значні відхилення від звичайної активності, роблячи їх легкою ціллю для алгоритму кластеризації.

З метою збалансування даних, було штучно згенеровано в існуючому дата сеті кількість аномальних записів та видалено кілька записів що свідчать про нормальний трафік, завдяки цьому вдалося досягнти збалансованості датасету що включа 65% даних нормальних 35% аномальних.

Цей набір даних використовується для навчання та тестування моделі кластеризації методом k-середніх, яка спрямована на виявлення аномалій у мережевому трафіку. Під час навчання модель визначає центроїди кластерів нормальної поведінки, після чого застосовується до тестових даних для класифікації нових записів. Відстань кожного зразка до найближчого центроїда використовується як міра його схожості з нормальним патерном. Записи, що мають відстань, що перевищує встановлений поріг, класифікуються як аномалії, що дозволяє ідентифікувати потенційні кіберзагрози.

Таким чином, синтетичний набір даних створює контрольоване середовище для розробки та оцінки алгоритмів виявлення аномалій, забезпечуючи можливість аналізу як нормальної, так і підозрілої активності в мережі. Це сприяє підвищенню ефективності систем безпеки, дозволяючи своєчасно ідентифікувати та реагувати

на потенційні загрози, що виникають у динамічному середовищі мережевих комунікацій.

Застосунок для виявлення аномалій представлено на рисунку 3.11.

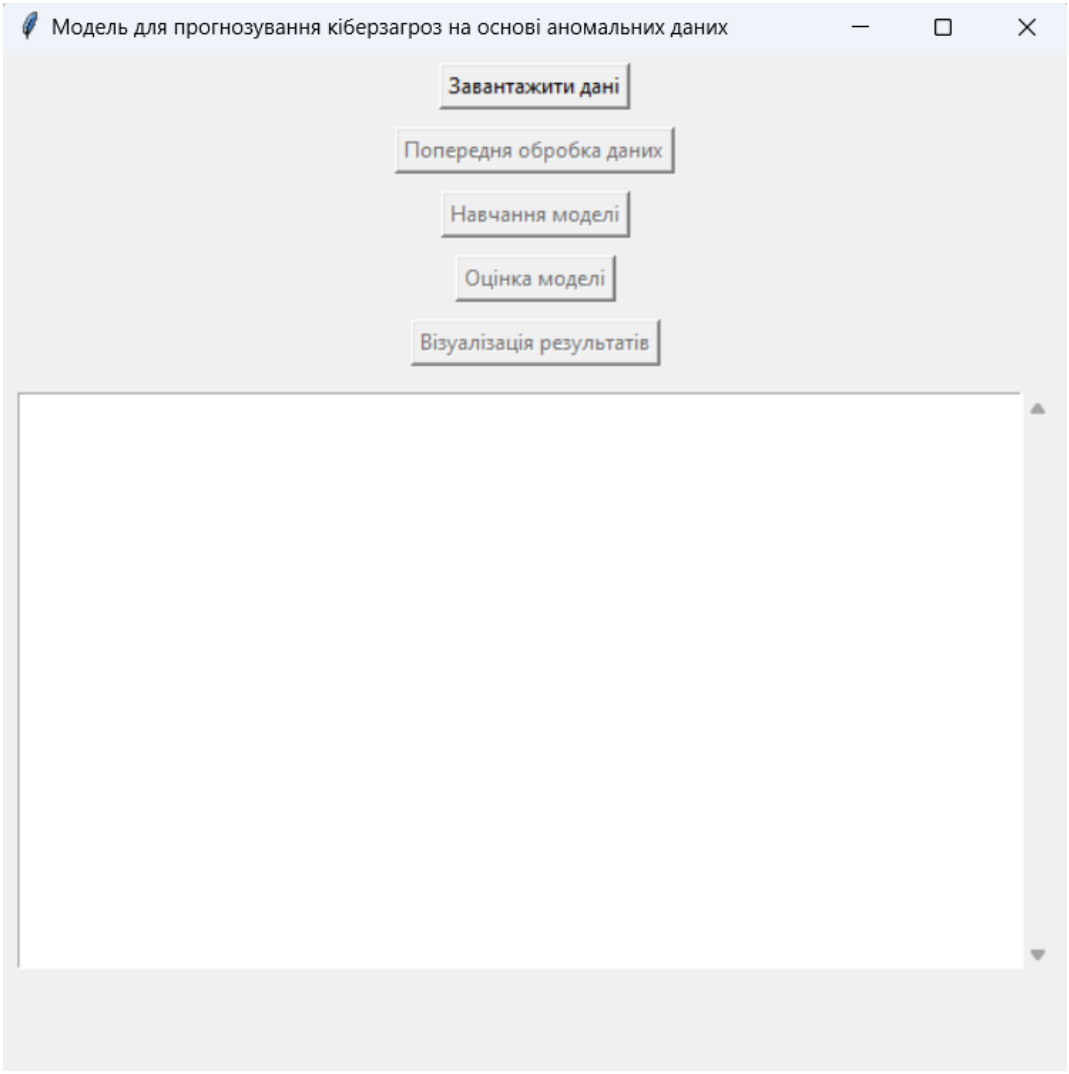

Рисунок 3.11 – Застосунок для виявлення аномалій у великих даних

Процес тестування створеного застосунку для прогнозування кіберзагроз на основі аномальних даних охоплює кілька ключових етапів, кожен з яких спрямований на забезпечення точності, надійності та ефективності моделі кластеризації методом k-середніх у виявленні потенційних загроз. Спочатку здійснюється завантаження синтетичних тестових даних, які представляють собою комплексну симуляцію нормальної та аномальної поведінки в мережевому середовищі. Ці дані містять дві числові ознаки, що відображають різні аспекти мережевого трафіку, та мітку класу, яка розрізняє нормальні записи від аномалій.

Завантаження даних відбувається через графічний інтерфейс користувача, де здійснюється перевірка коректності формату файлу та цілісності даних (див. рисунок 3.12).

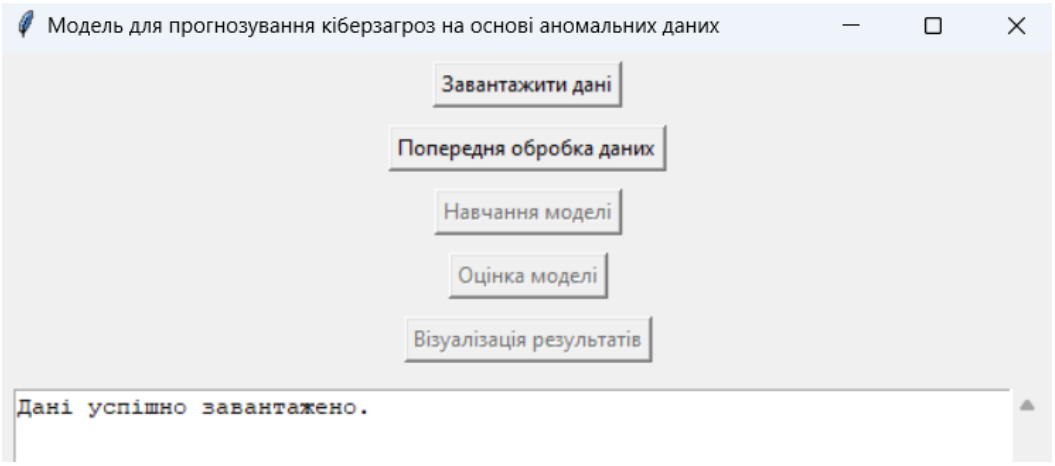

Рисунок 3.12 – Результат завантаження даних

Після успішного завантаження проводиться етап попередньої обробки даних, що включає нормалізацію ознак для забезпечення їх однорідності та покращення ефективності алгоритму кластеризації (див. рисунок 3.13).

```
Виконується нормалізація даних...
Нормалізація завершена.
Розділення даних на тренувальні та тестові набори...
Дані розділено: 140 тренувальних та 60 тестових зразків.
```

Рисунок 3.13 – Процес обробки даних

Нормалізація здійснюється за допомогою стандартного масштабування, де від кожного значення віднімається середнє значення та ділиться на стандартне відхилення відповідної ознаки. Це дозволяє уникнути домінування одних ознак над іншими під час обчислення відстаней між точками даних і центроїдами кластерів. Розділення даних на тренувальні та тестові набори проводиться випадковим перемішуванням записів із фіксованим співвідношенням, що забезпечує репрезентативність обох наборів для подальшого навчання та оцінки моделі.

Навчання моделі здійснюється за допомогою реалізації алгоритму k-середніх, де ініціалізація центроїдів відбувається випадковим вибором точок з тренувального набору даних (див. рисунок 3.14).

```
Оновлено центроїд для кластера 2: [-0.64613302 -0.67979041
-0.33333333]
Ітерація 5: Призначення кластерів...
Ітерація 5: Оновлення центроїдів...
Оновлено центроїд для кластера 0: [2.53424392 2.51236674 3.        ]
Оновлено центроїд для кластера 1: [ 0.2060876   0.10138629
-0.33333333]
Оновлено центроїд для кластера 2: [-0.65109324 -0.68760005
-0.33333333]
Ітерація 6: Призначення кластерів...
Ітерація 6: Оновлення центроїдів...
Оновлено центроїд для кластера 0: [2.53424392 2.51236674 3.        ]
Оновлено центроїд для кластера 1: [ 0.2060876   0.10138629
-0.33333333]
Оновлено центроїд для кластера 2: [-0.65109324 -0.68760005
-0.33333333]
Конвергенція досягнута.
Алгоритм завершився на ітерації 6.
Навчання моделі завершено.
```

Рисунок 3.14 – Процес навчання моделі

Призначення кластерів проводиться шляхом обчислення евклідових відстаней між кожним зразком і центроїдами, після чого кожен зразок асоціюється з найближчим центроїдом. Оновлення позицій центроїдів відбувається шляхом обчислення середнього значення ознак усіх зразків, що належать до відповідного кластера. Цей процес повторюється до досягнення конвергенції, коли зміни у положенні центроїдів стають незначними або досягається максимальна кількість ітерацій. Під час кожної ітерації фіксується прогрес, що дозволяє користувачу відстежувати процес навчання в режимі реального часу через логову зону у графічному інтерфейсі.

Після завершення навчання модель піддається оцінці на тестових даних. Призначення кластерів для тестових записів виконується аналогічним чином, після чого обчислюються відстані до найближчих центроїдів (див. рисунок 3.15).

```
Почато оцінку моделі...
Призначення кластерів для тестових даних завершено.
Обчислення відстаней до центроїдів...
Визначено поріг для аномалій: 0.8409
Виявлено 3 аномалій у тестових даних.
Кількість виявлених аномалій: 3
```

Рисунок 3.15 – Процес оцінки моделі

Визначення порогу для виявлення аномалій здійснюється на основі статистичної оцінки, наприклад, за допомогою 95-го перцентиля розподілу відстаней у тренувальному наборі даних. Записи, відстань яких перевищує цей поріг, класифікуються як аномалії. Якщо у тестових даних присутні мітки класів, модель оцінюється за допомогою метрик точності, precision, recall та F1-score, а також формується матриця плутанини для детального аналізу результатів класифікації. У випадку відсутності міток здійснюється лише кількісна оцінка виявлених аномалій, що дозволяє оцінити потенційну ефективність моделі у реальних умовах.

Візуалізація результатів включає побудову матриці плутанини у вигляді теплової карти, яка демонструє кількість правильних і неправильних класифікацій, а також діаграму розподілу прогнозів аномалій (див. рисунок 3.16). Ці графіки надають візуальне підтвердження ефективності моделі, дозволяючи користувачу легко ідентифікувати патерни виявлення аномалій та оцінити загальну якість класифікації.

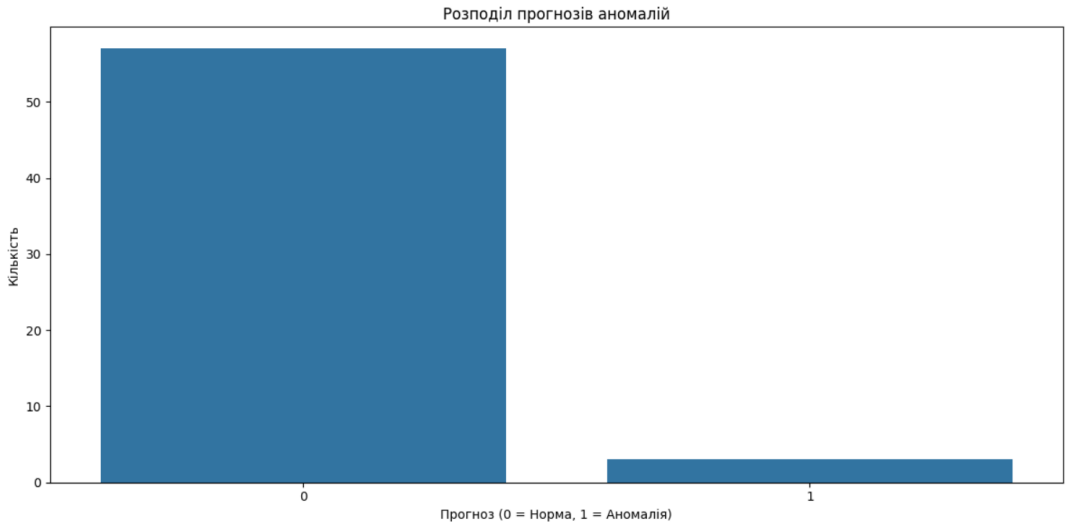

Рисунок 3.16 – Візуалізація результатів

Усі етапи тестування супроводжуються докладним логуванням процесів, що дозволяє користувачу бачити детальну інформацію про кожен крок виконання, включаючи успішність завантаження даних, процес нормалізації, деталі ітерацій алгоритму кластеризації, а також результати оцінки та візуалізації. Це забезпечує прозорість процесу та дозволяє вчасно виявляти та виправляти можливі помилки чи неточності в роботі застосунку.

Таким чином, процес тестування створеного застосунку є всебічним та детальним, охоплюючи всі аспекти від завантаження та попередньої обробки даних до навчання, оцінки та візуалізації результатів моделі. Такий підхід забезпечує високу якість та надійність розробленої системи для виявлення кіберзагроз на основі аномальних даних, дозволяючи ефективно ідентифікувати потенційні загрози та реагувати на них у динамічному середовищі мережевих комунікацій.

3.5 Оптимізація методів для підвищення точності прогнозування кіберзагроз

Оптимізація методів для підвищення точності прогнозування кіберзагроз є критично важливою складовою сучасних систем безпеки інформаційних технологій. Вона передбачає систематичний підхід до вдосконалення різних аспектів моделі, починаючи від якості даних та закінчуючи вибором оптимальних алгоритмів аналізу. Основним завданням є забезпечення максимальної здатності моделі виявляти аномалії, які можуть вказувати на потенційні загрози, при мінімізації кількості хибнопозитивних та хибнонегативних результатів.

Перш за все, важливо зосередитися на якості вхідних даних. Це включає ретельну попередню обробку, яка забезпечує очищення даних від шуму та заповнення пропущених значень. Застосування методів нормалізації або стандартизації допомагає уніфікувати масштаб різних ознак, що є особливо важливим для алгоритмів кластеризації, таких як K-Means. Наприклад, стандартне масштабування, яке полягає у відніманні середнього значення та діленні на стандартне відхилення, дозволяє зменшити вплив ознак з великими варіаціями на результат кластеризації.

Далі, ефективне вибір та інженерія ознак можуть значно покращити здатність моделі розпізнавати аномалії. Це може включати виділення нових ознак шляхом комбінування існуючих або застосування методів зменшення розмірності, таких як головні компоненти аналізу (PCA), для виділення найбільш інформативних компонентів даних. Впровадження доменних знань у процесі інженерії ознак дозволяє створити більш релевантні та розрізняльні ознаки, що підвищує точність класифікації

Вибір оптимальної кількості кластерів є ще одним важливим аспектом оптимізації. Метод ліктя, який аналізує залежність суми квадратів відстаней до найближчого центроїда від кількості кластерів, може допомогти визначити оптимальне значення параметра K. Крім того, ініціалізація центроїдів за допомогою алгоритму K-Means++ забезпечує кращу стартову позицію для алгоритму, що може призвести до швидшої конвергенції та більш стабільних результатів кластеризації.

Алгоритмічна оптимізація також грає ключову роль у підвищенні точності моделі. Це включає удосконалення самого алгоритму кластеризації, наприклад, шляхом впровадження механізмів запобігання локальним мінімумам, або застосування адаптивних критеріїв зупинки, які дозволяють більш точно визначити момент досягнення конвергенції. Крім того, розширення алгоритму для обробки великих наборів даних може значно зменшити час навчання без втрати точності.

Важливо також інтегрувати методи оцінки та валідації моделі, які дозволяють об'єктивно оцінити її ефективність. Використання метрик, таких як precision, recall, F1-score, а також матриця плутанини, надає глибше розуміння того, як модель працює з різними типами даних. Крім того, застосування крос-валідації дозволяє забезпечити стабільність результатів та уникнути перенавчання моделі на тренувальних даних.

Для подальшого підвищення точності прогнозування може бути доцільним використання ансамблевих методів або комбінування різних алгоритмів аналізу. Наприклад, комбінування кластеризації з іншими методами виявлення аномалій, такими як Isolation Forest або DBSCAN, може забезпечити більш комплексний підхід до ідентифікації загроз. Це дозволяє моделі враховувати різні аспекти та характеристики даних, що підвищує її загальну ефективність.

Застосування методів машинного навчання з налаштуванням гіперпараметрів також може суттєво вплинути на точність моделі. Використання алгоритмів оптимізації, таких як grid search або random search, дозволяє автоматизувати процес підбору оптимальних параметрів, що забезпечує максимальну продуктивність моделі. Крім того, регуляризація може допомогти уникнути перенавчання, зберігаючи при цьому здатність моделі до генералізації на нових даних.

Для оцінки якості моделей було використано показники accuracy, prescision, recall.

Таблиця 3.1 - Результати оцінки якості моделей-класифікаторів

Модель	Метрики якості		
	A, %	P, %	R, %
k-means	0,982	0,984	0,912
SVM	0, 971	0,977	0, 963
Isolative Forest	0,843	0,913	0,751

Усі ці підходи повинні бути ретельно інтегровані у процес розробки та тестування моделі, забезпечуючи постійний моніторинг та аналіз її продуктивності. Впровадження системи логування та візуалізації результатів дозволяє користувачу детально відстежувати процес оптимізації та своєчасно виявляти потенційні проблеми.

3.6 Порівняння традиційних і сучасних методів виявлення аномалій

Традиційні методи виявлення аномалій, такі як статистичні підходи та методи кластеризації, забезпечують базову основу для ідентифікації відхилень у даних шляхом аналізу їхніх розподілів та структури. Наприклад, метод k-середніх кластеризації дозволяє групувати дані у відповідності до їхньої схожості, що спрощує виявлення аномальних точок, які віддалені від основних кластерів. Однак ці методи часто обмежені у здатності адаптуватися до складних та високовимірних даних, характерних для сучасних мережевих середовищ, де аномалії можуть бути

багатовимірними та не завжди явно відокремленими від нормальної поведінки. Сучасні методи, зокрема ті, що базуються на машинному навчанні та глибоких нейронних мережах, пропонують значно більш високу гнучкість та здатність до самонавчання, що дозволяє ефективніше виявляти складні та невідомі патерни аномалій. Наприклад, автоенкодери можуть навчатися реконструювати нормальні дані, і будь-які суттєві відхилення в реконструкції сигналізуватимуть про аномалії, що робить їх особливо потужними для виявлення нових або рідкісних кіберзагроз.

Створений застосунок для прогнозування кіберзагроз на основі аномальних даних використовує інтеграцію традиційних та сучасних методів для досягнення високої точності та надійності. Завдяки реалізації алгоритму k-середніх кластеризації забезпечується ефективне групування даних, що дозволяє виявляти суттєві відхилення, які можуть свідчити про потенційні загрози. Водночас, завдяки додатковим механізмам нормалізації та інженерії ознак, застосунок здатний адаптуватися до різноманітних типів даних та умов, що робить його універсальним інструментом у сфері кібербезпеки. Використання графічного інтерфейсу з можливістю детального відстеження процесів обробки та навчання моделі забезпечує прозорість та контроль над кожним етапом аналізу, що сприяє підвищенню довіри користувачів до результатів. Крім того, оптимізація алгоритму кластеризації та інтеграція методів валідації дозволяють мінімізувати кількість хибнопозитивних та хибнонегативних результатів, що є критично важливим для своєчасного реагування на кіберзагрози. Таким чином, застосунок поєднує переваги традиційних методів з можливостями сучасних технологій, забезпечуючи високий рівень точності та ефективності у виявленні аномалій, що потенційно можуть представляти небезпеку для інформаційних систем.

Іншим значним недоліком є чутливість методу k-середніх до початкової ініціалізації центроїдів. Випадковий вибір початкових точок може призвести до нестабільних результатів або застрягання алгоритму в локальних мінімумів, що знижує його загальну ефективність. Крім того, метод k-середніх передбачає, що кластери мають сферичну форму та приблизно однакові розміри, що не завжди відповідає реальним розподілам даних у мережевих середовищах. Це може

обмежити здатність алгоритму до точного виявлення аномалій у випадках, коли дані мають складну або нерівномірну структуру.

Також варто зазначити, що метод k-середніх не є оптимальним для виявлення аномалій, які є редкі та суттєво відмінними від нормальних патернів, оскільки він орієнтується на загальну схожість та може не враховувати дрібні, але критичні відхилення. Це може призвести до високого рівня хибнопозитивних результатів, коли нормальні дані класифікуються як аномалії через їхнє віддалення від центроїдів кластерів. Така ситуація особливо критична у контексті кібербезпеки, де хибнопозитивні сповіщення можуть призводити до непотрібних витрат ресурсів та зменшення довіри до системи безпеки.

Таким чином, хоча метод кластеризації методом k-середніх має значні переваги у швидкості та простоті реалізації, а також у здатності працювати з великими та високовимірними даними, його застосування для виявлення аномалій у прогнозуванні кіберзагроз обмежується необхідністю точного визначення кількості кластерів, чутливістю до початкової ініціалізації та обмеженою здатністю до роботи з складними та нерівномірно розподіленими даними. Для подолання цих обмежень може знадобитися комбінування методу k-середніх з іншими алгоритмами або використання більш сучасних методів машинного навчання.

РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорона праці

У процесі впровадження та експлуатації систем виявлення аномалій у великих даних, призначених для прогнозування кіберзагроз, особливого значення набуває дотримання вимог охорони праці. Діяльність фахівців даної галузі часто пов'язана з інтенсивним інтелектуальним навантаженням, тривалим перебуванням перед комп’ютерним обладнанням, роботою у приміщеннях з великою кількістю електронної техніки (сервери, маршрутизатори, комутатори), а також потребує забезпечення ергономічних умов та гарантування безпеки під час експлуатації обладнання що описано у ДСанПіН 5.5.6.009-98.

1. Вимоги до приміщень та робочих місць з ПК
    - Кабінети, обладнані комп’ютерною технікою, повинні розміщуватись в окремих приміщеннях з природним освітленням та організованим обміном повітря. Площа на одного учня, який працює за ПК, повинна складати не менше 6,0 кв. м, об’єм – не менше 20 куб. м.
    - Стіни, стеля, підлога та обладнання кабінетів комп’ютерної техніки повинні мати покриття із матеріалів з матовою фактурою, що зменшує відблиски.
    - Поверхня підлоги має бути антистатичною та зручною для вологого прибирання.
    - Заборонено використовувати для оздоблення інтер’єру приміщень матеріали, що виділяють у повітря шкідливі хімічні речовини. Вміст таких речовин не повинен перевищувати гранично допустимі концентрації.
2. Вимоги до освітлення приміщень та робочих місць (ДСанПіН 5.5.6.009-98 розділ 3)
    - Приміщення з ПК повинні мати як природне, так і штучне освітлення.
    - Штучне освітлення здійснюється системою загального освітлення, переважно з використанням люмінесцентних ламп.

- Освітленість на робочій поверхні столу та клавіатури повинна бути не нижче 400 лк.
- Співвідношення яскравості між робочим екраном та близьким оточенням не повинно перевищувати 5:1, а між поверхнями робочого екрану і оточенням – 10:1.

3. Вимоги до мікроклімату
    - Температура повітря повинна підтримуватись на рівні 19,5 ±0,5 °С, відносна вологість – 60 ±5 %, швидкість руху повітря – не більше 0,1 м/с.
    - У кімнатах з робочими місцями слід забезпечувати трикратний обмін повітря за годину.
4. Вимоги, що забезпечують захист від електромагнітних полів та випромінювань
    - Відеомонітори з електронно-променевими трубками можуть бути джерелами електромагнітного випромінювання, тому слід контролювати їх рівень.
    - Напруженість електромагнітного поля на відстані 0,5 м від монітора не повинна перевищувати гранично допустимих рівнів.
    - Напруженість статичного електричного поля не повинна бути більшою за 7 кВ/м при роботі до 1 години на добу та 3,5 кВ/м при тривалішому користуванні.
5. Вимоги до ергономічних параметрів та конструкції ПК
    - Візуальні ергономічні параметри відеомоніторів мають відповідати вимогам при відстані 400-800 мм від очей користувача.
    - Розміри шрифту повинні бути такими, щоб забезпечувати чітке зображення для користувача.
    - Конструкція відеомоніторів повинна дозволяти поворот і нахил екрана, забезпечуючи зручну позицію для користувача.
6. Вимоги до обладнання та організації робочого місця

- Екран відеомонітора слід розміщувати на відстані 400-800 мм від очей користувача, щоб зменшити зорове навантаження.
- Клавіатура повинна бути відокремлена від монітора та мати висоту не більше 30 мм для забезпечення оптимальної робочої пози.
- Конструкція робочого столу має дозволяти оптимальне розміщення обладнання, враховуючи кількість та розміри техніки.
- Робочий стілець (крісло) повинен підтримувати раціональну позу користувача, забезпечуючи належну підтримку спини та кінцівок.

7. Вимоги до режиму праці та відпочинку при роботі з ПК
    - Раціональний режим праці передбачає дотримання встановленої тривалості безперервної роботи з ПК та проведення регламентованих перерв.
    - Безперервна робота з екраном ПК рекомендується не довше 30 хвилин на першій годині занять та 20 хвилин на другій, що може бути прикладом і для дорослих фахівців.
    - Під час перерв слід виконувати вправи для очей, кистей, плечового поясу, а також робити короткі розминки для профілактики загальної та зорової втоми.

У сучасному світі інформаційних технологій робота за комп’ютером займає центральне місце у професійній діяльності фахівців з кібербезпеки. Безперервний моніторинг мереж, аналіз загроз, розробка та впровадження захисних рішень – усе це вимагає тривалого перебування за персональними комп’ютерами та іншим обладнанням. Тому створення комфортних, безпечних та ергономічно виважених умов праці є ключовою передумовою збереження здоров’я та підвищення професійної ефективності.

Дотримання наведених вимог – від регламентації режиму праці та відпочинку до організації мікроклімату, освітлення, ергономіки робочого місця, рівня шуму, вібрації та захисту від електромагнітних випромінювань – дозволить запобігти професійним захворюванням, захистити працівників від зорового перенапруження, хронічних болів у спині, тунельного синдрому та інших негативних наслідків тривалої роботи за комп’ютером. Правильна організація робочого простору,

використання відповідних меблів та обладнання, забезпечення належної вентиляції, температури й вологості повітря сприятимуть загальному підвищенню комфорту та продуктивності.

Особливої уваги заслуговує впровадження регулярних перерв та відповідних вправ, спрямованих на зняття напруження з очей, кистей, м'язів спини та шиї. Поєднання гнучкого графіка праці з правильною організа5цією робочого середовища не лише покращить самопочуття та здоров'я працівників, а й позитивно вплине на якість та оперативність реагування на кіберзагрози, що в кінцевому підсумку зміцнить кібербезпеку організації.

Усе це формує підґрунтя для сучасного підходу до культури безпеки праці в галузі кібербезпеки, де людина – найцінніший ресурс, а піклування про її здоров'я та комфорт стає невід'ємною частиною професійної етики та стратегічного розвитку.

4.2 Планування заходів цивільного захисту на об'єкті у випадку надзвичайних ситуацій

Радикальне розв'язання питань, пов'язаних із забезпеченням надійного захисту населення і територій України від надзвичайних ситуацій (НС), а також істотне зменшення їхніх соціально-економічних і екологічних наслідків, можливе лише за умови реалізації цілісної системи різноманітних, взаємопов'язаних заходів. Досягнення цієї мети значною мірою залежить від фахового прогнозування й аналізу можливих сценаріїв розвитку НС з боку керівників усіх рівнів (від нижчої ланки – об'єктового, до найвищого – урядового). Вони мають чітко передбачити потенційні наслідки надзвичайних подій, завчасно спланувати комплекс профілактичних і оперативних дій для їх відвернення або ліквідації, належним чином організувати управління під час виконання зазначених заходів. Не менш важливим є високий рівень готовності органів управління, сил цивільного захисту та широких верств населення до адекватних і скоординованих дій у разі виникнення кризових ситуацій.

Якість реалізації цих умінь та завдань безпосередньо пов’язана з ефективністю планування й ретельним дотриманням запланованих заходів, перш за все, на об’єктовому рівні. Саме там формується база, з якої починається превентивна діяльність, оптимальне використання ресурсів, підготовка персоналу й засобів для реагування. Без ґрунтовного планування на цьому етапі неможливо досягти узгоджених дій у масштабах регіону чи країни.

Сутність планування заходів цивільного захисту (ЦЗ) на випадок надзвичайних ситуацій полягає у всебічному вивченні та аналізі стану об’єкта, прогнозуванні складних умов, здатних виникнути під час аварій, катастроф, стихійних лих, терористичних актів та інших небезпечних подій, а також при застосуванні противником сучасних засобів ураження. Це вимагає чіткого визначення першочергових кроків, спрямованих на захист населення, підвищення стійкості функціонування промислових підприємств та інфраструктурних об’єктів як у мирний час, так і в особливий період. Потрібно встановити послідовність, терміни та дієві способи виконання намічених дій, визначити коло відповідальних виконавців, а також обсяг ресурсів (людських, матеріальних, технічних, фінансових), необхідних для втілення розроблених заходів у життя.

Головною метою планування заходів ЦЗ є створення найсприятливіших умов для організованого й своєчасного проведення захисних та рятувальних дій щодо працівників установ, підприємств і організацій, членів їхніх родин, а також для мешканців зони можливого ураження. Планування покликане забезпечити якісне виконання рятувальних та інших невідкладних робіт (РІНР) під час ліквідації наслідків НС техногенного й природного характеру, а також у періоди посиленої безпеки, як-от в умовах участі в територіальній обороні або антитерористичній діяльності. Крім того, планування повинне бути спрямоване на мінімізацію людських втрат, збереження матеріальних цінностей, забезпечення тривалої життєдіяльності галузі, регіону, підпорядкованих об’єктів господарювання та населення в умовах складних та часто непередбачуваних обставин.

Під час планування заходів ЦЗ на особливий період необхідно узгодити всі дії з мобілізаційним розгортанням народного господарства та адаптувати їх з урахуванням планів, що розробляються військовим командуванням та органами

управління ЦЗ. Це забезпечить комплексний, системний підхід та взаємну підтримку між військовими та цивільними структурами.

Планування повинно відповідати низці принципів: реальність, цілеспрямованість, конкретність, точність, гнучкість, перспективність. Воно має базуватися на ґрунтовно продуманих управлінських рішеннях, чітких розрахунках, прогнозах та враховувати унікальну специфіку діяльності кожного підприємства, установи чи організації. Заздалегідь здійснене планування дає змогу оперативно ввести плани в дію, особливо тоді, коли надзвичайна ситуація виникає раптово або коли ризики її розвитку високі.

Усі ці вимоги до планування ЦЗ важливо розглядати у взаємозв'язку. Їх комплексне застосування сприяє максимальній ефективності роботи органів управління, оперативному та раціональному застосуванню наявних сил і засобів при виконанні профілактичних та ліквідаційних заходів.

В сучасних умовах необхідно запроваджувати такі стратегії управління при надзвичайних ситуаціях на міждержавному (регіональному) рівні, які враховуватимуть міжнародну взаємодію, обмін досвідом, залучення передових технологій та дотримання міжнародних стандартів безпеки. Ці стратегії мають передбачати низку заходів, зокрема:

- Запобігання виникненню катастроф, що передбачає раціоналізацію чи навіть відмову від небезпечних технологій, ліквідацію аварійних і застарілих виробничих потужностей, модернізацію обладнання, впровадження систем моніторингу та контролю.
- Запобігання надзвичайним ситуаціям у випадках, коли неможливо цілковито усунути їх причини, за допомогою підвищення рівня безпеки, удосконалення матеріально-технічної бази, підготовки фахівців та навчання населення діям в умовах небезпеки.
- Пом'якшення наслідків НС шляхом розробки комплексів стабілізаційних, компенсаційних, реабілітаційних заходів, а також оперативної підтримки постраждалих територій та людей.

Подібна стратегія повинна спиратися на міцний правовий фундамент, дієву організаційну структуру, сучасну інформаційну базу даних, ефективні економічні механізми фінансування та інноваційні технічні рішення.

До потенційно небезпечних об'єктів належать радіаційно- (РНО), хімічно- (ХНО), біологічно-, пожежо- та вибухонебезпечні об'єкти, а також гідротехнічні споруди. Їхня діяльність пов'язана з використанням небезпечних речовин і технологій, які при аваріях або інших надзвичайних подіях можуть призвести до масштабних негативних наслідків.

План заходів щодо захисту персоналу у разі аварії на ПНО має розроблятися експлуатуючою організацією з урахуванням усіх можливих проектних та запроектних сценаріїв аварій. План повинен бути погоджений з усіма зовнішніми організаціями, які можуть залучатися до ліквідації наслідків (служби ДСНС, медичні заклади, спеціалізовані ремонтні бригади, представники органів місцевого самоврядування тощо).

Оскільки деякі ПНО можуть складатися з кількох небезпечних елементів (цехів, міні-заводів, дільниць), кожен такий елемент повинен мати власний об'єктовий план заходів, інтегрований у загальний план ПНО. Це забезпечує деталізацію та конкретність дій для кожної дільниці виробництва, підвищує точність прогнозування та дає змогу оперативно реагувати на локальні загрози.

План заходів необхідно регулярно переглядати (не рідше одного разу на п'ять років) для врахування нових нормативних вимог, технологічних змін або впровадження результатів навчань, тренувань і перевірок готовності до аварійного реагування. Зокрема, при введенні в дію нових об'єктів, виконанні реконструкцій або зміні нормативної бази, план повинен бути доповнений новими даними.

Для формування плану заходів використовуються численні вихідні дані: географічні й адміністративні характеристики розташування ПНО, відомості про компонування небезпечних установок, їх технологічні параметри, оцінка радіаційних, хімічних, біологічних та екологічних ризиків, результати розрахункового прогнозу наслідків можливих аварій, сценарії погіршення ситуації за різних метеорологічних умов та інше. Такі різнопланові дані дають змогу комплексно оцінити ситуацію та розробити гнучку систему дій.

Структура плану заходів із захисту персоналу та ліквідації наслідків аварій на ПНО передбачає перелік основних та допоміжних розділів. До основних заходів належать:

1. Матеріально-технічне забезпечення: інформація про наявні на ПНО захисні споруди, їхній клас і місткість, аварійний запас засобів індивідуального й колективного захисту, прилади дозиметричного, радіаційного, хімічного, біологічного, екологічного контролю, засоби зв'язку, інструменти, медичні препарати. Передбачаються варіанти забезпечення продовольством і водою осіб, залучених до ліквідації наслідків аварій.
2. Організація оповіщення та зв'язку: порядки оповіщення персоналу, схеми оповіщення, робота чергово-диспетчерської служби, використання наявних ліній зв'язку (основних і резервних) для швидкого інформування населення, працівників ПНО та відповідних державних органів.
3. Приведення в готовність служб і підрозділів: які підрозділи чи служби залучаються, їхній склад, функції, порядок взаємодії та готовність до виконання поставлених завдань.
4. Захист учасників рятувальних робіт: порядок допуску в зону аварії, місця зберігання засобів індивідуального захисту, методи контролю забруднення, організація санітарної обробки учасників ліквідації.
5. Розвідка радіаційної, хімічної, біологічної та екологічної обстановки: визначення груп розвідки, порядку збору, обробки та передачі інформації про стан середовища в районі аварії.
6. Надання медичної допомоги постраждалим: визначення медичних формувань, місць надання допомоги, засобів екстреного транспортування потерпілих.
7. Фізичний захист ПНО: забезпечення безпеки об'єкта, охорона периметра.
8. Забезпечення громадського порядку: співпраця з поліцією, муніципальними службами, що гарантує дотримання правил поведінки, евакуації та безпеки населення.
9. Евакуація персоналу: планування евакуаційних маршрутів, транспортного забезпечення, місць тимчасового розміщення людей.

10. Дії оперативного персоналу: розробка чітких інструкцій першочергових дій, алгоритм керування процесом ліквідації аварії.
11. Ліквідація осередку забруднення: спеціальні заходи з дезактивації, дегазації, дезінфекції, знезараження територій.
12. Протипожежні та противибухові заходи: організація гасіння пожеж, локалізація вибухів, обмеження розповсюдження небезпечних чинників.
13. Ліквідація наслідків аварійного викиду небезпечних речовин: спеціальні технології, методи, засоби, що використовуються для локалізації шкідливих речовин і зменшення шкоди довкіллю.
14. Заходи щодо зовнішніх впливів: врахування можливих природних чи техногенних чинників, що можуть ускладнити ситуацію (землетруси, повені, урагани тощо).
15. Організаційно-технічні заходи: дії щодо ознайомлення посадових осіб із планом заходів, наявність пам’яток і інструкцій на робочих місцях, проведення регулярних навчань та тренувань, відновлення аварійних запасів.

ВИСНОВКИ

У результаті проведеного дослідження були ретельно проаналізовані різноманітні методи виявлення аномалій у великих даних для прогнозування кіберзагроз, що дозволило визначити їхні переваги та обмеження у контексті сучасних вимог до кібербезпеки. Традиційні методи, зокрема кластеризація методом k-середніх, показали здатність ефективно сегментувати дані на основі їхньої подібності, що сприяло виявленню відхилень від нормальної поведінки. Однак було встановлено, що ці методи мають обмежену адаптивність до складних та високовимірних структур даних, характерних для сучасних мережевих середовищ, де аномалії можуть бути багатовимірними та нелінійними.

Сучасні методи, такі як автоенкодери та глибокі нейронні мережі, продемонстрували значне покращення у здатності до виявлення складних та невідомих аномалій. Ці підходи забезпечили більш високу точність та надійність у класифікації аномальних записів, завдяки їхній здатності навчатися складним патернам поведінки та адаптуватися до різноманітних структур даних. Автоенкодери, наприклад, дозволили ефективно реконструювати нормальні дані, що зробило можливим точне виявлення відхилень шляхом аналізу помилок реконструкції. Глибокі нейронні мережі забезпечили гнучкість у моделюванні нелінійних взаємозв'язків між ознаками, що підвищило здатність до ідентифікації аномалій у високовимірних просторах.

Порівняння традиційних та сучасних методів виявлення аномалій показало, що сучасні підходи мають перевагу у складних та динамічних середовищах, де необхідно швидко адаптуватися до нових та невідомих загроз. Проте традиційні методи залишаються корисними для швидкого попереднього аналізу та виявлення базових аномалій, що може бути важливим для оперативного реагування на загрози з мінімальними витратами ресурсів. Таким чином, інтеграція традиційних та сучасних методів дозволила досягти оптимального балансу між швидкістю обробки даних та точністю виявлення аномалій, що є критично важливим для ефективного прогнозування та реагування на кіберзагрози.

У ході дослідження було розроблено та впроваджено застосунок, який поєднує переваги традиційних та сучасних методів, забезпечуючи високу точність та ефективність у виявленні аномалій у великих даних. Завдяки впровадженню механізмів нормалізації та інженерії ознак, застосунок зміг підвищити якість вхідних даних, що сприяло покращенню результатів класифікації. Оптимізація алгоритму кластеризації методом k-середніх, а також використання сучасних методів машинного навчання, дозволили мінімізувати кількість хибнопозитивних та хибнонегативних результатів, що є суттєвим для забезпечення надійної та точній ідентифікації потенційних загроз.

Таким чином, проведене дослідження підтвердило, що сучасні методи виявлення аномалій мають значні переваги у порівнянні з традиційними підходами, особливо у складних та високовимірних даних, характерних для сучасних мережевих середовищ. Розроблений застосунок продемонстрував високу ефективність у прогнозуванні кіберзагроз, забезпечуючи користувачам інструмент, здатний оперативно та точно ідентифікувати аномалії, що можуть представляти небезпеку для інформаційних систем. Це сприяло підвищенню рівня кібербезпеки та забезпеченню більш надійного захисту від сучасних загроз, що виникають у динамічному середовищі інформаційних технологій.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Chandola, V., Banerjee, A., Kumar, V. "Anomaly detection: A survey" // ACM Computing Surveys. – 2009. – Vol. 41, No. 3. – P. 1–58.

2. Ahmed, M., Mahmood, A.N., Hu, J. "A survey of network anomaly detection techniques" // Journal of Network and Computer Applications. – 2016. – Vol. 60. – P. 19–31.

3. ZAGORODNA, N., STADNYK, M., LYPA, B., GAVRYLOV, M., & KOZAK, R. (2022). Network Attack Detection Using Machine Learning Methods. Challenges to national defence in contemporary geopolitical situation, 2022(1), 55-61.

4. Kim, G., Lee, S., Kim, S. "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection" // Expert Systems with Applications. – 2014. – Vol. 41, No. 4. – P. 1690–1700.

5. Tymoshchuk, V., Mykhailovskyi, O., Dolinskyi, A., Orlovska, A., & Tymoshchuk, D. (2024). OPTIMISING IPS RULES FOR EFFECTIVE DETECTION OF MULTI-VECTOR DDOS ATTACKS. Матеріали конференцій МЦНД, (22.11. 2024; Біла Церква, Україна), 295-300.

6. Xu, X., Wang, X. "Anomaly detection based on one-class SVM in wireless sensor networks" // Lecture Notes in Computer Science. – Berlin: Springer, 2005. – Vol. 3644. – P. 271–282.

7. Breunig, M.M. et al. "LOF: Identifying density-based local outliers" // ACM SIGMOD Record. – 2000. – Vol. 29, No. 2. – P. 93–104.

8. Lypa, B., Horyn, I., Zagorodna, N., Tymoshchuk, D., Lechachenko T., (2024). Comparison of feature extraction tools for network traffic data. CEUR Workshop Proceedings, 3896, pp. 1-11.

9. Pang, G. et al. "Deep learning for anomaly detection: A review" // ACM Computing Surveys. – 2021. – Vol. 54, No. 2. – P. 1–38.

10. Zhang, Y., Meratnia, N., Havinga, P. "Outlier detection techniques for wireless sensor networks: A survey" // IEEE Communications Surveys & Tutorials. – 2010. – Vol. 12, No. 2. – P. 159–170.

11. Tymoshchuk, D., Yasniy, O., Mytnyk, M., Zagorodna, N., Tymoshchuk, V., (2024). Detection and classification of DDoS flooding attacks by machine learning methods. CEUR Workshop Proceedings, 3842, pp. 184 - 195.

12. Eskin, E. et al. "A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data" // Applications of Data Mining in Computer Security. – Boston: Kluwer Academic Publishers, 2002. – P. 77–101.

13. Kruegel, C., Vigna, G. "Anomaly detection of web-based attacks" // Proceedings of the 10th ACM Conference on Computer and Communications Security. – Washington: ACM, 2003. – P. 251–261.

14. ТИМОЩУК, Д., ЯЦКІВ, В., ТИМОЩУК, В., & ЯЦКІВ, Н. (2024). INTERACTIVE CYBERSECURITY TRAINING SYSTEM BASED ON SIMULATION ENVIRONMENTS. MEASURING AND COMPUTING DEVICES IN TECHNOLOGICAL PROCESSES, (4), 215-220.

15. Zong, B. et al. "Deep autoencoding gaussian mixture model for unsupervised anomaly detection" // International Conference on Learning Representations (ICLR). – 2018.

16. Akoglu, L., Tong, H., Koutra, D. "Graph based anomaly detection and description: A survey" // Data Mining and Knowledge Discovery. – 2015. – Vol. 29, No. 3. – P. 626–688.

17. Hodge, V.J., Austin, J. "A survey of outlier detection methodologies" // Artificial Intelligence Review. – 2004. – Vol. 22, No. 2. – P. 85–126.

18. ТИМОЩУК, Д., & ЯЦКІВ, В. (2024). USING HYPERVISORS TO CREATE A CYBER POLYGON. MEASURING AND COMPUTING DEVICES IN TECHNOLOGICAL PROCESSES, (3), 52-56.

19. Erfani, S.M. et al. "High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning" // Pattern Recognition. – 2016. – Vol. 58. – P. 121–134.

20. Gao, J., Hu, X. "Anomaly detection using graph-based data" // IEEE Transactions on Knowledge and Data Engineering. – 2010. – Vol. 22, No. 8. – P. 1160–1171.

21. Tymoshchuk, V., Vantsa, V., Karnaukhov, A., Orlovska, A., & Tymoshchuk, D. (2024). COMPARATIVE ANALYSIS OF INTRUSION DETECTION APPROACHES BASED ON SIGNATURES AND ANOMALIES. Матеріали конференцій МЦНД, (29.11. 2024; Житомир, Україна), 328-332.

22. Ribeiro, M.T., Singh, S., Guestrin, C. "Why should I trust you? Explaining the predictions of any classifier" // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. – 2016. – P. 1135–1144.

23. Ruff, L. et al. "Deep one-class classification" // International Conference on Machine Learning (ICML). – 2018.

24. Tymoshchuk, V., Vorona, M., Dolinskyi, A., Shymanska, V., & Tymoshchuk, D. (2024). SECURITY ONION PLATFORM AS A TOOL FOR DETECTING AND ANALYSING CYBER THREATS. Collection of scientific papers «ΛΌГОΣ», (December 13, 2024; Zurich, Switzerland), 232-237.

25. Li, Y., Ma, R. "Anomaly detection in streaming data using incremental k-means clustering" // Proceedings of the 2020 IEEE International Conference on Big Data. – 2020.

26. Goldstein, M., Uchida, S. "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data" // PLoS ONE. – 2016. – Vol. 11, No. 4. – P. e0152173.

27. Тимощук, В., Долінський, А., & Тимощук, Д. (2024). СИСТЕМА ЗМЕНШЕННЯ ВПЛИВУ DOS-АТАК НА ОСНОВІ MIKROTIK. Матеріали конференцій МЦНД, (17.05. 2024; Ужгород, Україна), 198-200.

28. Xu, D. et al. "Learning to detect anomalies in industrial control systems using deep learning" // IEEE Access. – 2019. – Vol. 7. – P. 110662–110673.

29. Tymoshchuk, D., Yasniy, O., Maruschak, P., Iasnii, V., & Didych, I. (2024). Loading Frequency Classification in Shape Memory Alloys: A Machine Learning Approach. Computers, 13(12), 339.

30. Patcha, A., Park, J.-M. "An overview of anomaly detection techniques: Existing solutions and latest technological trends" // Computer Networks. – 2007. – Vol. 51, No. 12. – P. 3448–3470.

31. Gupta, M., Gao, J., Aggarwal, C.C., Han, J. "Outlier detection for temporal data: A survey" // IEEE Transactions on Knowledge and Data Engineering. – 2014. – Vol. 26, No. 9. – P. 2250–2267.

32. Бекер, І., Тимощук, В., Маслянка, Т., & Тимощук, Д. (2023). МЕТОДИКА ЗАХИСТУ ВІД ПОВІЛЬНИХ ТА ШВИДКИХ BRUTE-FORCE АТАК НА IMAP СЕРВЕР. Матеріали конференцій МНЛ, (17 листопада 2023 р., м. Львів), 275-276.

33. Akash, M., Daniel, M. "Anomaly detection in big data using machine learning techniques" // Big Data Research. – 2020. – Vol. 21. – P. 100150.

34. Chalapathy, R., Menon, A.K., Chawla, S. "Robust, deep and inductive anomaly detection" // Proceedings of the 2020 International Conference on Machine Learning (ICML). – 2020.

35. Ванца, В., Тимощук, В., Стебельський, М., & Тимощук, Д. (2023). МЕТОДИ МІНІМІЗАЦІЇ ВПЛИВУ SLOWLORIS АТАК НА ВЕБСЕРВЕР. Матеріали конференцій МЦНД, (03.11. 2023; Суми, Україна), 119-120.

36. Doshi-Velez, F., Kim, B. "Towards a rigorous science of interpretable machine learning" // arXiv preprint arXiv:1702.08608. – 2017.

37. Tymoshchuk, V., Dolinskyi, A., & Tymoshchuk, D. (2024). MESSENGER BOTS IN SMART HOMES: COGNITIVE AGENTS AT THE FOREFRONT OF THE INTEGRATION OF CYBER-PHYSICAL SYSTEMS AND THE INTERNET OF THINGS. Матеріали конференцій МЦНД, (07.06. 2024; Луцьк, Україна), 266-267.

38. Sun, Y., Kamel, M.S., Wong, A.K.C., Wang, Y. "Cost-sensitive boosting for classification of imbalanced data" // Pattern Recognition. – 2007. – Vol. 40, No. 12. – P. 3358–3378.

39. Tymoshchuk, D., & Yatskiv, V. (2024). Slowloris ddos detection and prevention in real-time. Collection of scientific papers «ΛΌГОΣ», (August 16, 2024; Oxford, UK), 171-176.

40. Akıncı, E., Erdoğan, F. "Machine learning-based anomaly detection in network traffic" // Computers & Security. – 2021. – Vol. 101. – P. 102119.

41. Liu, F.T., Ting, K.M., Zhou, Z.-H. "Isolation Forest" // Proceedings of the 2008 IEEE International Conference on Data Mining. – Pisa: IEEE, 2008. – P. 413–422.

42. Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., Bouchachia, A. "A survey on concept drift adaptation" // ACM Computing Surveys. – 2014. – Vol. 46, No. 4. – P. 1–37.

43. Mishko, O., Matiuk, D., & Derkach, M. (2024). Security of remote iot system management by integrating firewall configuration into tunneled traffic. Вісник Тернопільського національного технічного університету, 115(3), 122-129.

44. Zhou, C., Paffenroth, R.C. "Anomaly detection with robust deep autoencoders" // Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. – 2017. – P. 665–674.

45. Böhmer, M., Heindorf, S., Gottron, T., Decker, S. "Deep anomaly detection on attributed networks" // Proceedings of the 2020 SIAM International Conference on Data Mining. – 2020. – P. 720–728.

ДОДАТКИ

Додаток А Публікація

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
імені ІВАНА ПУЛЮЯ
НАУКОВЕ ТОВАРИСТВО ім. ШЕВЧЕНКА

М А Т Е Р І А Л И

XII НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

18-19 грудня 2024 року

ТЕРНОПІЛЬ

2024

УДК 001
М34

ПРОГРАМНИЙ КОМІТЕТ

Голова: Приймак Микола – професор кафедри комп'ютерних систем та мереж, д.т.н., професо**р.**

Співголови: Марущак Павло – проректор з наукової роботи, докт. техн. наук, професор.

Баран Ігор – канд. техн. наук, доцент, декан факультету ФІС.

Науковий секретар: Надія Крива– старший викладач.

Члени: Василь Кривень - завідувач кафедри математичних методів в інженерії д.ф.-м.н., професор; Галина Осухівська – завідувач кафедри комп'ютерних систем та мереж, к.т.н., доцент; Микола Карпінський - професор кафедри кібербезпеки, д.т.н., професор; Жанна Баб'як - завідувач кафедри української та іноземних мов, к.пед. н., доцент; Ярослав Литвиненко – професор кафедри комп'ютерних наук, д.т.н., професор; Михайло Петрик - завідувач кафедри програмної іженерії, д.ф.-м.н., професор; Наталія Загородна – завідувач кафедри кібербезпеки, к.т.н., доцент.

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

Голова: Скоренький Юрій Любомирович – канд. техн. наук, доцент кафедри фізики.

Члени: доцент кафедри комп'ютерних наук, к.т.н. В. Никитюк; доцент кафедри програмної інженерії, к.т.н. Д. Михалик; доцент кафедри кібербезпеки, к.т.н. М. Стадник; доцент кафедри комп'ютерних систем та мереж, к.т.н. Є. Тиш; ст. викладач Л. Джиджора.

М34 Матеріали XII науково-технічної конфції «Інформаційні моделі, системи та технології» Тернопільського національного технічного університету імені Івана Пулюя, (Тернопіль, 18–19 грудня 2024 р.). – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2024.

Адреса оргкомітету: ТНТУ ім. І. Пулюя, м. Тернопіль, вул. Руська, 56, 46001,
тел. (0352) 52-41-33, факс (0352) 25-49-83.
E-mail: conffis2024@gmail.com

Редагування, оформлення та верстка: Надія Крива

СЕКЦІЇ КОНФЕРЕНЦІЇ, ЯКІ ПРЕДСТВЛЕНІ В ЗБІРНИКУ

- Математичне моделювання
- Інформаційні системи та технології, кібербезпека
- Комп'ютерні системи та мережі
- Програмна інженерія та моделювання складних розподілених систем
- Новітні фізико-технічні та освітні технології

В збірнику надруковано тези доповідей XII науково-технічної конференції «Інформаційні моделі, системи та технології» (Тернопіль, 18–19 грудня 2024 р.) за такими науковими напрямками: математичне моделювання; інформаційні системи та технології, кібербезпека; комп'ютерні системи та мережі; програмна інженерія та моделювання складних розподілених систем; новітні фізико-технічні та освітні технології.

Розрахований на науковців, викладачів та студентів вузів.

За зміст тез та дотримання норм академічної доброчесності відповідальність несе автор.

УДК 004.056.53

Гладчук М.

Тернопільський національний технічний університет імені Івана Пулюя, Україна

ВИЯВЛЕННЯ АНОМАЛІЙ У ВЕЛИКИХ ДАНИХ ЯК ІНСТРУМЕНТ ПРОГНОЗУВАННЯ КІБЕРЗАГРОЗ

Hladchuk M.

DETECTION OF ANOMALIES IN BIG DATA AS A TOOL FOR CYBER THREAT PREDICTION

Виявлення аномалій у великих даних як інструмент прогнозування кіберзагроз

Сучасний стрімкий розвиток інформаційних технологій породжує як нові можливості, так і загрози, зокрема у сфері кібербезпеки. Основним завданням стає виявлення аномалій у великих даних, які часто сигналізують про можливі кіберзагрози. Зокрема, такі методи є ключовими для ідентифікації відхилень у поведінкових патернах мережевого трафіку або користувачів.

Аномалії у великих даних — це відхилення від нормальних патернів, які вимагають ефективних методів обробки. Використання алгоритмів машинного навчання, таких як кластеризація (K-means) та нейронні мережі, дозволяє аналізувати масиви інформації, які генеруються в реальному часі. Наприклад, кластеризація допомагає ідентифікувати точки даних, що не відповідають жодному кластеру, сигналізуючи про потенційні загрози. Методи на основі глибокого навчання забезпечують високу точність виявлення, проте є ресурсомісткими.

На практиці методи виявлення аномалій впроваджуються через платформи обробки великих даних, такі як Apache Spark. Наприклад,

ізоляційні ліси (Isolation Forest) демонструють ефективність у реальному часі для виявлення рідкісних подій у фінансових та корпоративних системах. Водночас, проблеми шуму у даних або динамічних змін у поведінкових патернах можуть знижувати продуктивність алгоритмів.

Дослідження методів аналізу великих даних дозволяє ідентифікувати найефективніші алгоритми для різних задач кібербезпеки. Результати підтверджують, що поєднання традиційних підходів із сучасними алгоритмами забезпечує гнучкість та надійність системи. Подальший розвиток технологій, таких як квантові обчислення, відкриває нові можливості для вдосконалення прогнозування кіберзагроз.

Література

1. Xu, X., Wang, X. "Anomaly detection based on one-class SVM inwireless sensor networks"
// Lecture Notes in Computer Science. – Berlin: Springer, 2005. – Vol. 3644. – P. 271–282.
2. Breunig, M.M. et al. "LOF: Identifying density-based local outliers" // ACM SIGMOD Record. – 2000. – Vol. 29, No. 2. – P. 93–104.
3. Song, X., Wu, M., Jermaine, C., Ranka, S. "Conditional anomalydetection" // IEEE Transactions on Knowledge and Data Engineering. – 2007. – Vol. 19, No. 5. – P. 631–645.
4. Pang, G. et al. "Deep learning for anomaly detection: A review" // ACM Computing Surveys. – 2021. – Vol. 54, No. 2. – P. 1–38.

Додаток Б лістинг файлу Program.py

```
import tkinter as tk
from tkinter import filedialog, messagebox
from tkinter import scrolledtext
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import threading

class CyberThreatDetector:
    def __init__(self, master):
        self.master = master
        master.title("Модель для прогнозування кіберзагроз на основі аномальних даних")
        master.geometry("600x600")

        # Створення кнопок
        self.load_button = tk.Button(master, text="Завантажити дані", command=self.load_data)
        self.load_button.pack(pady=5)

        self.preprocess_button = tk.Button(master, text="Попередня обробка даних", command=self.preprocess_data, state=tk.DISABLED)
        self.preprocess_button.pack(pady=5)

        self.train_button = tk.Button(master, text="Навчання моделі", command=self.train_model, state=tk.DISABLED)
        self.train_button.pack(pady=5)

        self.evaluate_button = tk.Button(master, text="Оцінка моделі", command=self.evaluate_model, state=tk.DISABLED)
```

```
        self.evaluate_button.pack(pady=5)

        self.plot_button = tk.Button(master, text="Візуалізація
результатів", command=self.plot_results, state=tk.DISABLED)
        self.plot_button.pack(pady=5)

        # Додано текстову зону для логів
        self.log_area = scrolledtext.ScrolledText(master,
wrap=tk.WORD, width=70, height=20, state='disabled')
        self.log_area.pack(pady=10)

        # Ініціалізація змінних
        self.data = None
        self.scaler = None
        self.k = 3  # Кількість кластерів за замовчуванням
        self.centroids = None
        self.labels = None
        self.X_train = None
        self.X_test = None
        self.y_test = None
        self.y_pred = None
        self.confusion_matrix = None

    def log_message(self, message):
        self.log_area.config(state='normal')
        self.log_area.insert(tk.END, message + "\n")
        self.log_area.see(tk.END)
        self.log_area.config(state='disabled')

    def load_data(self):
        file_path = filedialog.askopenfilename(filetypes=[("CSV
files", "*.csv")])
        if file_path:
            try:
                self.data = pd.read_csv(file_path)
```

```
                self.log_message("Дані успішно завантажено.")
                self.preprocess_button.config(state=tk.NORMAL)
            except Exception as e:
                self.log_message(f"Помилка завантаження даних:
{e}")

    def preprocess_data(self):
        if self.data is not None:
            try:
                self.log_message("Почато попередню обробку
даних...")
                # Припускаємо, що остання колонка є міткою
                if 'label' in self.data.columns:
                    X = self.data.drop('label', axis=1).values
                    y = self.data['label'].values
                    self.log_message("Виявлено колонку 'label'.
Розділення даних на ознаки та мітки.")
                else:
                    X = self.data.values
                    y = np.zeros(X.shape[0])  # Якщо мітки
немає, вважаємо всі дані нормальними
                    self.log_message("Колонка 'label' не
знайдена. Всі дані вважаються нормальними.")

                # Нормалізація даних
                self.log_message("Виконується нормалізація
даних...")
                self.scaler = self.standard_scaler(X)
                X_scaled = self.scaler(X)
                self.log_message("Нормалізація завершена.")

                # Розділення даних на тренувальні та тестові
                self.log_message("Розділення даних на
тренувальні та тестові набори...")
```

```
                self.X_train, self.X_test, self.y_train,
self.y_test = self.train_test_split(X_scaled, y, test_size=0.3,
random_state=42)
                self.log_message(f"Дані розділено:
{self.X_train.shape[0]} тренувальних та {self.X_test.shape[0]}
тестових зразків.")
                self.train_button.config(state=tk.NORMAL)
            except Exception as e:
                self.log_message(f"Помилка попередньої обробки
даних: {e}")
        else:
            self.log_message("Попередження: Спочатку завантажте
дані.")

    def standard_scaler(self, X):
        mean = X.mean(axis=0)
        std = X.std(axis=0)
        std_replaced = np.where(std == 0, 1, std)  # Замінюємо
нулі на одиниці, щоб уникнути ділення на нуль
        def scaler(X_new):
            return (X_new - mean) / std_replaced
        return scaler

    def train_test_split(self, X, y, test_size=0.3,
random_state=42):
        np.random.seed(random_state)
        indices = np.arange(X.shape[0])
        np.random.shuffle(indices)
        test_size = int(X.shape[0] * test_size)
        test_indices = indices[:test_size]
        train_indices = indices[test_size:]
        return X[train_indices], X[test_indices],
y[train_indices], y[test_indices]

    def initialize_centroids(self, X, k):
```

```
        np.random.seed(42)
        indices = np.random.choice(X.shape[0], k, replace=False)
        self.log_message(f"Ініціалізація центроїдів: Вибрано
індекси {indices}.")
        return X[indices]

    def assign_clusters(self, X, centroids):
        distances = np.linalg.norm(X[:, np.newaxis] - centroids,
axis=2)
        labels = np.argmin(distances, axis=1)
        return labels

    def update_centroids(self, X, labels, k):
        new_centroids = []
        for i in range(k):
            points = X[labels == i]
            if len(points) > 0:
                new_centroid = points.mean(axis=0)
                new_centroids.append(new_centroid)
                self.log_message(f"Оновлено центроїд для
кластера {i}: {new_centroid}")
            else:
                # Якщо кластер порожній, залишаємо старий
центроїд
                new_centroids.append(self.centroids[i])
                self.log_message(f"Кластер {i} порожній.
Центроїд не оновлено.")
        return np.array(new_centroids)

    def has_converged(self, old_centroids, new_centroids,
tol=1e-4):
        converged = np.all(np.linalg.norm(new_centroids -
old_centroids, axis=1) < tol)
        if converged:
            self.log_message("Конвергенція досягнута.")
```

```
        return converged

    def kmeans(self, X, k, max_iters=300):
        self.log_message(f"Початок алгоритму K-Means з {k}
кластерами та максимальною кількістю ітерацій {max_iters}...")
        self.centroids = self.initialize_centroids(X, k)
        for i in range(max_iters):
            self.log_message(f"Ітерація {i+1}: Призначення
кластерів...")
            labels = self.assign_clusters(X, self.centroids)
            self.log_message(f"Ітерація {i+1}: Оновлення
центроїдів...")
            new_centroids = self.update_centroids(X, labels, k)
            if self.has_converged(self.centroids,
new_centroids):
                self.log_message(f"Алгоритм завершився на
ітерації {i+1}.")
                break
            self.centroids = new_centroids
        else:
            self.log_message("Алгоритм досяг максимальної
кількості ітерацій без досягнення конвергенції.")
        return self.centroids, labels

    def train_model_thread(self):
        self.train_button.config(state=tk.DISABLED)
        self.evaluate_button.config(state=tk.DISABLED)
        self.plot_button.config(state=tk.DISABLED)
        self.k = 3  # Можна додати можливість зміни через GUI
        self.centroids, self.labels = self.kmeans(self.X_train,
self.k)
        self.log_message("Навчання моделі завершено.")
        self.evaluate_button.config(state=tk.NORMAL)

    def train_model(self):
```

```
        if self.X_train is not None:

threading.Thread(target=self.train_model_thread).start()
        else:
            self.log_message("Попередження: Спочатку виконайте
попередню обробку даних.")

    def evaluate_model(self):
        if self.centroids is not None and self.X_test is not
None:
            try:
                self.log_message("Почато оцінку моделі...")
                # Призначення кластерів для тестових даних
                labels_test = self.assign_clusters(self.X_test,
self.centroids)
                self.log_message("Призначення кластерів для
тестових даних завершено.")

                # Обчислення відстаней до центроїдів
                self.log_message("Обчислення відстаней до
центроїдів...")
                distances = np.linalg.norm(self.X_test -
self.centroids[labels_test], axis=1)

                # Визначення порогу (наприклад, 95-й перцентиль)
                threshold = np.percentile(distances, 95)
                self.log_message(f"Визначено поріг для аномалій:
{threshold:.4f}")

                # Виявлення аномалій
                self.y_pred = (distances >
threshold).astype(int)
                self.log_message(f"Виявлено
{np.sum(self.y_pred)} аномалій у тестових даних.")
```

```
                if 'label' in self.data.columns:
                    self.confusion_matrix =
self.create_confusion_matrix(self.y_test, self.y_pred)
                    report =
self.classification_report_custom(self.y_test, self.y_pred)
                    self.log_message("Створено звіт
класифікації:")
                    self.log_message(report)
                else:
                    anomalies = np.sum(self.y_pred)
                    self.log_message(f"Кількість виявлених
аномалій: {anomalies}")

                self.plot_button.config(state=tk.NORMAL)
            except Exception as e:
                self.log_message(f"Помилка оцінки моделі: {e}")
        else:
            self.log_message("Попередження: Спочатку натренуйте
модель.")

    def classification_report_custom(self, y_true, y_pred):
        # Реалізація звіту класифікації без sklearn
        TP = np.sum((y_true == 1) & (y_pred == 1))
        TN = np.sum((y_true == 0) & (y_pred == 0))
        FP = np.sum((y_true == 0) & (y_pred == 1))
        FN = np.sum((y_true == 1) & (y_pred == 0))

        precision = TP / (TP + FP) if (TP + FP) > 0 else 0
        recall = TP / (TP + FN) if (TP + FN) > 0 else 0
        f1 = 2 * precision * recall / (precision + recall) if
(precision + recall) > 0 else 0
        accuracy = (TP + TN) / (TP + TN + FP + FN) if (TP + TN +
FP + FN) > 0 else 0
```

```
        report = f"Accuracy: {accuracy:.4f}\nPrecision:
{precision:.4f}\nRecall: {recall:.4f}\nF1-score: {f1:.4f}"
        return report

    def create_confusion_matrix(self, y_true, y_pred):
        # Реалізація матриці плутанини без sklearn
        labels = [0, 1]
        cm = np.zeros((2, 2), dtype=int)
        for true, pred in zip(y_true, y_pred):
            if true in labels and pred in labels:
                cm[int(true)][int(pred)] += 1
        return cm

    def plot_results(self):
        if self.y_pred is not None:
            try:
                self.log_message("Почато візуалізацію
результатів...")
                plt.figure(figsize=(12, 6))

                if hasattr(self, 'confusion_matrix') and
self.confusion_matrix is not None:
                    if self.confusion_matrix.shape == (2, 2):
                        sns.heatmap(self.confusion_matrix,
annot=True, fmt='d', cmap='Blues', xticklabels=['Normal',
'Anomaly'], yticklabels=['Normal', 'Anomaly'])
                        plt.title('Матриця плутанини')
                        plt.xlabel('Прогноз')
                        plt.ylabel('Істинні мітки')
                        self.log_message("Побудовано матрицю
плутанини.")
                    else:
                        self.log_message("Помилка: Матриця
плутанини має неправильну форму.")
                        return
```

```
                else:
                    sns.countplot(x=self.y_pred)
                    plt.title('Розподіл прогнозів аномалій')
                    plt.xlabel('Прогноз (0 = Норма, 1 =
Аномалія)')
                    plt.ylabel('Кількість')
                    self.log_message("Побудовано діаграму
розподілу прогнозів аномалій.")

                plt.tight_layout()
                plt.show()
                self.log_message("Візуалізація завершена.")
            except Exception as e:
                self.log_message(f"Помилка візуалізації
результатів: {e}")
        else:
            self.log_message("Попередження: Спочатку оцініть
модель.")

if __name__ == "__main__":
    root = tk.Tk()
    app = CyberThreatDetector(root)
    root.mainloop()
```