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

бакалавр
(освітній рівень)

на тему: *"Розробка програмного модуля для виявлення вторгнень методами машинного навчання"*

Виконав: студент

Спеціальності:
125 «Кібербезпека»
(шифр і назва напряму підготовки, спеціальності)

Романчук В. О.
підпис (прізвище та ініціали)

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

Нормоконтроль
підпис (прізвище та ініціали)

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

Рецензент
підпис (прізвище та ініціали)

м. Тернопіль – 2022

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)
«__» _____ 2022 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 125 Кібербезпека
(шифр і назва спеціальності)

Студенту Романчуку Володимиру Олеговичу
(прізвище, ім'я, по батькові)

1. Тема роботи: Розробка програмного модуля для виявлення вторгнень методами машинного навчання

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

Затверджені наказом ректора від « 23 » березня 2022 року № 4/7-178

2. Термін подання студентом завершеної роботи 17.06.2022

3. Вихідні дані до роботи Промаркований набір даних з ознаками вторгнень

4. Зміст роботи (перелік питань, які потрібно розробити)

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи хорони праці			

7. Дата видачі завдання 23.03.2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	23.03 – 28.03	*Виконано*
2.	Підбір джерел про виявлення вторгнень метоадми машинного навчання	29.03 – 05.04	*Виконано*
3.	Опрацювання джерел в галузі дослідження	06.04 – 11.04	*Виконано*
4.	Виконання дослідження щодо виявлення вторгнень в системи	12.04 – 18.04	*Виконано*
5.	Розроблення програмного коду	19.04 – 25.04	*Виконано*
6.	Оформлення розділу “Огляд літературних джерел”	26.04 – 29.04	*Виконано*
7.	Оформлення розділу “Теоретичні основи”	02.05 – 05.05	*Виконано*
8.	Оформлення розділу “Практична частина. Виявлення вторгнень методами машинного навчання”	06.05 – 11.05	*Виконано*
9.	Виконання завдання до підрозділу “Безпека життєдіяльності, основи хорони праці”	12.05 – 16.05	*Виконано*
10.	Оформлення кваліфікаційної роботи	17.05 – 08.06	*Виконано*
11.	Нормоконтроль	09.06 – 13.06	*Виконано*
12.	Перевірка на плагіат	14.06 – 15.06	*Виконано*
13.	Попередній захист кваліфікаційної роботи	16.06 – 17.06	*Виконано*
14.	Захист кваліфікаційної роботи	23.06	

Студент _____ *Романчук В. О.*
(підпис) (прізвище та ініціали)

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

АНОТАЦІЯ

Розробка програмного модуля для виявлення вторгнень методами машинного навчання // Кваліфікаційна робота ОР «Бакалавр» // Романчук Володимир Олегович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра кібербезпеки, група СБс-42 // Тернопіль, 2022 // С. – 53, рис. – 15, табл. – 2, кресл. – 15 , додат. – 0.

Ключові слова: ВИЯВЛЕННЯ ВТОРГНЕННЯ, IDS, МАШИННЕ НАВЧАННЯ, КЛАСИФІКАЦІЯ

Проблема виявлення вторгнень вирішується в кваліфікаційній роботі завдяки методам машинного навчання, а саме класифікаційної моделі LR та RF. В роботі детально розглянуто типи систем ідентифікацій вторгнень на основі аномалій, сигнатур та мережевого трафіку. Детально представлено дослідження теми фінансових шахрайств. Наведено аналітичний огляд моделей що використовуються для виявлення вторгнення.

Імплементовано алгоритм виявлення вторгнень з метою шахрайства з використанням двох моделей МН, а саме випадкового лісу та логістичної регресії. Проведено порівняльний аналіз результатів моделей, що свідчать про доцільність застосування моделі RF для виконання поставленої задачі. Для оцінки якості роботи моделей використано показники точності та повноти, наведено матрицю похибок.

ANNOTATION

Development of a software module to detect any intrusion by machine learning methods // Qualification thesis of educational level “Bachelor” // Romanchuk Vladimir Olegovich // Ternopil Ivan Pulyuy National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Cybersecurity, СБс-42 group // Ternopil, 2022 // P. – 53, fig. – 15, table – 2 , drawing – 15 , apendix – 0.

Key words: INTRUSION DETECTION, IDS, MACHINE LEARNING, CLASSIFICATION.

The problem of intrusion detection is solved in the qualification work due to the methods of machine learning, namely the classification model LR and RF. The paper considers in detail the types of intrusion identification systems based on anomalies, signatures and network traffic. A study of the topic of financial fraud is presented in detail. An analytical review of the models used to detect intrusion is given.

An algorithm for detecting intrusions for fraud using two MN models, namely random forest and logistic regression, has been implemented. A comparative analysis of the results of the models that indicate the feasibility of using the RF model to perform the task. To assess the quality of the models, accuracy and completeness indicators were used, and an error matrix was given.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ABS	Anomaly Based System
DT	Decision Tree
IDS	Intrusion Detection System
LR	Logistic Regression
NIDS	Network Intrusion Detection System
NN	Neural Network
RF	Random Forest
SBS	Signature Based System
МН	Машинне навчання
ОС	Операційна система
ПЗ	Програмне забезпечення

ВСТУП

З високим рівнем використання Інтернету в наші дні сьогодні, безпека мережі стала ключовою основою для всіх веб-додатків, таких як онлайн-аукціони, онлайн-роздрібні продажі тощо. Для виявлення вторгнення чи спроби виявити атаки комп'ютера шляхом вивчення різної інформації із історичних файлів, що спостерігаються в мережевих процесах, є найбільш доцільним. По суті він є одним із значущих способів ефективного вирішення проблем мережевої безпеки. Вторгнення в Інтернет може поставити під загрозу безпеку даних за допомогою кількох Інтернет-мереж. У наш час швидке зростання поширення мереж, швидкість передачі даних і непередбачуване використання Інтернету додали ще більше проблем з аномаліями. Таким чином, дослідникам необхідно розробити більш надійні, ефективні та самоконтролюючі системи, які сортують негаразди та можуть виконувати роботу без взаємодії з людьми. Застосовуючи такі спроби, можна зменшити катастрофічні збої сприйнятливих систем.

Проте вторгнення відбувається не лише в комп'ютерній мережі і не лише з використанням лог-файлів. Вторгнення може відбуватись і у веб-додатках. Для прикладу купівля товарів з зламаного акаунта покупця. Великі гіганти в області ecommerce намагаються запровадити різноматні алгоритми щодо виявлення вторгнень. Забороняють проводити оплати при різній геолокації покупця, з заборонених адрес, перевіряють транзакції із використанням подвійної автентифікації.

Вторгнення може відбуватись і в додатки та програмне забезпечення. Певним чином шкідлива частина коду імплментується в безпечну і таким чином відбувається вторгнення. Для прикладу хробак Stuxnet врахувавши вразливості операційної системи Windows заблокував кілька контролерів на серверах, що управляли видобутком урану в Ірані. Звичайно наслідки такого вторгнення були критичними.

Вторгнення в соціальні мережі та "взлом" аккаунту користувача не несе аж таких критичних наслідків. Проте таким чином на картку зловмиснка друзі користувача могли надіслати певні кошти.

Метою кваліфікаційної роботи є виявлення вторгнення та його особливого виду фінансового шахрайства з використанням моделей машинного навчання. Для досягнення цієї мети необхідно вирішити ряд завдань, які представлено нижче у вигляді списку:

- Дослідити предметну область та процес успішного вторгнення.
- Проаналізувати основні системи виявлення вторгнень та їх типи.
- Дослідити широковживані алгоритми для виявлення вторгнення.
- Обґрунтувати обрання моделі класифікатора.
- Дослідити якість роботи класифікатора для виявлення фінансового шахрайств.
- З використанням тестувального набору даних протесувати якість роботи моделі класифікації.

1 ОГЛЯД ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1.1 Intrusion Detection System (IDS)

Система виявлення вторгнень (IDS) – це пристрої або програмне забезпечення, яке використовується для моніторингу мережі на будь-яку недоброзичливість дії, що перешкоджають нормальній функціональності системи, отже в результаті спричиняє певне порушення політики. У цьому розділі розглядаються деякі з системи виявлення вторгнень та програмне забезпечення, що відображає їх основну класифікацію, оцінки та вимірювання ефективності.

Виявлення вторгнень – це процес виявлення дій, які намагаються порушити загальну цілісність і конфіденційність ресурсу. Таким чином, мета виявлення вторгнень полягає в тому, щоб ідентифікувати зловмисників, які намагаються вторгнутися та порушити контроль безпеки системи. Сучасні IDS перевіряють усі ознаки даних, щоб виявити будь-які шаблони вторгнення та неправильного використання, хоча деякі ознаки можуть бути надлишковими і можуть меншою мірою сприяти процесу виявлення [1]. Сучасні системи виявлення вторгнень на основі аномалій та багато інших технічних підходів були розроблені та впроваджені для відстеження нових атак на системи. Таким чином, за допомогою цих методів можна досягти 98% швидкості виявлення при високій і 1% при низькій частоті тривоги [2].

1.2 Класифікація IDS

На думку В. Джотсна [3], існує три основних типи систем виявлення вторгнень:

- системи виявлення вторгнень на основі сигнатур (SBS);
- системи виявлення вторгнень на основі аномалій (ABS)
- системи виявлення вторгнень в мережі (NIDS).

Системи SBS, такі як Snort [3], використовують методи розпізнавання шаблонів, підтримуючи базу даних сигнатур раніше відомих атак, щоб порівняти їх із нещодавно проаналізованими даними. Тривога піднімається, коли встановлено схожість. З іншого боку, системи ABS, такі як PAYL [4], будують статистичну модель для опису нормального мережевого трафіку, де ідентифікується будь-яка ненормальна поведінка, яка відхиляється від моделі. Навпаки, системи на основі аномалій мають ту перевагу, що вони можуть виявляти атаки нульового дня [2].

1.2.1 Виявлення на основі сигнатур

З розмахом інтернет-комерції, послуг електронного бізнесу в Інтернеті, електронного банкінгу та інших високопрофільні програми, організації, що надають ці послуги, повинні підготуватися до найкращого захисту від несанкціонованого проникнення [5]. Виявлення сигнатур включає пошук мережевого трафіку на наявність серії шкідливих байтів або послідовностей пакетів. Основна перевага цієї техніки полягає в тому, що сигнатури дуже легко розробити та зрозуміти, якщо ми знаємо, яку поведінку мережі ми намагаємося визначити. Події, згенеровані IDS на основі сигнатур, можуть повідомляти причину попередження. Як можна зробити відповідність шаблону більш ефективно на сучасних системах, тому кількість енергії, необхідної для виконання цього зіставлення, мінімальна для набору правил. Цю техніку можна легко обдурити, оскільки вони засновані лише на регулярних виразах і збігу рядків. Ці механізми шукають лише рядки в пакетах, що передаються по мережевому кабелі. Підписи добре працюють лише проти фіксованого поведінкового зразка, вони не впораються з атаками, створеними людиною або хробаком з характеристиками поведінки, що самомодифікуються.

Система виявлення на основі сигнатур (також називається на основі неправильного використання), цей тип виявлення дуже ефективний проти відомих атак і залежить від отримання регулярних оновлень шаблонів [6]. Але виявлення на основі сигнатур не працює добре, коли користувач використовує передові

технології, такі як генератори NOP, кодери корисного навантаження та зашифровані канали даних. Ефективність систем на основі сигнатур значно знижується, оскільки вони повинні створювати новий підпис для кожної варіації. Оскільки сигнатури продовжують збільшуватися, продуктивність системного механізму знижується. Завдяки цьому багато механізмів виявлення вторгнень розгорнуті в системах з кількома процесорами і кількома гігабітними мережевими картами. Розробники IDS розробляють нові підписи до того, як це зробить зловмисник, щоб запобігти новим атакам на системи. Різниця в швидкості створення нових сигнатур між розробниками та зловмисниками визначає ефективність системи [2].

1.2.2 Виявлення на основі аномалій

Система виявлення вторгнень на основі аномалій — це система виявлення вторгнень для виявлення обох мереж. і комп'ютерні вторгнення та зловживання шляхом моніторингу діяльності системи та класифікації її як нормальну чи аномальну. Класифікація заснована на евристиці або правилах, а не на шаблонах або підписах, і намагається виявити будь-який тип неправильного використання, який випадає з нормальної системи операція. Це на відміну від систем на основі сигнатур, які можуть виявляти лише атаки, для яких раніше була створена сигнатура [7].

Виявлення аномалій засноване на визначенні поведінки мережі. Поведінка мережі відповідає попередньо визначеній поведінці, тоді вона приймається, або ініціюється.

Важливим етапом визначення поведінки мережі є здатність механізму IDS прорізати різні протоколи на всіх рівнях. Механізм повинен мати можливість обробляти протоколи та розуміти їх мету. Хоча цей аналіз протоколу є витратним у обчислювальному відношенні, переваги, які він генерує, наприклад збільшення набору правил, допомагає зменшити кількість помилкових спрацьовувань. Основним недоліком виявлення аномалій є визначення набору правил. Ефективність системи залежить від її якості впроваджено та протестовано на всіх

протоколах. На процес визначення правил також впливають різні протоколи, які використовуються різними постачальниками. Крім цього, користувацькі протоколи також визначають правила, що визначають складну роботу. Щоб виявлення відбувалося правильно, адміністратори повинні розробити прийнятну поведінку мережі. Але як тільки правила визначені і протокол створений, системи виявлення аномалій запрацюють добре.

1.2.3 Система виявлення вторгнень в мережу

NIDS розгорнута на стратегічній точці мережевої інфраструктури. NIDS може фіксувати та аналізувати дані для виявлення відомих атак шляхом порівняння шаблонів або підписи бази даних або виявлення незаконної діяльності шляхом сканування трафіку на предмет аномальної активності. NIDS також називають «перехоплювачами пакетів», оскільки вони захоплюють пакети, що проходять через комунікаційні середовища [6]. Мережа IDS зазвичай має два логічні компоненти: датчик і станцію керування. Датчик знаходиться в сегменті мережі, відстежуючи підозрілий трафік. Станція управління отримує сигнали тривоги від датчика і відображає їх оператору.

Датчики, як правило, є спеціалізованими системами, які існують лише для моніторингу мережі. Вони мають мережевий інтерфейс у безладному режимі, що означає, що вони отримують весь мережевий трафік, а не лише той, що призначений для їх IP-адреси, і вони захоплюють прохідний мережевий трафік для аналізу. Якщо вони виявляють щось незвичайне, вони передають це назад на станцію аналізу. Станція аналізу може відображати сигнали тривоги або робити додатковий аналіз. Фундаментальною проблемою для систем виявлення вторгнення в мережу (NIDS), які пасивно відстежують мережеве посилання, є здатність кваліфікованого зловмисника ухилятися від виявлення, використовуючи неоднозначність у потоці трафіку, як бачить NIDS [8].

1.3 Оцінка ефективності IDS

Більшість опублікованих статей, які претендують на оцінку IDS, проводяться як порівняння, а не оцінки. Оцінку слід розглядати як визначення рівня, до якого конкретна IDS відповідає заданим цілям продуктивності [9]. У системі виявлення вторгнень є класифікація мережевої діяльності як нормальна або ненормальна, мінімізуючи неправильну класифікацію [10]. В IDS існує багато проблем, які потребують вирішення, наприклад, низька здатність виявлення невідомої мережевої атаки, висока частота помилкових тривог і недостатня здатність аналізу. Як правило, виявлення вторгнень націлено на проблему класифікації, щоб відрізнити звичайну діяльність від шкідливої діяльності [11].

Згідно з публікацією NSS “Intrusion Detection Systems Group Test (2001), оцінка кожного IDS складається з двох компонентів. Перший компонент – це якісний аналіз різних особливостей і функцій кожного продукту. Коментарі та аналіз різних ознак добре продумані та неупереджені [12]. Далі група встановила, що кількісний компонент складався з чотирьох тестів NIDS в контрольованій лабораторній мережі. Ці тести зосереджено на конкретних показниках продуктивності, розпізнаванні атак, продуктивності під навантаженням, здатності виявляти прийоми ухилення та тесті роботи з визначенням стану.

Використовувані в цій оцінці показники ефективності: співвідношення виявлення атак до помилкових спрацьовувань, здатність виявляти нові та приховані атаки, порівняння систем на базі хоста та мереж для виявлення різні типи атак, здатність методів виявлення аномалій виявляти нові атаки, вдосконалення між 1998 і 1999 роками, а також здатність систем точно ідентифікувати атаки. Дослідження також намагався встановити причину, чому кожен IDS не зміг виявити атаку або генерував помилковий результат. Оцінки 1998 і 1999 років виявили низку слабких місць у існуючих IDS.

Деякі з цих питань були вирішені, а інші залишаються. У процесі тестування використовувався зразок створеного мережевого трафіку, журнали аудиту, системні журнали та інформація про файлову систему. Потім інформація була розповсюджена різним оцінювачам, які надавали відповідні дані системі

виявлення вторгнень. Це гарантувало, що кожна система отримувала ідентичні дані, водночас дозволяючи належну конфігурацію кожної системи.

Автор статті Ранума (2001) встановив, що побудувати хороші тести для IDS було важко, і щоб точно виміряти складність IDS, потрібно було докласти значних зусиль при розробці тестів, переконавшись, що тести не були упередженими або неточними. Це було викликом для IDS, особливо оскільки вони залежать від робочого середовища. Далі він дійшов висновку, що якщо проводити тести, то вони повинні базуватися на якісних і порівняльних показниках. У своєму резюме він представив деякий досвід порівняльного аналізу IDS з акцентом на погано розроблені тести та їх наслідки. І технологія продовжує просувати управління IDS системи будуть ставати все більш неефективними [13].

Автор роботи [14] запропонував використовувати схему систематичного опису для регулювання описів, які використовуються для опису функцій IDS. Це підхід повинен дозволяти оцінку IDS на основі їх описів, без необхідності експериментувати. Недоліком такого підходу є вимога точних описів. Наразі такого підходу не існує, тому реалізувати його неможливо. Такий підхід має певну перспективу на майбутнє.

1.4 Критерії вимірювання ефективності

Здатність ідентифікувати атаки. Основною вимогою до продуктивності NIDS є виявлення вторгнень. Однак визначення вторгнення наразі неясно. Зокрема, багато постачальників і дослідників, схоже, вважають будь-яку спробу розміщення шкідливого трафіку в мережі як вторгнення. Насправді більш корисна система реєструватиме шкідливий трафік і інформує оператора лише в тому випадку, якщо трафік становить серйозну загрозу безпеці цільового хоста. Snort прагне до цього напрямку, використовуючи діапазон класифікації попереджень від 1 до 10. При цьому 1 представляє лише об’єкт інтересу, а 10 – головну загрозу безпеці.

Відомі вразливості та атаки. Усі NIDS повинні бути здатні виявляти відомі вразливості. Проте дослідження вказують на це багато комерційних IDS не можуть виявити нещодавно виявлені атаки [15] [12]. З іншого боку, якщо відомо про вразливість або атаку, усі системи слід виправити або застосувати обхідні шляхи, таким чином, необхідність у NIDS для виявлення цих подій буде усунена. На жаль, реальність така що багато систем не виправляються чи не оновлюються у міру виявлення вразливостей. На це чітко вказує кількість системних компромісів, які трапляються щодня, а також той факт, що більшість проблем у списку двадцятки SANS є переважно старими добре відомими проблемами з доступними виправленнями.

Стабільність. Будь-який IDS повинен мати можливість продовжувати стабільну роботу за будь-яких обставин. Програма та операційна система повинні працювати роками без збоїв сегментації або витоку пам’яті. Важливою функцією NIDS є постійне повідомлення про ідентичні події однаковим чином. Одним із недоліків продукту, що використовує розпізнавання підпису, є можливість різних користувачів налаштовувати різні сповіщення для надання різних повідомлень. Таким чином, трафік в одній мережі може викликати різне сповіщення про той самий трафік в іншій системі того ж типу.

Простота або складність конфігурації. На жаль, зручність використання системи зазвичай обернено пропорційна гнучкості та настроюваності цієї системи. Прагнення до гнучкості, яку можна налаштувати системи, буде визначатися користувачами система, мережа, в якій вона буде працювати, і рівень функціональності, що вимагається від системи. Якщо систему буде обслуговувати адміністратор мережі, який також відповідає за стандартне керування мережею, у нього навряд чи буде час доступний для оптимізації та налаштування системи, тому зручність використання буде першочерговою мірою. З іншого боку, якщо аналітик вторгнення використовується спеціально для управління виявленням вторгнень, може знадобитися більш складна система з більшою функціональністю.

Можливі варіанти конфігурації. NIDS повинна бути оптимізована для систем у мережі. Як згадувалося раніше немає сенсу виконувати http-аналіз, якщо веб-сервер не працює в мережі, що перевіряється. Рівень трафіку в мережі також визначатиме інтенсивність проведеного аналізу. Проста система, яка підходить для окремого сегмента мережі з низьким трафіком, зможе поєднати функції датчика та аналізу в межах єдиний блок. Мережі з високим рівнем трафіку може знадобитися розділити функції датчика та аналізу між різними хостами.

Масштабованість. Більшість організацій з часом ростуть і розширюються. У міру їх розширення розвивається і допоміжна інфраструктура, включаючи комп'ютерні мережі. Будь-який IDS повинен мати можливість розширюватися разом із мережею. У міру додавання нових сегментів мережі можуть також знадобитися нові NIDS. Іншим важливим питанням буде збереження цієї інформації. Якщо невелика мережа контролюється, зберігання даних може бути можливим у плоских файлах. Однак у міру зростання кількості зібраних даних може знадобитися передати їх зберігання даних у базі даних.

Дослідження сумісності довели, що найефективніший виявлення вторгнень вимагає *кореляції інформації з низки джерел*. Це включає NIDS, HIDS, системні журнали, журнали брандмауера та будь-які інші доступні джерела інформації. На момент написання цього матеріалу Робоча група з виявлення вторгнень (IDWG) подала низку документів документи, що визначають стандарти зв'язку між IDS. Очікується, що вони будуть випущені як RFC найближчим часом. Після впровадження цих стандартів будь-які IDS, які використовують стандартні протоколи, зможуть спілкуватися з іншими IDS. Це дозволить організації впровадити ряд IDS від різних постачальників і зберігати сумісність.

Підтримка постачальників. Рівень підтримки постачальників, необхідний для впровадження буде визначатися рівнем кваліфікації персоналу, який впроваджує систему. Однак, оскільки в IT-індустрії поширені показники плинності кадрів, варто враховувати рівень підтримки, який надається постачальником.

Оновлення підписів. Будь-який IDS на основі підпису залежить від нього підписи для виявлення вторгнень. Виявилося, що здатність цих систем виявляти нові або навіть модифіковані вторгнення погана (Allen 2000). Щоб ці системи були ефективними, повинні бути доступні оновлені підписи, оскільки з'являються нові вразливості та експлойти виявлено. Багато систем на основі підписів тепер дозволяють оператору створювати власні підписи. Це може дозволити системі відстежувати нові сповіщення в разі їх виявлення, не покладаючись на оновлення постачальника. Однак відстеження вразливостей і написання підписів у міру їх виникнення є складним завданням.

2 ТЕОРЕТИЧНІ ОСНОВИ ФІНАНСОВИХ ВТОРГНЕНЬ ТА ШАХРАЙСТВ

2.1 Фінансове шахрайство як наслідок вторгнення в систему оплати

Якщо зловмисники можуть отримати продукт безкоштовно, вони можуть перепродати його за ціною нижче вашої та заробити гроші. Ця проблема стосується не лише фізичних товарів, які мають великий ринок секонд-хендів (наприклад, електроніка), але й послуг, які можна арбітражувати на конкурентному ринку (наприклад, поїздки). Якщо ваша аудиторія досить велика, знайдуться люди, які намагатимуться вкрасти ваш продукт незалежно від того, який він. Щоб зупинити цих людей, вам потрібно визначити, чи кожна покупка на вашому сайті є законною.

Переважна більшість фінансових шахрайств здійснюється з використанням викрадених кредитних карток. Шахрайство з кредитними картками старше за Інтернет, і було зроблено величезну роботу, щоб зрозуміти та запобігти цьому. Одна транзакція кредитної картки проходить через багато різних організацій під час обробки:

- Продавець.
- Платіжний процесор.
- Купецький банк.
- Карткова мережа (Visa/Mastercard/American Express/Discover), яка маршрутизує міжбанківські операції.
- Банк емітента картки.

Кожна з цих організацій має механізми виявлення шахрайства і може стягувати плату з вищезазначених організацій за шахрайські транзакції, які їм надходять. Таким чином, ціною шахрайства для продавця є не лише втрата його продукту чи послуги, а й плата, що оцінюється на різних рівнях екосистеми. У надзвичайних випадках завдана шкоди кредиту та репутації вашого бізнесу може призвести до того, що деякі банки чи мережі можуть стягнути великі штрафи або навіть відмовитися від співпраці з вами.

Звичайно, кредитні картки – не єдиний спосіб оплати, який ви можете прийняти на сайті. Прямий дебет є популярним методом в Європі, де отримати кредитні картки складніше, ніж у Сполучених Штатах. Багато сайтів приймають послуги онлайн-платежів, такі як PayPal, Apple Pay або Android Pay; у цих випадках зберігатись будуть дані облікового запису, а не дані кредитної картки та банку. Однак принципи виявлення шахрайства по суті однакові для всіх видів платежів.

Багато компаній пропонують виявлення шахрайства як послугу, проте продавець повинен вирішити яким чином і потрібно виявити шахрайство. Можливо, продавець не може надсилати конфіденційні дані третій стороні. Можливо, продукт має незвичайну схему оплати. Або, можливо, було підрахувано, що вигідніше створити систему виявлення шахрайства самостійно. У будь-якому випадку потрібно буде зібрати ознаки, які вказують на шахрайство. Деякі з них включають:

Профілі витрат клієнтів:

- Скільки стандартних відхилень від середньої покупки клієнта має дана транзакція.
- Швидкість покупок кредитною карткою.
- Поширеність поточного продукту або категорії товару в історії клієнта.
- Чи це перша покупка (наприклад, безкоштовний користувач раптом починає робити багато покупок).
- Чи є цей спосіб оплати/тип картки типовим для клієнта.
- Як недавно цей спосіб оплати був доданий до облікового запису.

Географічна залежність від часу:

- Усі кореляційні сигнали для аутентифікації (наприклад, географічне зміщення, IP/історія браузера.
- Географічна швидкість (наприклад, якщо транзакція фізичної кредитної картки була здійснена в Лондон о 20:45 та в Нью-Йорку о 21:00 того ж дня, географічна швидкість користувача є аномально висока)

Невідповідність даних:

- Чи відповідає платіжна адреса кредитної картки інформації профілю користувача на рівні міста/держави/країни
- Невідповідність адреси виставлення рахунку та адреси доставки
- Чи знаходиться банк кредитної картки в тій самій країні, що й користувач

Профіль облікового запису:

- Вік облікового запису користувача
- Репутація на основі оцінки створення облікового запису.

Статистика взаємодії з клієнтами:

- Кількість разів, коли клієнт проходить платіжний потік
- Кількість випробуваних кредитних карток
- Скільки разів пробують дану картку
- Кількість замовлень на адресу виставлення рахунку або доставки.

Якщо потрібно навчити контрольований алгоритм виявлення шахрайства, то буде необхідно промарковані дані для алгоритмів класифікації. “Золотим стандартом” для промаркованих даних є повернення платежів – покупки, визнані власником картки шахрайськими та скасовані банком. Проте зазвичай повернення платежу триває щонайменше один місяць, а в багатьох випадках це може тривати до шести місяців. У результаті відкликання платежів не можна використовувати для короткострокових показників або для розуміння того, як зловмисник адаптується до останніх змін. Таким чином, необхідно швидко перевіряти розроблені моделі шахрайства і тому необхідно використати додаткова метрика. Це може включати будь-яке з наступного:

- Повідомлення клієнтів про шахрайство.
- Відшкодування клієнту.
- Покупки, здійснені за допомогою підроблених або зламаних облікових записів.

Нарешті, під час створення системи важливо ретельно продумати, де в платіжному потоці можливо її інтегрувати. Як і з іншими проблемами

зловживання, якщо почекати довше ви можете збирати більше даних, щоб прийняти більш обґрунтоване рішення, але ризикуєте більше зашкодити. Ось деякі можливі точки інтеграції:

Попередня авторизація. Можна використати принаймні мінімальний бал, перш ніж надсилати будь-які дані компанії кредитної картки; якщо забагато карток відхилено, можна нарахувати штрафи для клієнта. Крім того, перевірки попередньої авторизації не дозволяють зловмисникам використовувати ваш сайт як тестовий майданчик, щоб визначити, яка з їхніх вкрадених карт працює.

Післяавторизація, попередня покупка. Це типове місце для проведення оцінки шахрайства, оскільки воно дозволяє ігнорувати картки, які банк відхилив, і уникати повернення платежів, враховуючи, що продавець не будете брати кошти за шахрайські покупки. Якщо на сайті налаштовано функцію аутентифікації, ви як власник сайту можете дозволити клієнту продовжити так, ніби покупка відбувається, і зібрати більше даних для подальшого оцінювання, перш ніж фактично отримати кошти.

Після покупки. Якщо існує фізичний товар, який потребує певного часу для підготовки до доставки, або якщо є віртуальна служба, яка не може завдати значної шкоди за короткий час, то можна дозволити покупці пройти і зібрати більше поведінкових сигналів, перш ніж прийняти рішення щодо операції купівлі та скасування/повернення коштів, які вважаються шахрайськими. Однак, підраховуючи бали після покупки, продавці наражаються на повернення платежів у випадку, якщо справжній власник швидко виявить шахрайство.

2.2 Ботнет активності для здійсненя вторгнень та шахрайств

У деяких випадках зловмисники можуть отримати велику цінність від однієї жертви. Банківський рахунок є очевидним прикладом, але будь-який рахунок, на якому можуть зберігатися активи, що продаються, є ціллю; до них належать облікові записи для поїздок чи житла, облікові записи з винагородами чи облікові записи для публікації оголошень. Для цих високоцінних облікових записів зловмисникам мають бажання працювати вручну, щоб уникнути виявлення. З

іншого боку, у багатьох випадках очікувана вартість однієї жертви дуже мала і насправді менша, ніж вартість людських зусиль, необхідних для доступу до облікового запису чи використання; приклади включають розсилку спаму, заповнення облікових даних та вилучення даних. У цих випадках зловмисники повинні використовувати автоматизацію, якщо вони сподіваються отримати прибуток. Навіть у більш цінних випадках людські зусилля можуть масштабуватися лише до цього часу, і автоматизація, ймовірно, забезпечить більшу віддачу від інвестицій для зловмисників.

Звідси випливає, що виявлення зловживань у багатьох випадках еквівалентно пошуку автоматичної активності (або ботів) на вашому сайті або в додатку.

Боти можуть спробувати виконати будь-які дії, зокрема такі:

Створення облікового запису. Цих ботів можна зупинити до або після створення облікового запису.

Stuffing доступів. Запуск пропущених списків пар імен користувача/пароль у вашій інфраструктурі входу, щоб спробувати скомпрометувати облікові записи. Цих ботів слід зупинити, перш ніж вони отримають доступ до облікового запису, в ідеалі без витоку інформації про те, чи дійсні облікові дані.

Scraping. Завантаження даних сайту для арбітражного чи іншого незаконного використання. Завантаження ботів має бути зупинено до того, як вони отримають будь-які дані, тому виявлення має бути синхронним і дуже швидким, щоб уникнути великої затримки для законних користувачів.

Click fraud. Збільшення кількості кліків, щоб принести додатковий дохід сайту, на якому розміщується реклама, або використання штучних кліків, щоб вичерпати рекламний бюджет конкурента. Мінімальна вимога тут полягає в тому, щоб рекламодавцям не виставлялися рахунки за шахрайські кліки; це обчислення може здійснюватися в режимі реального часу або так само повільно, як місячний або квартальний цикл розрахунків. Однак, якщо дані про останні кліки використовуються для визначення поточного рейтингу оголошення, шахрайські кліки повинні оброблятися майже в реальному часі (хоча й асинхронно).

Шахрайство з рейтингом. Штучне збільшення переглядів, оцінок “подобається” або поширення, щоб охопити ширшу аудиторію. Ці шахрайські дії повинні бути знищені незабаром після їх здійснення.

Онлайн ігри. Боти можуть імітувати діяльність, яка була б стомлюючою або дорогою для людей, наприклад, переміщення на великі відстані географічно (через підроблені сигнали GPS) або заробляння балів чи іншої ігрової валюти за допомогою повторюваних дій (наприклад, багаторазова боротьба з одним і тим же ворогом, щоб отримати величезна кількість очок досвіду).

Як і у випадку з фінансовим шахрайством, існують цілі компанії, які займаються виявленням і зупинкою діяльності ботів; тут ми даємо деякі вказівки для виявлення та припинення базових атак ботів.

Боти бувають найрізноманітніших рівнів складності та різноманітних намірів. Багато ботів навіть законні: пошукові системи, такі як Googlebot або Bingbot. Ці боти зазвичай рекламують себе як такі, а також враховують файл robots.txt, який ви розміщуєте на сайті, вказуючи шляхи, які заборонені для ботів.

“Dumbest” боти - це ті, які рекламують себе як такі в рядку свого агента користувача. Сюди входять такі інструменти, як curl і wget, фреймворки, такі як запити python, або скрипти, які прикидаються легальними сканерами, як-от Googlebot.

Після того, як було усунуто ботів, ключем до виявлення подальшої автоматизованої діяльності є агрегація – чи можете ви згрупувати разом запити, які надходять від однієї сутності. Якщо ви вимагаєте, щоб користувачі ввійшли в систему, щоб брати участь у будь-якій діяльності, яку намагаються автоматизувати боти, у вас уже буде ідентифікатор користувача, за яким ви зможете об’єднати дані. Якщо ідентифікатор користувача недоступний або якщо ви хочете об’єднати кілька користувачів, ви можете переглянути одну чи кілька IP-адрес, перенаправлень, агентів користувача, ідентифікаторів мобільних додатків чи інших параметрів.

Нехай маємо агреговані запити, які надходять від однієї особи. Як визначити, чи автоматизована діяльність? Ключова ідея тут полягає в тому, що

шаблон запитів від ботів буде відрізнятися від шаблонів, які демонструють люди. Конкретні кількісні характеристик сигналів включають наступне:

Швидкість запитів. Боти будуть робити запити швидше, ніж люди.

Регулярність запитів, виміряна розбіжністю в часі між запитами. Боти надсилатимуть запити через більш регулярні проміжки часу, ніж люди. Навіть якщо оператор бота введе випадковість у час запиту, розподіл часу між прибуттям все одно буде відрізнятися від розподілу людей.

Запитувана ентропія шляхів/сторінок. Боти будуть зосереджуватися на своїх цілях, а не переглядати різні частини сайту. Боти-скребки запитуватимуть кожну сторінку рівно один раз, тоді як реальні користувачі знову відвідуватимуть популярні сторінки.

Повторювані шаблони в запитах. Бот може неодноразово запитувати сторінку A, потім B, потім C, щоб автоматизувати потік.

Незвичайні переходи. Наприклад, бот може публікувати на кінцевій точці генерації вмісту, не завантажуючи сторінку, що містить форму подання.

Різноманітність заголовків. Боти можуть змінювати IP-адреси, користувацькі агенти, реферери та інші заголовки клієнтського сайту, щоб виглядати як люди, але розповсюдження, створене ботом, навряд чи буде відображати типовий розподіл на вашому сайті. Наприклад, користувач-бот може зробити однакову кількість запитів від кожного з кількох користувачів агенти.

Різноманітність cookies. Звичайний веб-сайт встановлює файли cookie сеансу та може встановлювати інші файли cookie залежно від потоку. Боти можуть ігнорувати деякі або всі ці запити на встановлення файлів cookie, що призведе до незвичайної різноманітності файлів cookie.

Розповсюдження кодів відповідей. Велика кількість помилок, особливо 403 або 404, може вказувати на запити ботів, сценарії яких базуються на старішій версії сайту.

Кілька систем виявлення ботів в літературі використовують деякі або всі ці сигнали. Наприклад, система PubCrawl6 включає кластеризацію та аналіз часових

рядів запитів для виявлення розподілених сканерів, тоді як алгоритм кластерує послідовності запитів на основі метрики подібності.

Корисні дані, які можливо зібрати із запиту, включають наступне:

- Можливість клієнта запускати JavaScript. Існують різні методи оцінки здібностей Java-Script, від простих перенаправлень до складних потоків виклик-відповідь; вони відрізняються введеною затримкою, а також складністю ботів, які вони можуть виявити.
- Відбиток HTML5, за допомогою якого можна визначити, чи користувальницький агент підроблений.
- Порядок, регістр та написання заголовків запитів, які можна порівняти з законними запитами від заявленого агента користувача.
- Відбиток TLS, який можна використовувати для ідентифікації конкретних клієнтів.
- Орфографія та регістр значень у полях запиту HTTP із кінцевою кількістю можливостей (наприклад, Accept-Encoding, Content-Type) — сценарії можуть містити помилку або використовувати незвичайні значення.
- Дані мобільного обладнання (наприклад, мікрофон, акселерометр), які можна використовувати для підтвердження того, що заявлений мобільний запит надходить від реального мобільного пристрою.
- IP-адреса та інформація про браузер/пристрій, яку можна використовувати для пошуку попередньо обчислених показників репутації.

Впроваджуючи виявлення на основі запитів розробники стикаються з типовим компромісом між запобіганням зловживань і збільшенням рівня безпеки для безпечних користувачів. Якщо надати кожному запиту інтерактивну CAPTCHA, можливо зупинити переважну більшість ботів, але також змусите багатьох безпечних користувачів піти. Як менш екстремальний приклад, запуск JavaScript для збору браузерної чи іншої інформації може призвести до неприйнятної затримки під час завантаження сторінки.

Позначення запитів ботів для обчислення показників або навчання контрольованих моделей є складною справою. На відміну від спаму, який може

бути наданий людині для оцінки, немає розумного способу представити окремий запит рецензенту і змусити цю людину позначити запит як бот чи ні.

Перший набір міток ботів, які ви повинні використовувати, — це боти, які рекламують себе як такі в заголовку User-Agent. Доступні як списки з відкритим кодом, так і власні списки, і характеристики цих ботів залишаться актуальними для більш складних ботів.

Якщо боти займаються автоматичним записом (наприклад, спамом, поширенням, кліками), ймовірно, власники сайтів вже отримували скарги, а дані, створені ботом, видалено. Набір даних про видалення може забезпечити хороший вихідний матеріал для навчальних моделей або кластеризації; також можна подивитися на фейкові акаунти з достатньо великою кількістю записів.

2.3 Техніки виявлення вторгнень у фінансовій сфері

Доступна значна література щодо виявлення вторгнень у фінансовій сфері через його велике значення для зменшення кіберзлочинності, а також з точки зору бізнесу. Деякі дослідники також проводили огляди літератури статей, опублікованих у 2000-х і 2010-х роках.

Щоб виявити фінансове шахрайство, дослідники зазвичай використовують методи виявлення викидів із дуже незбалансованими наборами даних. Можливі також різні види фінансових махінацій. Одна стаття пропонує чотири категорії фінансового шахрайства – шахрайство у фінансовій звітності, шахрайство з транзакціями, страхове шахрайство та кредитне шахрайство. У практичній частині роботи зосереджено на шахрайстві з транзакціями.

Для виявлення фінансового шахрайства було випробувано різноманітні методи.

У роботі [15] використовували нейронні мережі, наївні байєси та дерева рішень для виявлення шахрайства зі страхуванням автомобілів.

Стаття [16] виявляють шахрайство у фінансовій звітності в китайських компаніях, в іншій статті використано SVM, генетичне програмування, логістичну регресію та нейронні мережі.

Кластеризація на основі щільності [17] і чутливі до витрат дерева рішень використовувалися для шахрайства з кредитними картками.

В статті [18] обговорюють як контрольовані, так і неконтрольовані підходи на основі машинного навчання, що включають ANN (штучні нейронні мережі), SVM, HMM (приховані моделі Маркова) та кластеризацію.

Автори статті [19] розглядають проблему незбалансованих даних, які призводять до дуже великої кількості хибних спрацьовувань, і в деяких роботах пропонуються методи для вирішення цієї проблеми.

Автори [20] представляють систематичний огляд найбільш використовуваних методів виявлення фінансових шахрайств. 5 найкращих методів наведені в таблиці 3.1.

Таблиця 3.1 – Частота використання технік машинного навчання в проблемах виявлення вторгнень в фінансовій системі

Модель МН	**Відсоток використання, %**
LR	13 (17 статей)
Нейронні мережі	11 (15 статей)
DT	11 (15 статей)
SVM	9 (12 статей)
Наївний Баєес	6 (8 статей)

Однак існує дуже мало літератури щодо виявлення шахрайських транзакцій у мобільних платежах, ймовірно, через відносно недавні досягнення в технології.

3 РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ ДЛЯ ВИЯВЛЕННЯ ВТОРГНЕНЬ МЕТОДАМИ МАШИННОГО НАВЧАННЯ

3.1 Етапи розробки алгоритму ідентифікації аномалій мережевого трафіку

Методологія виявлення вторгнень методами МН є основою для розробки програмного модуля. Кожна фаза методології відображає етап, який необхідний для досягнення поставленої мети. На кожній фазі відбувається порівняння та покращення з метою досягнення найкращої якості класифікації вторгнення. Кожна фаза має вихід, що описують отримані на ній результати. Детальне представлення фази наведено на рис. 3.1.

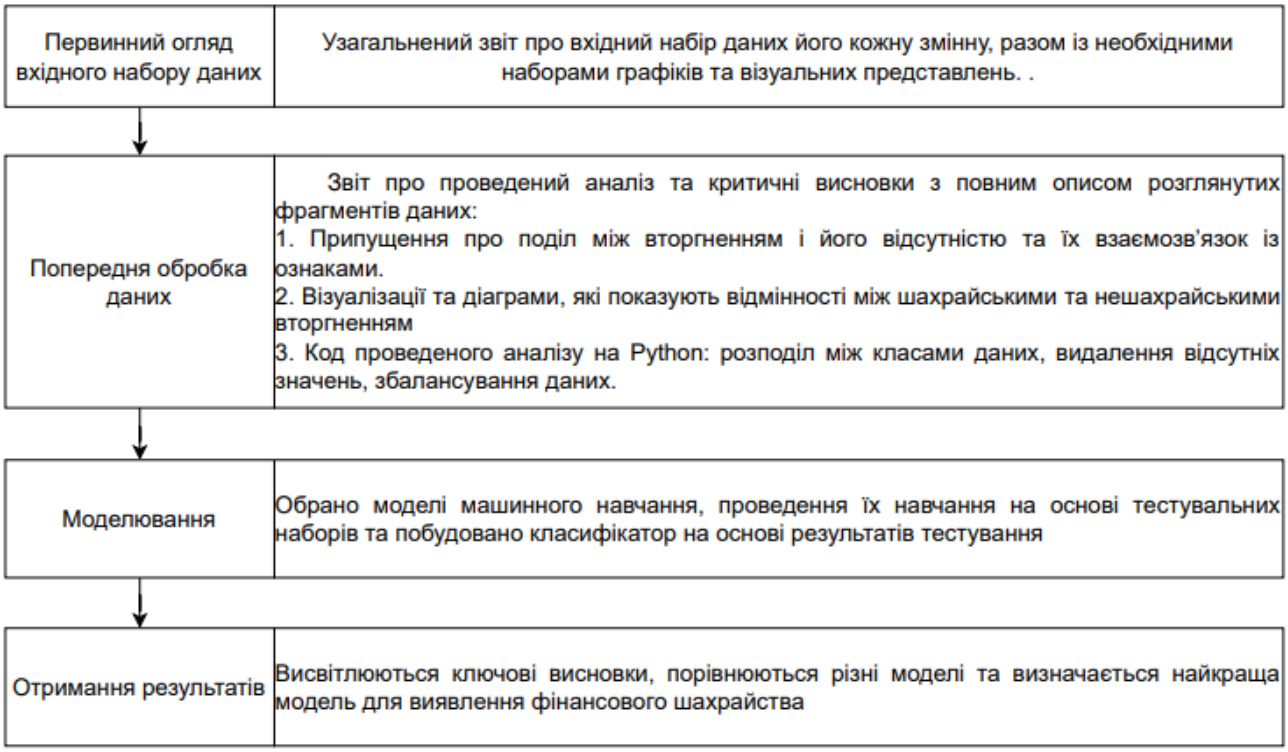

Рисунок 3.1 – Покрокові етапи методології створення програмного модуля для виявлення вторгнень

На кожному з етапів використовуються відповідні бібліотеки Python. Для читання DataFrame було використано Pandas, для роботи з масивами даних та їх

трансформації – Numpy, для виведення графіків – Seaborn, і найголовніше для задачі МН, завантаження моделей – Sklearn.

3.2 Вхідний набір даних та їх значення

Оскільки в роботі будемо досліджувати вторгнення фінансового характеру, а саме фінансові шахрайства, як особливий вид вторгнень, то необхідно мати на увазі що вони мають приватний характер і у відкритому доступі такі дані надто важко знайти. Загальнодоступних наборів даних, які можна використовувати для аналізу дуже не вистачає. Через це у проекті використовується синтетичний набір даних, загальнодоступний на Kaggle, згенерований за допомогою симулятора під назвою PaySim. Набір даних було створено з використанням агрегованих показників із приватного набору даних багатонаціональної компанії з мобільних фінансових послуг, а потім були введені шкідливі записи. (TESTIMON@ NTNU, Kaggle).

Набір даних містить 11 стовпців ознак для понад 6 мільйонів записів даних. Доступні ключові стовпці:

- “step” – Часовий крок транзакції.
- “type” – тип операцій (CASH_IN, CASH_OUT, DEBIT, PAYMENT, TRANSFER).
- “amount” – сума транзакцій.
- “nameOrig” – Ідентифікатор клієнта.
- “nameDest” – ідентифікатор одержувача.
- “oldBalanceOrg” – старий баланс клієнта.
- “newBalanceOrg” – новий баланс клієнта.
- oldBalanceOrg” – старий баланс одержувача.
- “newBalanceOrg” – новий баланс одержувача.
- “is Fraud” (аналогічні значення має isFlaggedFraud) – Наявність шахрайського вторгнення чи ні. Значення “1” вказує на шахрайство, а “0” – на відсутність шахрайства.

На рис. 3.2 представлено початкові типи кожної ознаки. Для подальшого аналізу необхідно, щоб усі стовпці в даних були відповідного типу для аналізу. Тому на початку було перевірено на необхідність у перетворенні типів.

```
step              int64
type             object
amount          float64
nameOrig         object
oldbalanceOrg   float64
newbalanceOrig  float64
nameDest         object
oldbalanceDest  float64
newbalanceDest  float64
isFraud            int64
isFlaggedFraud     int64
dtype: object
```

Рисунок 3.2 – Початкові типи даних в наборі PaySim від Kaggle

Ознака “isFraud” представляється як ціле число з типом даних int64 . Оскільки це змінна класу, було перетворено її на тип “object”.

Перш ніж приступити до аналізу, наведемо підсумкову статистику змінних. У разі числових змінних ми оцінюємо середнє значення, стандартне відхилення та діапазон значень у різних процентилях (рис. 3.3).

	step	*amount*	*oldbalanceOrg*	*newbalanceOrig*	*oldbalanceDest*	*newbalanceDest*
count	6362620	6362620	6362620	6362620	6362620	6362620
mean	243.40	179861.90	833883.10	855113.67	1100701.67	1224996.4
std	142.33	603858.23	2888242.67	2924048.50	3399180.11	3674128.9
min	1.00	0.00	0.00	0.00	0.00	0.0
25%	156.00	13389.57	0.00	0.00	0.00	0.0
50%	239.00	74871.94	14208.00	0.00	132705.66	214661.4
75%	335.00	208721.48	107315.18	144258.41	943036.71	1111909.2
max	743.00	92445516.64	59585040.37	49585040.37	356015889.35	356179278.9

Рисунок 3.3 – Статистика числових ознак в наборі PaySim від Kaggle

У випадку категоріальних змінних ми оцінюємо лише кількість унікальних категорій, найпоширенішу категорію та її частоту (рис. 3.4).

	type	nameOrig	nameDest	isFraud	isFlaggedFraud
count	6362620	6362620	6362620	6362620	6362620
unique	5	6353307	2722362	2	2
top	CASH_OUT	C1976208114	C1286084959	0	0
freq	2237500	3	113	6354407	6362604

Рисунок 3.4 – Статистика категоріальних ознак в наборі PaySim від Kaggle

Також було перевірено чи є відсутні значення в наборі даних. Лістинг 3.1 і результати його виконання вказують на загальну кількість відсутніх значень у всіх стовпцях, яка дорівнює нулю. Тобто таких значень не існує, і відповідних записів не потрібно видаляти із набору даних.

Лістинг 3.1 – Перевірка на наявність записів із відсутніми значеннями

```
print('Maximum number of missing values in any column: ' +
str(data.isnull().sum().max()))
```

```
Maximum number of missing values in any column: 0
```

Тобто таких значень не існує, і відповідних записів не потрібно видаляти із набору даних.

3.2 Попередня обробка даних

Перш ніж навчати будь-яку модель МН необхідно впевнитись у збаалансованості даних. Дисбаланс класів визначається як відсоток від загальної кількості транзакцій, представлених у стовпці isFraud. Результати представлені на рис. 3.5.

Як свідчать результати лише 0,13% (8213) транзакцій у наборі даних є шахрайськими вторгненнями, що вказує на дисбаланс високого класу в наборі даних. Це важливо, тому що якщо ми побудуємо модель машинного навчання на

цих дуже викривлених даних, нешахрайські транзакції майже повністю впливатимуть на навчання моделі, а це вплине на процес виявлення. В подальшому цю проблему дисбалансу промаркованих даних будемо вирішувати.

	Fraud Flag	Percentage_Transactions
0	Non-Fraud	99.87
1	Fraud	0.13

Рисунок 3.5 – Розподіл значень у класі “isFraud”

Дослідимо також зміну “type”, тобто розподіл між типами тразакцій. Наступний графіку відображає кількості різних транзакцій.

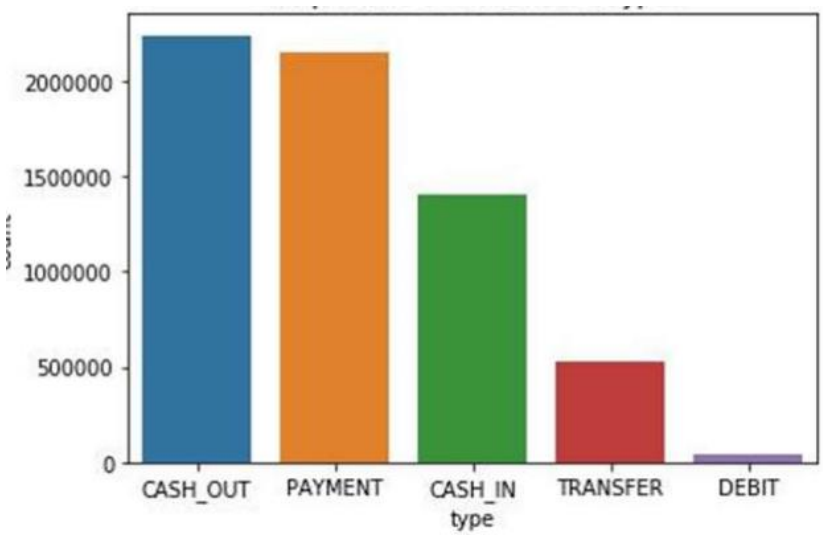

Рисунок 3.6 – Розподіл значень у класі “type”

Найпоширенішими видами транзакцій є CASH-OUT і PAYMENT. З перерахованих вище можливих видів операцій шахрайськими вторгненнями вважаються лише виведення готівки та переказ, тому і їх будемо використовувати у подальшому навчанні класифікаторів.

Знайдемо кількість зловмисних вторгнень в цих двох типах транзакцій і представимо на рис. 3.7.

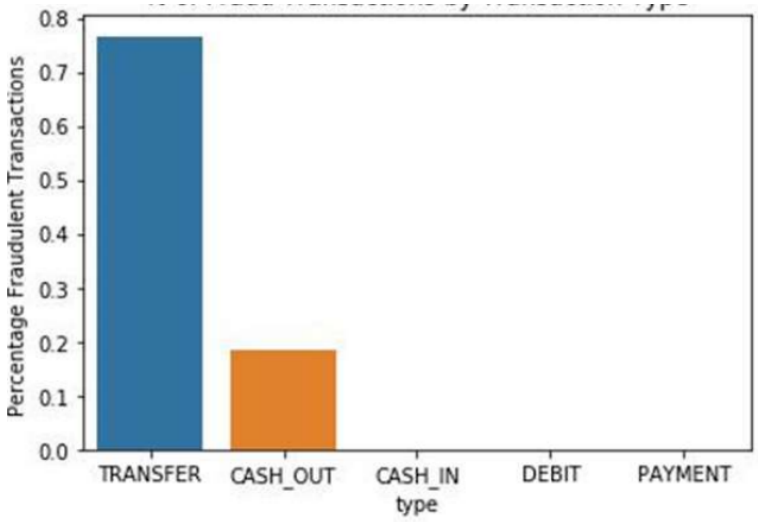

Рисунок 3.7 – Відсоткове значення вторгнень у відповідних типах транзакцій

Шахрайські вторгнення можуть бути лише в транзакціях операції є CASH-OUT і TRANSFER. Отже, має сенс зберегти лише ці два типи транзакцій у нашому наборі даних для подальшої обробки. Всі інші типи не будуть використані у роботі, видаляючи їх із набору даних (лістинг 3.2).

Лістинг 3.2 – Видалення записів з типом транзакцій відмінним від CASH-OUT і TRANSFER

```
# Retaining only CASH-OUT and TRANSFER transactions

data = data.loc[data['type'].isin(['CASH_OUT', 'TRANSFER']),:]
print('The new data now has ', len(data), ' transactions.')
```

```
The new data now has 2770393 transactions.
```

Таким чином було зменшено кількість даних з понад 6 мільйонів транзакцій до ~2,8 мільйонів транзакцій.

Наступним кроком попередньої підготоки даних є чи сума на аккаунті клієнта є позитивним числом. Є лише кілька випадків, коли сума транзакцій дорівнює 0, і дослідвиши їх бачимо, що всі вони є шахрайським вторгненям. Отже, було зроблено припущення, що якщо сума транзакції дорівнює 0, транзакція є шахрайською. Відповідно було видалено ці транзакції з даних і включаємо це припущення під час остаточної класифікації (лістинг 3.3).

Лістинг 3.3 – Видалення транзакцій, у яких сума дорівнює 0

```
# Remove 0 amount values
data = data.loc[data['amount'] > 0,:]
```

Також перевіряємо чи є неясності в балансі клієнта та отримувача. аведений нижче вихід визначає випадки, коли початковий баланс відправника або кінцевий баланс одержувача дорівнює 0. В результаті отримано дані, представлені на рис.3.8.

```
Percentage of transactions where originators initial balance is 0: 47.23%
Percentage of transactions where destination's final balance is 0: 0.6%
```

Рисунок 3.7 – Результати перевірки балансу клієнта та одержувача

Таким чином, майже в половині транзакцій початковий залишок одержувача був записаний як 0. Однак менш ніж у 1% випадків кінцевий залишок одержувача був записаний як 0. В ідеалі кінцевий залишок одержувача має дорівнювати початковому балансу одержувача плюс сума операції. Аналогічно, кінцевий залишок ініціатора має дорівнювати початковому балансу ініціатора мінус сума операції. Перевіряємо ці вимоги і чи вони точно фіксуються.

```
% transactions where originator balances are not accurately captured: 93.72
% transactions where destination balances are not accurately captured: 42.09
```

Рисунок 3.8 – Перевірка умов виконання умов транзакцій

Тому в більшості трансакцій остаточний залишок відправника не фіксується точно, а майже в половині випадків остаточний залишок одержувача не фіксується точно. Було б цікаво побачити, чи відрізняються виявлені вище розбіжності між шахрайськими та нешахрайськими операціями.

З опису даних ми знаємо, що кожен крок часу становить годину. Тому один з кроків підготовки даних є дослідження ознаки “step” та як вона корелює із шахрайськими вторгненнями. На рис. 3.9 представлено залежність кількості шахрайських та не шахрайських тразакцій за ознакою “step”.

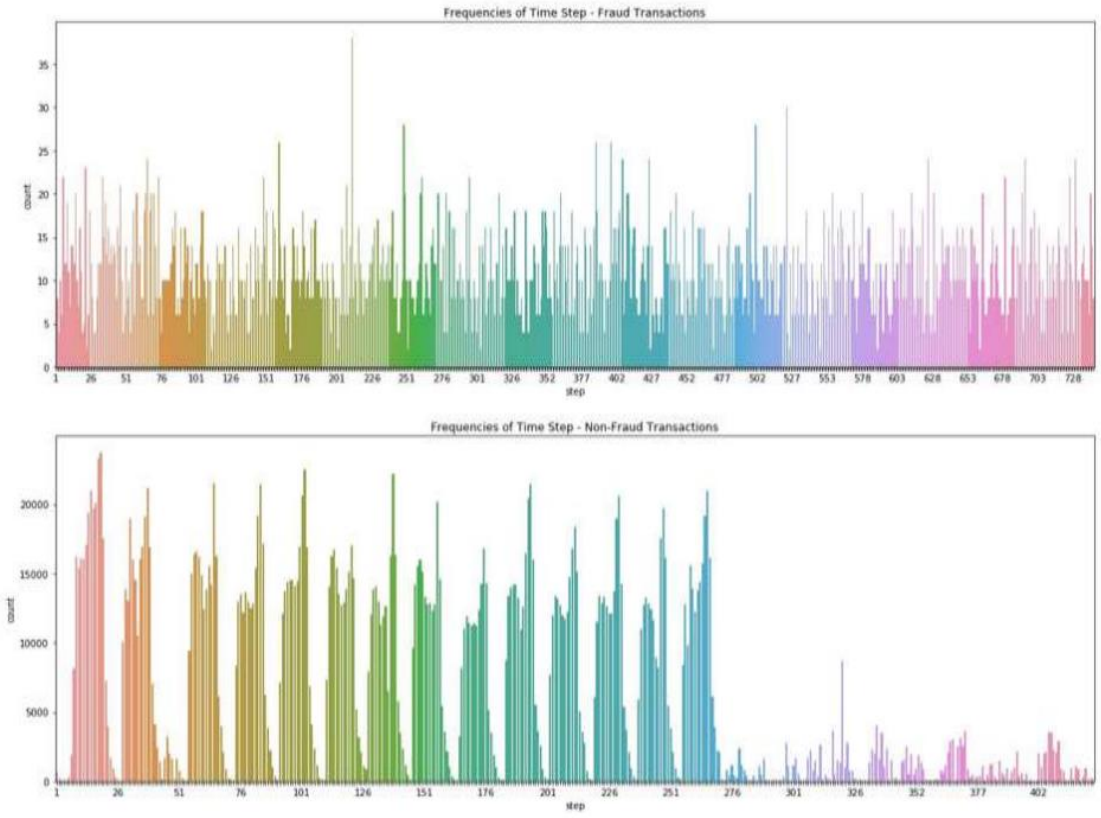

Рисунок 3.9 – Залежність кількості шахрайських та не шахрайських тразакцій за від значень ознаки “step”

На рис.3.9 видно, що шахрайські транзакції майже рівномірно розподілені за часовими кроками, тоді як нешахрайські транзакції більш зосереджені на певних часових кроках. Це може бути відмінністю між двома категоріями і може допомогти у навчанні моделей класифікації.

Також було перевірено неточність змінної балансу та порівняно між шахрайством і нешахрайством. Неточність визначається як різниця між тим, яким балансом має бути облікована сума операції, і тим, що він записується як залишок. Було обчислено неточність балансу як для відправника, так і для отримувача наступним чином:

Лістинг 3.4 – Обчислення неточності змінної балансу

```
data['origBalance_inacc'] =   (data['oldbalanceOrg'] - data['amount']) -
data['newbalanceOrig']

data['destBalance_inacc'] = (data['oldbalanceDest'] + data['amount']) -
data['newbalanceDest']
```

На рис. 3.8 та 3.9 представлено розподіл функції неточності балансу для відправника та отримувача для шахрайських та нешахрайських операцій.

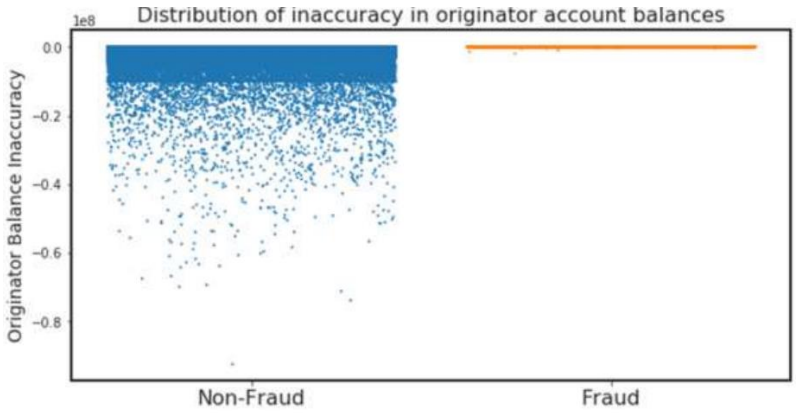

Рисунок 3.10 – Розподіл функції неточності балансу для відправника відносно класу “isFraud”

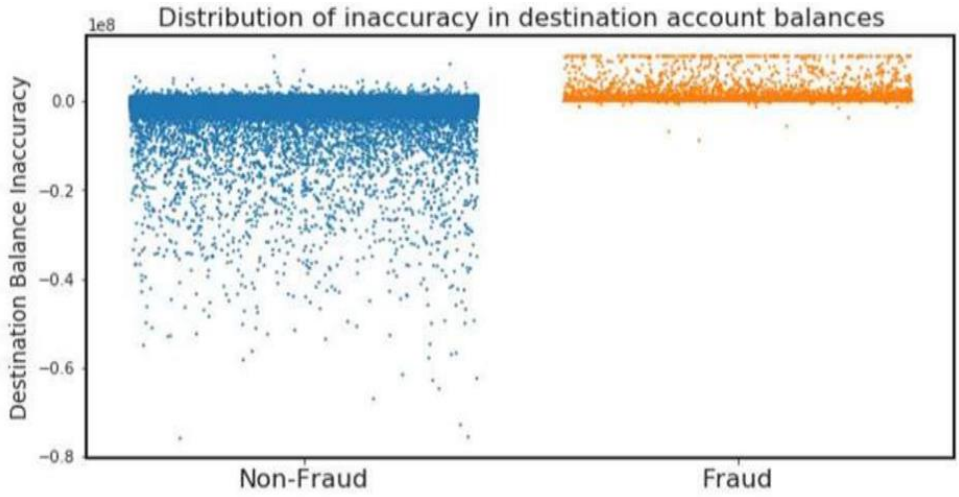

Рисунок 3.11 – Розподіл функції неточності балансу для отримувача відносно класу “isFraud”

Існують відмінності між шахрайством і нешахрайством у показниках неточності, які ми проаналізували вище. Зокрема, виявляється, що неточність балансу отримувача майже завжди є негативною для операцій без шахрайства, тоді як для шахрайських операцій вона майже завжди позитивна. Це також може бути потенційною ознакою вторгнення.

Загалом ми визначили кілька параметрів, за якими можна відрізнити шахрайські трансакції від нешахрайських. Це такі:

- часовий крок
- залишки
- неточності в балансі

Також потрібно підготовити дані для можливості подачі їх на вхід будь якого класифікатора. Тому необхідно обрати ознаки, перекодувати категоріальні зміні та стандартизумо числові. Для подачі на вхід будь якої моделі МН нам не потрібно назви клієнта та одержувача, тому вони будуть просто видаленими.

Існує одна категоріальна ознака в наборі даних – тип транзакції. Цю ознаку потрібно закодувати як двійкові змінні, а також створити фіктивні змінні. Для цього використовується наступний фрагмент коду.

Лістинг 3.5 – Кодування категорійної змінної “type”

```
# Creating dummy variables through one hot encoding for 'type' column
data = pd.get_dummies(data, columns=['type'], prefix=['type'])
```

Це створює дві двійкові фіктивні ознаки в наборі даних – type_CASH_OUT і type_TRANSFER.

Також перетворюємо всі стовпці в даних, щоб мати однаковий діапазон. Це робиться за допомогою стандартної функції масштабування, доступної в python. Для виконання цього перетворення використовується наступний фрагмент коду.

Останнім кроком попередньої обробки даних є отримання наборів даних на для навчання та тестування класифікатора. Для цього використовуємо 70% вхідних даних для навчання, а решту 30% для тестування. Наведений нижче

лістинг 3.6 використовується для створення навчальних та тестових наборів даних.

Лістинг 3.5 – Стандартизація даних

```
# Normalization of the dataset
std_scaler = StandardScaler()
data_scaled =
pd.DataFrame(std_scaler.fit_transform(data.loc[:,~data.columns.isin(['isFraud'])]))
data_scaled.columns = data.columns[:-1]
data_scaled['isFraud'] = data['isFraud']
```

Лістинг 3.6 – Формування тренувальних та тестувальних наборів даних

```
X = data_scaled.loc[:, data_scaled.columns != 'isFraud']
y = data_scaled.loc[:, data_scaled.columns == 'isFraud']

X_train_original, X_test_original, y_train_original, y_test_original =
train_test_split(X,y,test_size = 0.3, random_state = 0)

label_encoder = LabelEncoder()
y_train_original = label_encoder.fit_transform(y_train_original.values.ravel())
y_test_original = label_encoder.fit_transform(y_test_original.values.ravel())
```

Потім було перевірено, чи схожий дисбаланс класів у тренувальному та тестовому наборах даних і отримано наступні результати:

```
Class imbalance in train dataset: 0.297%
Class imbalance in test dataset 0.291%
```

Оскільки дисбаланс класів схожий, то можна приступати до навчання алгоритмів.

3.3 Моделі класифікації для виявлення шахрайських вторгнень

На основі аналітичного огляду було обрано дві моделі МН для викоаннян класифікації: логістичну регресію (LR) та випадковий ліс (RF).

Для вимірювання продуктивності моделей корисним показником є recall (повнота). Набори даних про дисбаланс високого класу зазвичай призводять до поганого значення повноти, хоча точність може бути високою. Точність також буде враховуватися, оскільки зниження точності означає, що компанія, яка намагається виявити шахрайство, понесе більше витрат на перевірку транзакцій. Однак у проблемах виявлення шахрайства точна ідентифікація шахрайських транзакцій є більш критичною, ніж неправильна класифікація законних транзакцій як шахрайських.

Наведений нижче фрагмент коду (лістинг 3.7) використовується для визначення точності двох моделей. Також задаємо параметри для виконаня перехресної перевірки, щоб переконатися, що моделі не переповнюють навчальні дані. Для цього було використано 5-кратний Stratified, оскільки потрібно переконатися, що дисбаланс класів зберігається в наборах перевірки.

Лістинг 3.7– Ініціалізація моделей LR та RF

```
scr = 'recall'
accuracy_dict = {}
model_lr = LogisticRegression()
model_rf = RandomForestClassifier()
skf = StratifiedKFold(5)
```

Для тренування моделі LR запускаємо наступний код та отримуємо усередненене значення оцінки повноти як результат навчання (лістинг 3.8).

Модель логістичної регресії за замовчуванням здатна охопити лише половину фактичних випадків шахрайства. Також представимо матриці похибок для вже тестових і тренувальних наборів даних моделі логістичної регресії, а також оцінимо точність і повноту в кожному випадку (рис.3.12).

З наведених на рис. 3.12 результатів отримуємо наступне: набори даних для навчання та тестування узгоджені, і немає перенавчання. Висока точність і низька повнота вказують на те, що виконання алгоритму над даними з дисбалансом високого класу не дасть відмінних результатів.

Лістинг 3.8– Ініціалізація моделей LR та RF

```
sc_lr = cross_val_score(model_lr, X_train_original, y_train_original, cv=skf,
scoring=scr)
```

```
Logistic Regression's average recall score across validation sets is: 50.67%
```

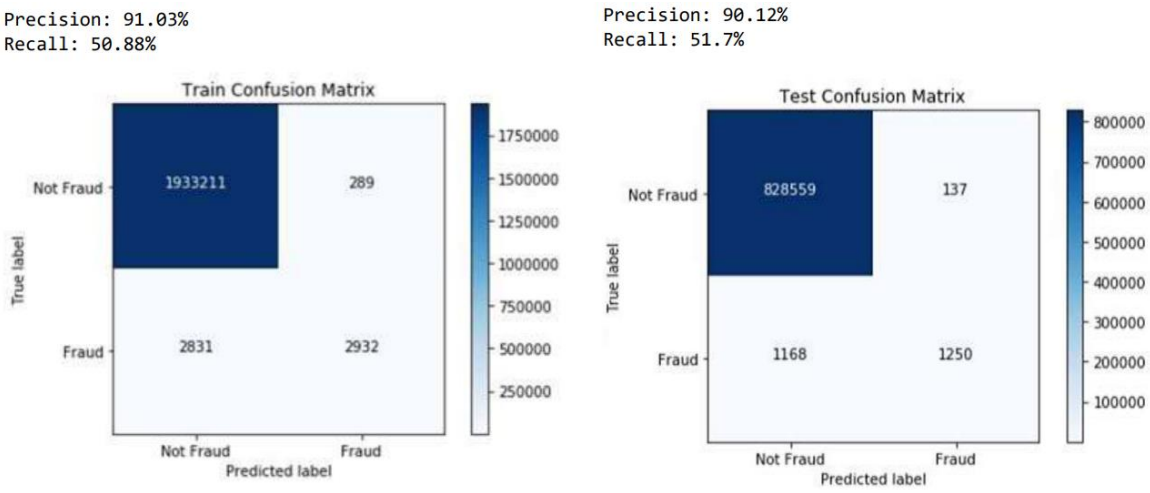

Рисунок 3.12 – Матриця похибок моделі LR для тренувального (зображення зліва) та тестового наборів (справа) відповідно

Аналогічно виконуємо і для RF. Спершу ініціюємо дану модель МН, навчаємо на тренувальних даних, а вже потім на тестових даних. Результати представлені на рис. 3.13.

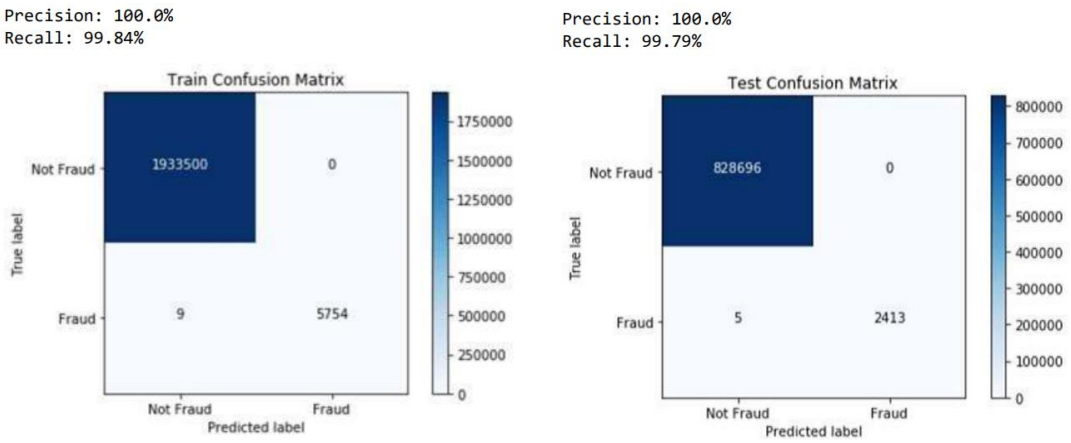

Рисунок 3.13 – Матриця похибок моделі RF для тренувального (зображення зліва) та тестового наборів (справа) відповідно

Алгоритм RF дає майже ідеальні результати. Порівнюючи оцінку повноти з LR, RF працює набагато краще у виявленні шахрайства. Крім того, продуктивність моделі RF узгоджується між наборами даних для навчання та тестування. Отже, перенавчання немає.

У наступній таблиці порівнюються результати двох моделей:

Таблиця 3.1 – Порівняння результатів навчання LR та RF

Модель	Тренувальні дані		Тестувальні дані	
	Точність, %	Повнота, %	Точність, %	Повнота, %
LR	91.03	50.88	90.12	51.7
RF	100	99.84	100	99.79

Незалежно від позитивних результатів моделі RF, було виконано спроби покращення результатів LR за допомогою налаштування параметрів і усунення дисбалансу класів завдяки перехресній перевірці та підбором даних (функційї штрафів та витрат, що є параметрами моделі LR). Результати представлено на рис. 3.14.

```
Recall of Logistic Regression for l1 penalty and C = 0.001 is: 0.0%
Recall of Logistic Regression for l1 penalty and C = 0.01 is: 22.22%
Recall of Logistic Regression for l1 penalty and C = 0.1 is: 41.02%
Recall of Logistic Regression for l1 penalty and C = 1 is: 43.83%
Recall of Logistic Regression for l1 penalty and C = 10 is: 44.15%
Recall of Logistic Regression for l1 penalty and C = 100 is: 44.16%
Recall of Logistic Regression for l1 penalty and C = 1000 is: 44.16%
Recall of Logistic Regression for l2 penalty and C = 0.001 is: 43.21%
Recall of Logistic Regression for l2 penalty and C = 0.01 is: 44.13%
Recall of Logistic Regression for l2 penalty and C = 0.1 is: 44.16%
Recall of Logistic Regression for l2 penalty and C = 1 is: 44.16%
Recall of Logistic Regression for l2 penalty and C = 10 is: 44.16%
Recall of Logistic Regression for l2 penalty and C = 100 is: 44.16%
```

Рисунок 3.14 – Налаштування параметрів моделі LR та оцінка повноти при кожному з набрів

Як бачимо найкраща модель логістичної регресії має оцінку повноти менше за 50%. Модель RF за замовчуванням працює краще, ніж модель LR.

3.4 Результати

Найкращі результати продемонструвала модель RF. Параметри цієї моделі представлені в наступному коді.

Лістинг 3.9 – Параметри моделі RF

```
<bound method BaseEstimator.get_params of RandomForestClassifier(bootstrap=Tr
ue, class_weight=None, criterion='gini', max_depth=None, max_features='auto',
max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None,
min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0,
n_estimators=10, n_jobs=None, oob_score=False, random_state=None, verbose=0,
warm_start=False)>
```

Модель використовує 10 дерев у лісі (n_estimators) і має нескінченну максимальну глибину. Позитивні результати перехресної перевірки усувають можливість перенавчання. На рис. 3.15 представлено відносну важливість ознак моделі RF.

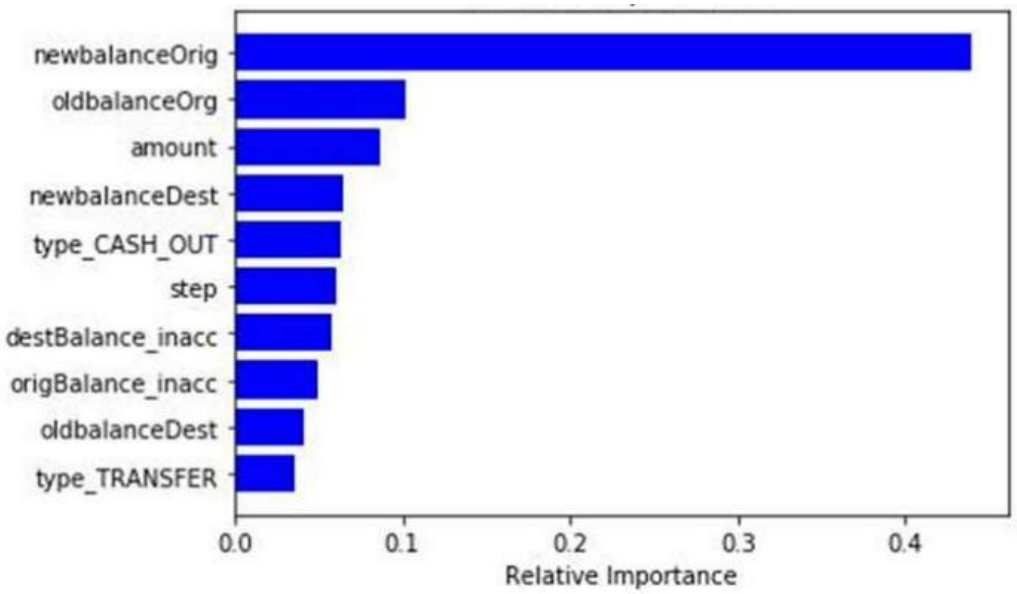

Рисунок 3.15 – Важливість ознак на якість класифікації моделі RF

Таким чином, баланс функції користувача “newbalanceOrig” має вирішальне значення для прогнозування порівняно з усіма іншими змінними.

4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ХОРОНИ ПРАЦІ

4.1 Заходи захисту обладнання від статичної електрики

Накопичення зарядів статичної електрики відбувається під час користування одягом із штучного волокна, вовни, шовку, взуттям з підошвами, що не проводять електричного струму, виконання робіт з речовинами діелектриками та шліфувальною шкуркою.

Дія статичної електрики для людини безпечна, бо сила струму дуже мала, але: розряд енергії відбувається у вигляді помірного і сильного уколу або поштовху; вплив зарядів може призвести до тяжких нещасних випадків внаслідок рефлекторного руху поблизу незахищених та рухомих частин, перебування на висоті; іскрові розряди можуть спричинити спалах або вибух горючих речовин; вибухи при перевезенні рідин у незаземлених цистернах тощо.

Заходи щодо захисту від статичної електрики:

- заземлення технологічного устаткування, трубопроводів, апаратів;
- застосування загального і місцевого зволоження повітря в небезпечних приміщеннях робочої зони, якщо це допустимо за умовами виробництва;
- використання струмопровідної підлоги, а також спецвзуття зі струмопровідною підошвою, антистатичних рукавичок;
- іонізація повітря, застосування індукційних або тканинних нейтралізаторів.

Великою небезпекою є дії грозових розрядів. Всі можливі заходи для усунення небезпеки розряду атмосферної електрики, забезпечення безпеки людей, збереження будинків, устаткування і матеріалів від руйнування, вибухів і пожеж, називається захистом від блискавки.

Атмосферна електрика, електричні заряди виникають в процесі руху крапель води в атмосфері. Процес утворення, поділу і накопичення електричних зарядів у хмарах відбувається через виникнення в них могутніх висхідних

повітряних потоків з інтенсивною конденсацією водяного пару і розбризкуванням водяних крапель. Дрібнодисперсний водяний пил, що розбризкується, заряджається негативно, а важкі краплі води – позитивно.

Прямий удар блискавки зумовлює миттєве нагрівання струмо-провідних конструкцій до температури плавлення або навіть випару, вибух чи розщеплення непровідних конструкцій, вибух будинків і будівель. При такому прямому ударі блискавки в край металевого резервуару, тепла, що виділяється, достатньо для оплавлення в місці контакту сталевого листа товщиною 4 мм.

Вторинні прояви блискавки виникають через різницю потенціалів на металевих частинах устаткування, трубопроводах і струмопроводах в результаті електромагнітної й електростатичної індукції від прямого удару блискавки.

Електромагнітна індукція. При розряді блискавки в просторі виникає магнітне поле, що змінюється з часом. Це поле індукує у металевих конструкціях, трубопроводах, електричних провідниках електрорушійну силу. У контурах із замкненою конфігурацією електричний струм викликає нагрівання конструкцій. У незамкнених металевих контурах, наприклад, у трубопровідних комунікаціях, прокладених на поверхні землі, електромагнітна індукція може викликати іскріння чи нагрівання в місцях зближення трубопроводів різних контурів.

Досвідом встановлено, що значна кількість пожеж цистерн і резервуарів з нафтопродуктами зумовлена головним чином вторинними проявами блискавки, а не прямими її ударами.

Електростатична індукція. Під грозовою хмарою у наземних об'єктах індукуються електричні заряди, рівні за величиною і протилежні за знаком зарядам хмари. Електростатичні заряди індукуються навіть на об'єктах, добре ізольованих від землі:

- на металевих дахах будинків;
- на проводах повітряних ліній зв'язку;
- на водопровідних і каналізаційних трубах;
- на електропровідниках усередині будинків та інших заземлених конструкціях.

Між об’єктом і землею потенціал тим вищий, чим вищий об'єкт. Всі металеві провідні елементи будинків і споруд заземлюють для боротьби з виникненням на спорудах і усередині будинків різниці потенціалів між устаткуванням внаслідок електростатичної індукції.

Занесення високих потенціалів у будинки і споруди відбувається через повітряні, надземні і підземні комунікації (труби, повітряні лінії зв'язку й електроенергії).

Пожежі і руйнування від розрядів атмосферної електрики відбуваються переважно від прямих ударів блискавки. Це найбільш небезпечний її прояв.

Небезпечна блискавка і для людей. Ураження можуть бути через виникнення високих напруг на окремих частинах устаткування усередині будинків й поза будинками, а також під час виникнення крокової напруги.

4.2 Значення адаптації в трудовому процесі

Сучасний розвиток технічного та технологічного стану виробництва передбачає постійну автоматизацію та оптимізацію виробничих процесів. Сьогодні, напевно, важко уявити компанію, господарська діяльність в якій здійснювалась би без використання комп’ютерної техніки. Через масовий характер робіт, що виконуються працівниками за допомогою комп’ютера, законодавством України чітко врегульовано норми та вимоги до використання комп’ютерної техніки на підприємстві, безпосередньо й охорона праці при роботі з комп’ютером.

Важливими вимоги до приміщення є наступними: приміщення, в яких планується установка та подальша робота з комп’ютером, повинні відповідати проектній документації будинку, погодженій з уповноваженими державними органами. Крім того, роботодавець повинен враховувати санітарні нормативи освітлення, вимоги до параметрів мікроклімату (температура, відносна вологість), ступеня і сили вібрації, звукового шуму і вогнестійкості приміщення, а також характеристики електромагнітного, ультрафіолетового та інфрачервоного полів.

Конкретні показники зазначених санітарних норм див. в Державних санітарих правилах і нормах роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПІН 3.3.2.007-98, затверджених Постановою Головного державного санітарного лікаря України №7 від 10 грудня 1998 року. Правила поширюються на умови й організацію праці при роботі з візуальними дисплейними терміналами (ВДТ) усіх типів вітчизняного та зарубіжного виробництва на основі електронно-променевих трубок (ЕПТ), що використовуються в електронно-обчислювальних машинах (ЕОМ) колективного використання та персональних ЕОМ (ПЕОМ). Так, наприклад, роботодавцю заборонено установлювати комп'ютери в приміщеннях, розташованих у підвалах будинків. Для уникнення можливих аварій та замикань, поряд з приміщеннями, де вестиметься робота з комп'ютером (над чи під ними), також не дозволяється проведення робіт, що потребують здійснення надмірно вологих технологічних процесів. Відповідне приміщення повинно бути укомплектоване системами центрального або індивідуального опалення, кондиціонування чи вентиляції повітря. Але при установці зазначених систем, необхідно переконатись, що батареї опалення, водопровідні труби, вентиляційні кабелі тощо, надійно сховані під захисними щитками, які перешкоджатимуть можливому потраплянню робітника під напругу.

У кожній кімнаті, де обладнуватимуться робочі місця співробітників, що працюватимуть на комп'ютері, повинні бути наявні елементи природного та штучного освітлення. При цьому, на вікнах слід встановити легко регульовані жалюзі чи штори, які дозволять працівникам коригувати рівень освітлення в приміщенні. Бажано розмістити комп'ютери в кімнаті таким чином, щоб світло потрапляло на екрани моніторів з півдня чи північного сходу. З метою досягнення максимального рівня безпечності і охорони праці при роботі з комп'ютером, виробничі приміщення необхідно обладнати аптечками першої медичної допомоги, системами автоматичної пожежної сигналізації і вогнегасниками. В приміщенні, в якому разом працюють 5 або більше комп'ютерів, на видимому

місці установлюється службовий вимикач, який у разі потреби дозволить повністю відключити електричне живлення кімнати.

Роботодавець, який використовує найману працю робітників, повинен забезпечити відповідність їхніх робочих місць комфортним та безпечним умовам. Розмір одного робочого місця має становити не менше 6 квадратних метрів. При необхідності, суміжні робочі місця співробітників, що працюють з комп'ютером, слід розділити перегородками висотою до 2 метрів. При визначенні достатнього розміру приміщення і робочого місця на одну особу необхідно додатково враховувати шафи, сейфи, тумби або інші предмети меблів чи обладнання, які знаходяться в кімнаті. На столі працівника можливо розмістити допоміжні для роботи пристрої (принтери, колонки, сканери), а також місця для зберігання документів, за умови, що це не обмежуватиме видимість екрану і не заважатиме працівнику. У разі надмірного шуму чи вібрації технічного обладнання, роботодавець повинен забезпечити працівників антивібраційними килимками. Робочий стілець співробітника має бути підйомно-поворотним, легко регульованим за висотою та забезпечувати належну підтримку та зручне положення спини і хребта особи. Щодня необхідно проводити вологе прибирання приміщення, та очищати робоче місце та безпосередньо монітор комп'ютера від запиленості.

На підприємстві забороняється: проводити ремонт та технічне обслуговування комп'ютера за робочим місцем працівника; самочинно ремонтувати або намагатись здійснити технічне налагодження комп'ютера без залучення компетентних спеціалістів; складувати на робочому місці зайві документи, деталі та предмети, що не потрібні для роботи; використовувати монітори з нечітким зображенням та монітори, у яких наявні поламки екрану; працювати з матричним принтером без антивібраційного покриття та зі знятою кришкою. Допускати до роботи осіб, які не пройшли затверджений на підприємстві курс охорони праці для роботи з комп'ютером, не дозволяється.

При прийнятті на роботу кожна особа має пройти лікарський огляд. Окрім того, при подальшій трудовій діяльності в компанії, така особа підлягає

регулярному лікарському огляду не рідше ніж раз на 2 роки. Обов'язковим є проходження таких лікарів як терапевта, невропатолога та офтальмолога. В компанії мають бути чітко встановлені перерви для відпочинку працівників (окрім обідньої), як правило, тривалістю 10-15 хвилин раз на годину або дві, в залежності від складності роботи. В будь-якому випадку, роботодавець повинен передбачити такий розпорядок роботи на підприємстві, щоб час неперервної роботи з комп'ютером був не більше ніж 4 години. Додатково, для збереження належного рівня здоров'я та професійної придатності робітників, рекомендується виділити на підприємстві окреме побутове приміщення для перепочинку працівників і зняття ними нервово-емоційного напруження, що виникає при роботі з комп'ютером.

ВИСНОВКИ

Основні теоретичні відомості щодо систем виявлення вторгнень, їх типи та класифікація представлено в першому оглядовому розділі. На основі отриманих даних отримано чітке розуміння принципу роботи системи вторгнення на основі аномалій, сигнатур та мережевого трафіку. Також наведено список показників якості для оцінки роботи системи виявлення вторгнення.

Виявлення вторгнень – це область, в якій методи машинного навчання показали значну ефективність. Перш ніж зануритися в складні алгоритми та статистичні моделі, потрібно зважити усі переваги та недоліки моделі. Також надзвичайно важливо розуміти проблему над якою працюєте та яку потрібно вирішити. Відповіддю на кращу систему виявлення вторгнень може бути не використання більш просунутого алгоритму, а створення більш повного та описового набору вхідних даних, що і було виконано в практичній частині роботи. Через великий обсяг загроз, які вплив яких потрібно зменшити, системи безпеки мають тенденцію до неконтрольованого зростання складності. При створенні або вдосконаленні систем виявлення вторгнень потрібно обирати значно простіші моделі при достатньо опрацьованому наборі даних.

В другому розділі досліджено один з методів вторгнення, а саме фінансове шахрайство, основні принципи та процес здійснення шахрайства. Основні моделі, що використовуються для виявлення шахрайства наведену в узагальненій таблиці на основі уже опрацьованих оглядів предметної області. В результаті найбільш вживаними є логістична регресія та нейронні мережі.

Було розроблено успішно систему для виявлення шахрайств з фінансовими даними. Ця структура допоможе зрозуміти нюанси виявлення шахрайства, наприклад створення похідних змінних, які можуть допомогти розділити класи, подолати дисбаланс класів і вибрати правильний алгоритм машинного навчання.

Було проведено експеримент з двома алгоритмами машинного навчання – логістичною регресією та випадковим лісом. Алгоритм Random Forest дає набагато кращі результати, ніж логістична регресія, що вказує, що алгоритми на основі дерева добре працюють для даних транзакцій з добре диференційованими

класами. Це також підкреслює корисність проведення ретельного дослідницького аналізу для детального розуміння даних перед розробкою моделей машинного навчання. Завдяки цьому дослідницькому аналізу було штучно введено кілька особливостей, які відрізняли класи краще, ніж вихідні дані.

СПИСОК ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Srilatha Chebrolua, Ajith Abrahama, Johnson P. Thomasa,. (2005). Feature deduction and ensemble design of intrusion detection systems. ELSEVIER, Pp. 295–307
2. V. Jyothsna, V. V. Rama Prasad, K. Munivara Prasad. (2011). A Review of Anomaly based Intrusion Detection Systems. International Journal of Computer Applications, pp. 26-36
3. M. Denis, C. Zena, and T. Hayajneh, "Penetration testing: Concepts, attack methods, and defense strategies," 2016 IEEE Long Island Systems, Applications and Technology Conference (LISAT), 2016
4. Shirazi, H. M. (2009). "Anomaly Intrusion Detection System using Information Theory, K-NN and KMC Algorithms. Australian Journal of Basic and Applied Sciences, pp. 2581-2597.
5. Wang. K and Stolfo.S.J. (2004). Anomalous Payload based Network Intrusion Detection. 7th Symposium on Recent Advances in Intrusion Detection (pp. pp. 203–222). USA: LNCS Springer-Verlag.
6. Brox, A. (2002, May 01st). THE CYBER SECURITY SOURCE. Retrieved December 20th, 2016, from SC Magazine US: https://www.scmagazine.com/ signature-based-or-anomaly-based-intrusion detection-the-practice-and-pitfalls/article/548733
7. Asmaa Shaker Ashoor, Prof. Sharad Gore. (2005). Importance of Intrusion Detection System (IDS). International Journal of Scientific Engineering Research, pp. 1-7
8. Anomaly-based intrusion detection system. (2016, July 16th). Retrieved December 20th, 2016, from Wikipedia Encyclopedia: https://en.wikipedia.org/ wiki/Anomalybased_intrusion_detection_system
9. Mark Handley, Vern Paxson and Christian Kreibich. (2001). Network Intrusion Detection: Evasion, Traffic Normalization, and End-to-End Protocol Semantics. Berkeley, CA 94704 USA: International Computer Science Institute.

10. Wilkison, M. (2002, June 10th). IDFAQ: How to Evaluate Network Intrusion Detection Systems? Retrieved from SANS Technology Institute: https:// www.sans.org/security-resources/idfaq/how-to evaluate-network-intrusion-detection-systems/8/10
11. Leila Mohammadpour, Mehdi Hussain, Alihossein Aryanfar, Vahid Maleki Raee and Fahad Sattar. (2015). Evaluating Performance of Intrusion Detection System using Support Vector Machines: Review. International Journal of Security and Its Applications, pp.225-234
12. Kuang, F., Xu, W., & Zhang, S. (2014). A novel hybrid KPCA and SVM with GA model for intrusion detection. Applied Soft Computing, pp. 178-184
13. The NSS Group. (2001, March 23rd). Intrusion Detection Systems Group Test (edition 2). Retrieved from NSS Group: http://www.nss.co.uk
14. TESTIMON @ NTNU, Synthetic Financial Datasets for Fraud Detection, Kaggle, retrieved from https://www.kaggle.com/ntnu-testimon/paysim1
15. Phua et.al., Minority Report in Fraud Detection: Classification of Skewed Data. ACM SIGKDD Explorations Newsletter 2004; 6: 50-59..
16. Albashrawi et.al., Detecting Financial Fraud Using Data Mining Techniques: A Decade Review from 2004 to 2015, Journal of Data Science 14(2016), 553-570
17. Dharwa et.al., A Data Mining with Hybrid Approach Based Transaction Risk Score Generation Model (TRSGM) for Fraud Detection of Online Financial Transaction, International Journal of Computer Applications 2011; 16: 18-25.
18. Sorournejad et.al., A Survey of Credit Card Fraud Detection Techniques: Data and Technique Oriented Perspective, 2016
19. Wedge et.al., Solving the False Positives Problem in Fraud Prediction Using Automated Feature Engineering, Machine Learning and Knowledge Discovery in Databases, pp 372-388, 2018
20. Albashrawi et.al., Detecting Financial Fraud Using Data Mining Techniques: A Decade Review from 2004 to 2015, Journal of Data Science 14(2016), 553-570