Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)
Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Моделі машинного навчання для
виявлення шахрайства у фінансових транзакціях
(тема)

Виконав:
здобувач четвертого року навчання,
групи ІТШІ-21-5
Олександра Шершень
(власне ім'я, прізвище)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)
Тип програми освітньо-професійна
Освітня програма Штучний інтелект
(повна назва освітньої програми)
Керівник ас. Дмитро Водяницький
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ _____ Олег ЗОЛОТУХІН
(підпис) (власне ім'я, прізвище)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Штучного інтелекту

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Штучний інтелект
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Шершень Олександрі Олегівні
(прізвище, ім'я, по батькові)

1. Тема роботи Моделі машинного навчання для виявлення шахрайства у фінансових транзакціях

затверджена наказом університету від 19 травня 20 25 р. № 378Ст
2. Термін подання студентом роботи до екзаменаційної комісії 17 червня 20 25 р.
3. Вихідні дані до роботи наукові статті та документація з області машинного навчання, роботи, присвячені методам виявлення фінансового шахрайства, документація та вихідні коди відкритих наборів даних PaySim і Credit Card Fraud Detection, а також офіційна документація бібліотек Scikit-learn, Pandas та Matplotlib.

4. Перелік питань, що потрібно опрацювати в роботі
1) Аналіз предметної галузі та постановка задачі
2) Теоретичні дослідження
3) Програмна реалізація
4) Експериментальні дослідження

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	19.05.2025	виконано
2	Визначення проблеми	20.05.2025	виконано
3	Аналіз теоретичної основи	21.05.2025	виконано
4	Підготовка до проведення експериментів	29.05.2025	виконано
5	Експериментальні дослідження моделей	01.06.2025	виконано
6	Оформлення пояснювальної записки	10.06.2025	виконано
7	Захист	17.06.2025	виконано

Дата видачі завдання 19 травня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ ас. Дмитро Водяницький
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 99 с., 19 рис., 2 табл., 1 дод., 23 джерела.

АНСАМБЛЕВИЙ МЕТОД, КЛАСОВА НЕЗБАЛАНСОВАНІСТЬ, ЛОГІСТИЧНА РЕГРЕСІЯ, МАШИННЕ НАВЧАННЯ, ТРАНЗАКЦІЇ, ФІНАНСОВЕ ШАХРАЙСТВО, RANDOM FOREST, UNDERSAMPLING.

Об’єктом дослідження є процес виявлення шахрайства у фінансових транзакціях у цифровому середовищі.

Предметом дослідження є алгоритми машинного навчання, що використовуються для класифікації транзакцій як шахрайських або легітимних.

Метою роботи є порівняльний аналіз ефективності моделей машинного навчання для виявлення фінансового шахрайства з урахуванням особливостей транзакційних даних.

У дослідженні застосовано методи навчання з учителем, зокрема логістичну регресію та Random Forest, а також метод попередньої обробки даних у вигляді undersampling для усунення класової незбалансованості.

У результаті дослідження було побудовано дві моделі виявлення шахрайських транзакцій, виконано їх порівняльну оцінку та встановлено перевагу ансамблевого підходу; новизна полягає у практичній перевірці впливу дисбалансу класів на точність моделей та доцільності використання ансамблевих рішень у фінансовому контексті.

Отримані результати рекомендовано використовувати для побудови автоматизованих систем моніторингу фінансових транзакцій у банках, фінтех-компаніях і платіжних сервісах. Розроблений підхід може бути адаптований для роботи з реальними потоками транзакцій у режимі реального часу.

ABSTRACT

Bachelor’s thesis contains: 99 pp., 19 fig., 2 tabl., 1 ann., 23 references.

CLASS IMBALANCE, ENSEMBLE METHOD, FINANCIAL FRAUD, LOGISTIC REGRESSION, MACHINE LEARNING, RANDOM FOREST, TRANSACTIONS, UNDERSAMPLING.

The object of the research is the process of fraud detection in financial transactions within a digital environment.

The subject of the research is machine learning algorithms used to classify transactions as fraudulent or legitimate.

The purpose of the work is to conduct a comparative analysis of the effectiveness of machine learning models for detecting financial fraud, taking into account the specific characteristics of transactional data.

The study applies supervised learning methods, specifically logistic regression and Random Forest, as well as a data preprocessing technique in the form of undersampling to address class imbalance.

As a result of the study, two models for fraud detection were built, their comparative evaluation was carried out, and the advantage of the ensemble approach was established; the novelty lies in the practical assessment of the impact of class imbalance on model accuracy and the appropriateness of using ensemble methods in the financial context.

The obtained results are recommended for the development of automated financial transaction monitoring systems in banks, fintech companies, and payment services. The proposed approach can be adapted for processing real-time transaction streams.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Ансамблеві методи – підхід у машинному навчанні, який об’єднує кілька моделей для підвищення точності прогнозування;

Класова незбалансованість – ситуація в наборі даних, коли кількість прикладів одного класу значно перевищує кількість прикладів іншого;

Логістична регресія – статистичний метод для бінарної класифікації, що оцінює ймовірність належності об’єкта до певного класу;

Машинне навчання – підгалузь штучного інтелекту, яка досліджує методи створення алгоритмів, здатних навчатися на даних без явного програмування;

Фінансове шахрайство – будь-яке незаконне або нечесне використання фінансових операцій для отримання вигоди;

CNN – Convolutional Neural Network – згорткова нейронна мережа;

PaySim – генератор симульованих фінансових транзакцій, заснований на реальних мобільних грошових системах;

Random Forest – ансамблева модель класифікації, що поєднує велику кількість дерев рішень;

RNN – Recurrent Neural Network – рекурентна нейронна мережа;

SMOTE – Synthetic Minority Over-sampling Technique – техніка синтетичного надзразкового розширення меншості;

SVM – Support Vector Machine – машина опорних векторів;

Undersampling – техніка балансування вибірки шляхом зменшення кількості прикладів більшості класу.

ВСТУП

Сучасний фінансовий сектор переживає період стрімкої цифрової трансформації, що охоплює всі аспекти платіжних систем, банківських послуг і електронної комерції. Широке впровадження онлайн-транзакцій, мобільного банкінгу та платіжних платформ призвело до суттєвого зростання обсягів фінансових операцій, які здійснюються щосекунди у глобальному масштабі. Разом із цим стрімким розвитком зростає і кількість шахрайських дій, які стають дедалі складнішими та менш передбачуваними. Традиційні системи безпеки, що базуються на статичних правилах або фіксованих шаблонах поведінки, вже не справляються з новими викликами, оскільки вони не здатні оперативна адаптуватися до змін у тактиках шахраїв. У такому контексті застосування інтелектуальних підходів, зокрема машинного навчання, набуває ключового значення у забезпеченні фінансової безпеки.

Актуальність тематики виявлення фінансового шахрайства обумовлена не лише фінансовими втратами, яких зазнають банки, страхові компанії та інші установи, але й високими репутаційними ризиками. Шахрайство в електронних транзакціях є не лише технічною проблемою, а й соціальним явищем, що підриває довіру користувачів до сучасних фінансових сервісів. За даними міжнародних звітів, глобальні втрати від фінансового шахрайства зросли в декілька разів за останнє десятиліття, що свідчить про необхідність впровадження нових, ефективніших рішень для виявлення таких загроз. Сучасні дослідження показують, що методи машинного навчання здатні аналізувати великі обсяги даних у реальному часі, виявляти аномалії та приховані закономірності, які важко виявити традиційними методами. Це відкриває нові можливості для створення автоматизованих систем виявлення шахрайства, які можуть оперативно реагувати на зміни в поведінці користувачів та шахрайські патерни.

З огляду на вищезазначене, дане дослідження присвячено вивченню можливостей машинного навчання для виявлення шахрайських транзакцій у фінансовій сфері, а також порівнянню ефективності окремих алгоритмів класифікації на основі імітованих транзакційних даних. Для реалізації поставленої мети було обрано два методи навчання з учителем – логістичну регресію як просту та інтерпретовану модель, а також ансамблевий метод Random Forest, що має високу точність та стійкість до шуму в даних. Набір даних, використаний у роботі, був згенерований за допомогою симулятора PaySim і містить реалістичні фінансові операції з мітками про наявність або відсутність шахрайства. Такий підхід дозволяє в умовах обмеженого доступу до реальних транзакцій провести наочний і практично орієнтований експеримент.

Практична значущість роботи полягає в тому, що запропонований підхід може бути інтегрований у реальні системи виявлення шахрайства, які застосовуються в банках, платіжних агрегаторах або інших фінансових установах. Висновки, отримані в результаті експериментів, можуть бути використані як орієнтир для розробників безпекових систем при виборі алгоритмів та налаштуванні параметрів моделей. У перспективі результати дослідження можуть слугувати базою для побудови гібридних моделей виявлення шахрайства або для створення систем, що працюють у режимі реального часу з потоковими транзакціями. Таким чином, виконання даної роботи є не лише актуальним з наукової точки зору, а й має прикладне значення для фінансової галузі, яка потребує інноваційних рішень для протидії постійно змінюваним загрозам.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Виявлення фінансового шахрайства за допомогою машинного навчання

Фінансове шахрайство є серйозною проблемою сучасного цифрового світу, оскільки зростання обсягів транзакцій і розвиток цифрових фінансових послуг створюють нові можливості для зловмисників. Шахрайські схеми охоплюють широкий спектр дій, від крадіжки особистих даних до складних схем відмивання грошей, що ставить під загрозу фінансові установи та їхніх клієнтів. Це спричиняє не лише фінансові втрати, а й репутаційні ризики, через що виникає потреба у нових підходах до захисту.

У минулому для виявлення шахрайських дій використовувалися переважно традиційні підходи, засновані на жорстко заданих правилах і процедурах, що визначали підозрілі транзакції на основі певних шаблонів. Однак цей метод має значні обмеження, оскільки не здатний гнучко адаптуватися до швидко змінюваних умов та нових типів шахрайських схем. Це спричинило необхідність автоматизованих методів і систем, здатних працювати з великими обсягами даних у режимі реального часу та адаптуватися до нових загроз.

Машинне навчання пропонує новий підхід до цієї проблеми, надаючи інструменти для автоматичного виявлення аномальних патернів у даних без постійного ручного втручання. Завдяки здатності до адаптації та аналізу великих обсягів даних, методи машинного навчання можуть значно підвищити точність і надійність моніторингу транзакцій.

Цей підрозділ надає огляд проблеми фінансового шахрайства, обґрунтовує потребу в автоматизації процесів виявлення шахрайства та розглядає потенціал машинного навчання як ефективного інструменту для цього завдання.

1.1.1 Огляд проблеми шахрайства у фінансових транзакціях

Фінансове шахрайство є серйозною загрозою для сучасних фінансових установ і користувачів їхніх послуг, особливо в умовах стрімкого розвитку цифрових технологій та онлайн-банкінгу. З кожним роком обсяг фінансових транзакцій зростає, що створює сприятливе середовище для зловмисників, які використовують різні методи для незаконного заволодіння коштами.

Фінансове шахрайство у транзакціях є серйозною загрозою для сучасних фінансових установ, і ця проблема набуває все більшого масштабу. Глобальні втрати від шахрайства (рисунок 1.1) зросли з 7,8 мільярдів доларів у 2010 році до понад 31 мільярда доларів у 2020 році. Це зростання свідчить про посилення активності шахраїв, що створює значні фінансові та репутаційні ризики для банків та платіжних систем.

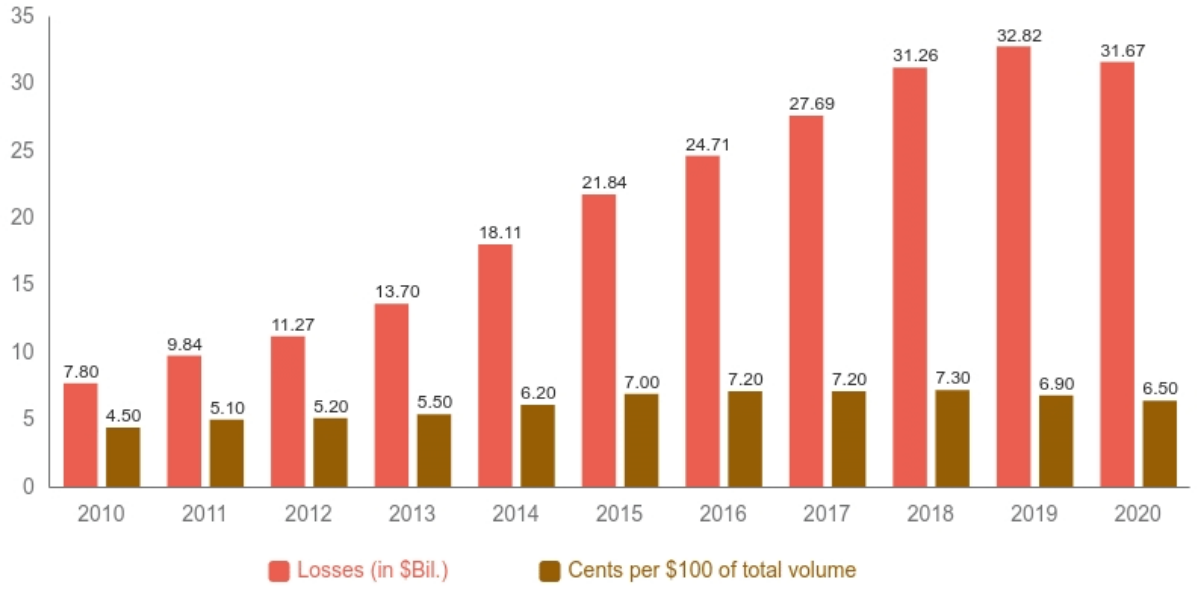

Рисунок 1.1 – Глобальні втрати у мільярдах від фінансового шахрайства з 2010 по 2020 рік

Шахрайство у фінансових транзакціях призводить до значних фінансових втрат для користувачів і фінансових установ, які змушені покривати ці збитки та інвестувати в системи безпеки. Крім того, такі інциденти завдають шкоди репутації фінансових організацій, знижуючи довіру клієнтів і ставлячи під загрозу їхню конкурентоспроможність. У сучасних умовах високої конкуренції репутаційні втрати можуть мати довготривалі наслідки для банків і фінансових компаній.

Проблема шахрайства є складною через швидку еволюцію шахрайських схем і зростання обсягів транзакцій. Традиційні методи, засновані на правилах, стають менш ефективними, оскільки не можуть швидко адаптуватися до нових загроз. Сучасне шахрайство також характеризується організованістю: шахраї діють групами, використовуючи автоматизовані інструменти для проведення численних транзакцій, що ускладнює роботу систем безпеки.

Таким чином, проблема фінансового шахрайства залишається актуальною і потребує постійної уваги з боку фінансових установ та розробників систем безпеки. Виклики, пов’язані зі збільшенням обсягів транзакцій, швидкою зміною схем шахрайства та організованістю зловмисників, підкреслюють потребу в автоматизованих системах на основі машинного навчання, здатних ефективно моніторити та виявляти підозрілі дії в реальному часі.

1.1.2 Підходи до виявлення фінансового шахрайства

Виявлення шахрайства є однією з найважливіших задач у фінансовій галузі, оскільки шахрайські дії можуть завдати значних збитків. У зв'язку з постійним зростанням кількості та складності шахрайських схем, фінансові організації повинні застосовувати сучасні методи та підходи для ефективного виявлення і запобігання незаконним діям.

Підходи до виявлення шахрайства (рисунок 1.2) можуть варіюватися залежно від характеру операцій, обсягу даних і типів шахрайських схем, з якими доводиться стикатися фінансовим установам.

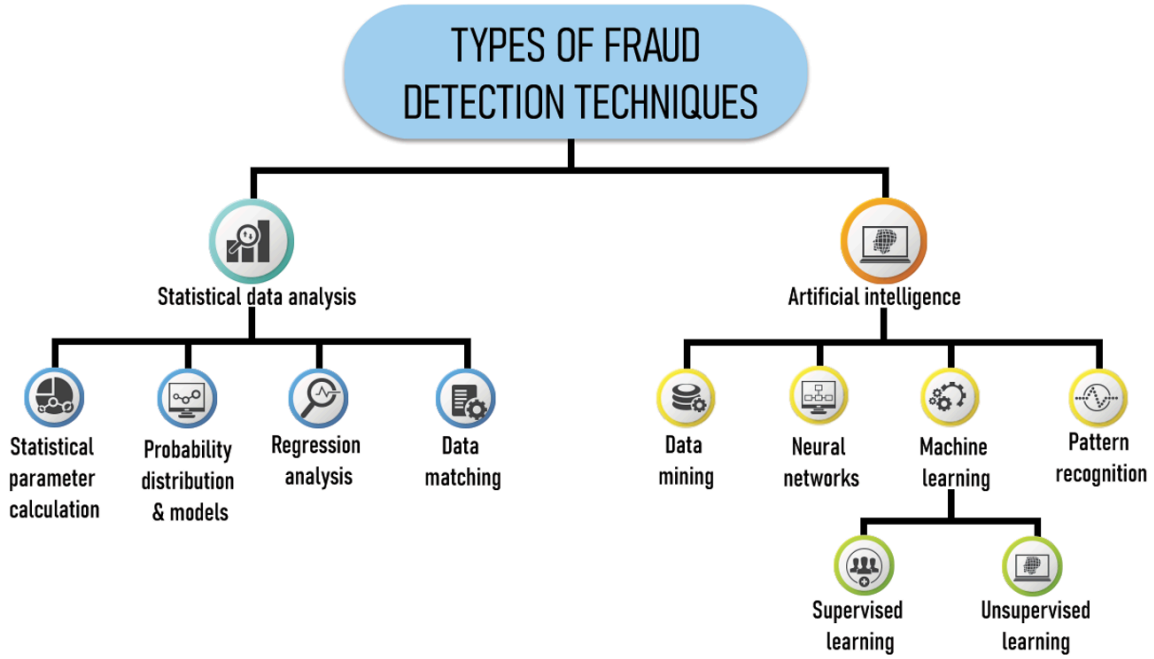

Рисунок 1.2 – Типи виявлення шахрайства

Сучасні технології [1], [2], [3] дозволяють використовувати як традиційні правила та методи статистичного аналізу, так і більш складні моделі, засновані на машинному навчанні та штучному інтелекті. Кожен з цих підходів має свої переваги і недоліки, а також застосовується для різних типів шахрайства. Наприклад, правила моніторингу транзакцій можуть бути ефективними для виявлення відомих шахрайських схем, тоді як машинне навчання дозволяє знаходити нові, раніше невідомі типи шахрайства, аналізуючи великі обсяги даних і виявляючи складні патерни.

Підходи до виявлення шахрайства можна поділити на дві основні категорії: наглядові та ненаглядові методи. Наглядові методи базуються на попередньо мічених даних, де вже відомо, які транзакції є шахрайськими, а які – ні. Ці методи дозволяють навчати моделі на історичних даних для того,

щоб автоматично класифікувати нові транзакції. Ненаглядові методи, навпаки, працюють без попередньої інформації про мічені дані і намагаються знайти аномалії або відхилення від нормальної поведінки в транзакціях. Кожен з цих підходів може бути застосований у різних ситуаціях, залежно від характеру даних та типу шахрайства, яке потрібно виявити.

Іншою важливою частиною виявлення шахрайства є інтеграція методів обробки великих даних та режим реального часу. У сучасних умовах фінансові установи стикаються з величезними обсягами транзакцій, які потребують негайного аналізу для виявлення підозрілих дій. Технології великих даних дозволяють аналізувати транзакції у масштабі та виявляти шахрайські дії ще до завершення операції. Використання методів потокової обробки даних та алгоритмів машинного навчання в реальному часі дозволяє фінансовим установам швидко реагувати на потенційні загрози та блокувати підозрілі транзакції.

Таким чином, підходи до виявлення шахрайства охоплюють широкий спектр методів і технологій, від базових правил моніторингу до передових алгоритмів машинного навчання. Успішна боротьба з шахрайством вимагає комплексного підходу, що включає як технічні рішення, так і стратегії моніторингу та обробки даних. У наступних підрозділах будуть детально розглянуті різні підходи до виявлення шахрайства, зокрема наглядові та ненаглядові методи, а також новітні технології, що дозволяють забезпечити високу ефективність боротьби з фінансовим шахрайством.

1.1.3 Потреба в автоматизованих системах виявлення шахрайства

Зростання обсягів фінансових транзакцій і розвиток цифрових платіжних систем значно ускладнюють задачу моніторингу та виявлення шахрайства. Традиційні методи на основі правил (рисунок 1.3), що використовуються для ідентифікації підозрілих операцій, мають обмежену

ефективність у сучасних умовах. Вони не здатні швидко адаптуватися до нових шахрайських схем і потребують постійного оновлення вручну, що є трудомістким і не завжди забезпечує своєчасне виявлення нових типів загроз. Це призводить до збільшення фінансових втрат для установ та їхніх клієнтів, а також підвищує репутаційні ризики.

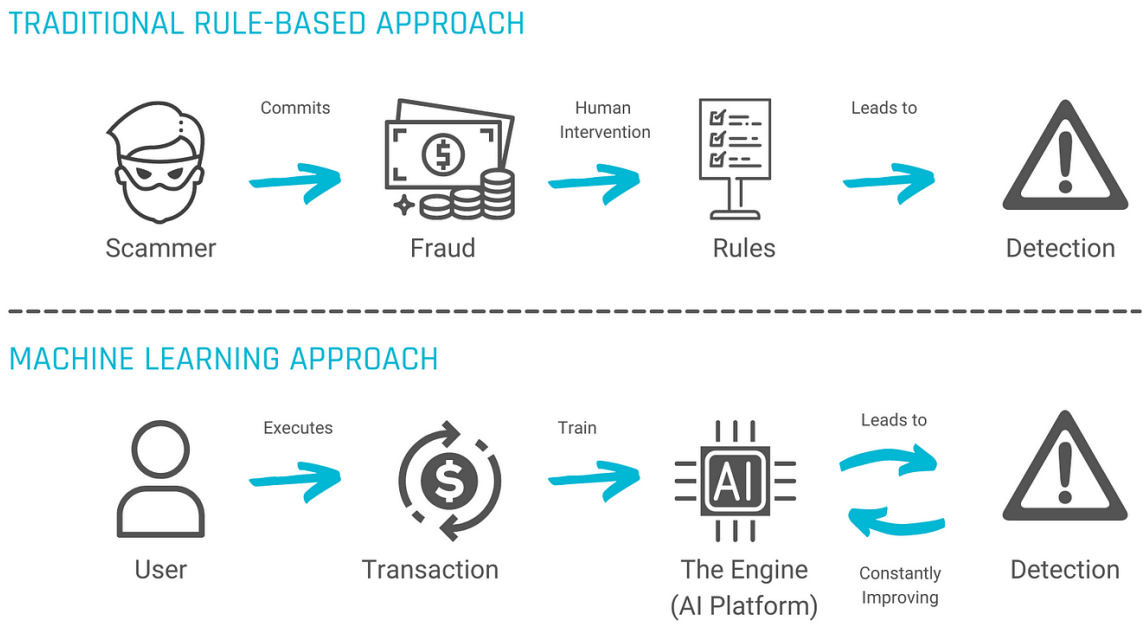

Рисунок 1.3 – Порівняння традиційного підходу на основі правил та підходу машинного навчання у виявленні шахрайства

Необхідність автоматизованих методів виявлення шахрайства зумовлена тим, що вони здатні працювати з великими обсягами даних у режимі реального часу, аналізуючи тисячі транзакцій щосекунди. Такі методи дозволяють автоматично ідентифікувати аномальні патерни поведінки та виявляти підозрілі дії без постійного втручання людини. Крім того, автоматизовані системи на основі сучасних алгоритмів машинного навчання мають здатність самонавчатися на нових даних, що дає їм змогу адаптуватися до змін у поведінці шахраїв і залишатися ефективними навіть при появі нових шахрайських схем.

Автоматизація процесу виявлення шахрайства також дозволяє зменшити витрати фінансових установ на забезпечення безпеки, оскільки знижує потребу в ручній перевірці кожної транзакції. Це робить такі системи більш гнучкими та ефективними, забезпечуючи оперативне реагування на загрози та підвищуючи загальний рівень захисту фінансових операцій. Таким чином, автоматизовані системи стають ключовим інструментом для фінансових установ у боротьбі з шахрайством.

1.1.4 Машинне навчання як підход до виявлення шахрайства

Машинне навчання стає одним із найефективніших інструментів у виявленні шахрайства завдяки своїй здатності автоматично виявляти складні патерни у великих обсягах даних. У порівнянні з традиційними методами, що базуються на фіксованих правилах, алгоритми машинного навчання можуть навчатися на історичних даних і адаптуватися до нових типів загроз. Це особливо актуально у сфері фінансових послуг, де шахраї постійно змінюють свої методи, і статичні правила швидко втрачають ефективність.

Застосування машинного навчання дозволяє фінансовим установам аналізувати тисячі транзакцій у реальному часі, автоматично виявляючи аномалії, які можуть вказувати на шахрайські дії. Завдяки здатності до самонавчання, такі моделі можуть адаптуватися до змін у поведінкових патернах користувачів і реагувати на нові схеми шахрайства. Це забезпечує підвищену точність і швидкість виявлення загроз, що є критично важливим у динамічних умовах сучасних фінансових ринків.

Крім того, машинне навчання дозволяє використовувати передові алгоритми, такі як глибокі нейронні мережі та ансамблеві методи, що дають змогу обробляти складні багатовимірні дані. Це робить машинне навчання не лише ефективним, але й гнучким інструментом для побудови систем виявлення шахрайства, здатних враховувати як історичні дані, так і реальні

часові показники. Таким чином, машинне навчання стає незамінним підходом для виявлення шахрайства у фінансових транзакціях, підвищуючи рівень безпеки та знижуючи ризики фінансових втрат.

1.2 Основи машинного навчання для виявлення шахрайства

Машинне навчання є основою сучасних підходів до виявлення шахрайства, оскільки здатне ефективно працювати з великими обсягами фінансових даних і виявляти приховані аномалії. Завдяки можливостям машинного навчання фінансові установи можуть не лише аналізувати історичні дані, але й застосовувати ці знання для швидкого реагування на нові типи загроз, що є критично важливим в умовах стрімкої еволюції шахрайських схем. Використання машинного навчання забезпечує вищу гнучкість і адаптивність систем безпеки, знижуючи ризики фінансових втрат і репутаційних збитків.

У цьому підрозділі розглядаються основні концепти і методи машинного навчання, що є фундаментом для побудови автоматизованих систем виявлення шахрайства. Спочатку надається загальний огляд ключових підходів машинного навчання, які використовуються для аналізу транзакційних даних, включаючи класифікацію, кластеризацію та виявлення аномалій. Далі розглядаються специфічні методи, такі як дерева рішень, нейронні мережі та Support Vector Machines (SVM), які є основними інструментами для класифікації фінансових операцій. Також увага приділяється ролі класифікаційних методів, оскільки вони забезпечують точність і надійність моделей у процесі виявлення шахрайських дій.

Цей підрозділ закладає основу для розуміння застосування машинного навчання у сфері виявлення шахрайства, надаючи огляд методів і концептів, які використовуються для побудови гнучких і адаптивних систем.

1.2.1 Концепти машинного навчання для виявлення шахрайства

Машинне навчання відіграє центральну роль у виявленні фінансового шахрайства [4] завдяки своїй здатності автоматично знаходити приховані закономірності у великих масивах даних. Основні концепти машинного навчання, що застосовуються у цій сфері, включають класифікацію, кластеризацію, регресію та виявлення аномалій. Ці методи дозволяють фінансовим установам не лише автоматизувати процес моніторингу транзакцій, а й значно підвищити його ефективність, знижуючи ризик хибних спрацьовувань та пропущених шахрайських операцій.

Класифікація є одним із ключових методів у виявленні шахрайства, оскільки дозволяє моделі вчитися на основі мічених даних, де транзакції вже визначені як шахрайські чи чесні. Це дозволяє системі будувати прогностичні моделі, які можуть класифікувати нові транзакції з високою точністю. Метод кластеризації також є цінним інструментом, оскільки він дозволяє виявляти групи транзакцій, що демонструють схожі ознаки та можуть свідчити про певний тип шахрайства. У випадках, коли доступні немічені дані, кластеризація дозволяє знаходити аномалії, навіть якщо шахрайські транзакції раніше не були зареєстровані.

Виявлення аномалій є ще одним важливим концептом у машинному навчанні для виявлення шахрайства. Цей підхід особливо корисний у ситуаціях, коли шахраї змінюють свої схеми, що робить їх непомітними для традиційних методів на основі правил. Аномальні транзакції, які значно відрізняються від загальної поведінки користувача, можуть свідчити про шахрайство, і методи виявлення аномалій дозволяють системі швидко ідентифікувати такі операції. Таким чином, виявлення аномалій доповнює класифікацію та кластеризацію, забезпечуючи більш комплексний підхід до виявлення шахрайських дій.

Регресія є ще одним підходом, що може бути корисним для прогнозування певних параметрів транзакцій і аналізу закономірностей, які

вказують на потенційне шахрайство. Хоча регресія не завжди застосовується для виявлення конкретних шахрайських дій, вона може допомогти у визначенні загальної тенденції та показників ризику, які вказують на підвищену ймовірність шахрайства.

Основні концепти машинного навчання виявляються надзвичайно корисними для розробки систем виявлення шахрайства, оскільки дозволяють автоматизувати аналіз великих обсягів транзакційних даних, підвищуючи ефективність і точність виявлення. Використання різних методів машинного навчання дозволяє будувати багаторівневі системи, здатні реагувати на динамічні зміни у схемах шахрайства і швидко адаптуватися до нових загроз.

1.2.2 Огляд методів для виявлення шахрайства

Для виявлення шахрайства у фінансових транзакціях застосовуються різноманітні методи машинного навчання, кожен з яких має свої переваги та сфери ефективного використання. Одним із найбільш розповсюджених методів є логістична регресія, яка часто використовується для бінарної класифікації, наприклад, для визначення, чи є транзакція шахрайською чи чесною. Завдяки своїй простоті та ефективності у випадках, коли транзакційні дані мають лінійні зв'язки, логістична регресія залишається одним із базових підходів для виявлення шахрайства.

Іншим популярним методом є дерева рішень, які дозволяють класифікувати транзакції на основі послідовних критеріїв, створюючи гнучкі та інтуїтивно зрозумілі моделі. Дерева рішень також часто використовуються у поєднанні з методами ансамблевого навчання, такими як Random Forest і Gradient Boosting, що дозволяє підвищити точність моделі шляхом об'єднання результатів кількох дерев у більш стабільний і точний прогноз. Ці ансамблеві методи здатні ефективно обробляти складні

закономірності в транзакційних даних і знижувати ймовірність хибно-позитивних результатів, що є важливим для виявлення шахрайства.

Нейронні мережі, зокрема глибоке навчання, є ще одним ефективним методом у боротьбі з шахрайством. Використання багатошарових нейронних мереж дозволяє обробляти великі обсяги даних і виявляти складні, багатовимірні патерни, які часто зустрічаються в шахрайських діях. Глибокі нейронні мережі, такі як згорткові (CNN) і рекурентні (RNN), добре підходять для обробки різних аспектів транзакційних даних: CNN допомагають виділяти значущі ознаки, а RNN корисні для аналізу послідовностей транзакцій, що можуть вказувати на шахрайську поведінку.

Метод підтримуючих векторів (SVM) також широко застосовується у виявленні шахрайства, особливо для класифікації, коли дані мають складні розподіли. SVM добре працює у випадках, коли потрібно чітко розмежувати шахрайські та чесні транзакції, навіть якщо межі між класами є складними. Завдяки своїй здатності до ефективної класифікації у багатовимірному просторі, SVM може успішно знаходити шахрайські транзакції у складних фінансових даних.

Методи кластеризації також відіграють важливу роль у виявленні шахрайства, особливо коли мічені дані недоступні. Алгоритми, такі як k-means та DBSCAN, дозволяють групувати подібні транзакції та виявляти ті, що значно відрізняються від нормальних, що може вказувати на підозрілу активність. Кластеризація допомагає визначити нові типи шахрайства, які ще не мали міток у системі.

Нарешті, методи виявлення аномалій стали важливим інструментом для виявлення рідкісних і нових видів шахрайських транзакцій. Ці методи можуть виявляти відхилення від стандартних патернів у фінансових даних, сигналізуючи про потенційні шахрайські операції навіть за відсутності детальних міток або інформації про попередні шахрайства.

Кожен із цих методів вносить свій внесок у виявлення шахрайства, дозволяючи фінансовим установам створювати комплексні системи, які забезпечують високий рівень захисту.

1.3 Виклики у виявленні фінансового шахрайства

Виявлення фінансового шахрайства залишається однією з найбільш складних і динамічних задач для сучасних фінансових установ. Незважаючи на розробку і впровадження потужних алгоритмів машинного навчання, існує низка значних викликів, що ускладнюють процес виявлення шахрайських транзакцій. Основні труднощі пов’язані як зі специфікою фінансових даних, так і з динамічністю шахрайських схем, що швидко адаптуються до нових методів захисту.

Цей підрозділ досліджує основні виклики у виявленні фінансового шахрайства, розглядаючи особливості роботи з фінансовими даними, проблему незбалансованості класів та потребу в адаптивності алгоритмів для ефективного реагування на постійно змінювані схеми шахрайства, щоб захищати фінансові установи.

1.3.1 Специфіка обробки фінансових даних

Обробка фінансових даних для виявлення шахрайства є складним і багатогранним процесом (рисунок 1.4), який вимагає врахування специфічних особливостей фінансових транзакцій і самих даних. Оскільки шахрайство може призвести до значних фінансових втрат як для окремих користувачів, так і для фінансових установ, ефективне виявлення підозрілих операцій залежить від того, наскільки якісно та точно обробляються ці дані. Вони часто включають конфіденційну інформацію, вимагають швидкої обробки у режимі реального часу та потребують високого рівня точності для запобігання шахрайству без зайвих помилкових спрацьовувань.

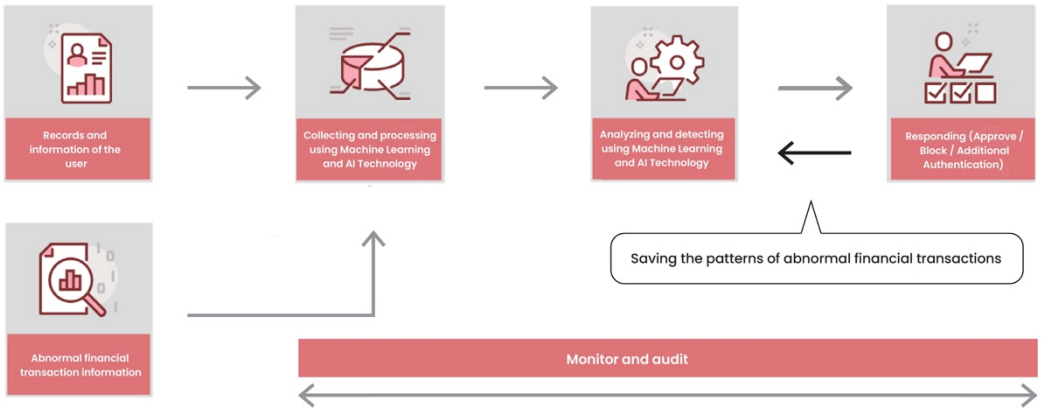

Рисунок 1.4 – Процес виявлення фінансового шахрайства

Основною проблемою, з якою стикаються фахівці з обробки фінансових даних, є різноманітність і складність самих даних. Транзакції можуть включати безліч різних типів інформації: від суми операції, до часу та місця її здійснення, типу платіжного засобу, профілю клієнта, його історії операцій і поведінкових характеристик. Таке різноманіття вимагає створення складних моделей обробки, здатних інтегрувати різні типи даних та використовувати їх для виявлення аномалій і потенційних шахрайських схем.

Проблеми якості даних також є одним із центральних викликів у процесі виявлення шахрайства. Недостатньо повні або некоректні дані можуть значно знизити точність моделей, що використовуються для виявлення шахрайських дій. Неточні або відсутні дані можуть призводити до хибних прогнозів, що, у свою чергу, може стати причиною або пропуску шахрайських дій, або блокування легітимних операцій. Тому питання забезпечення якості даних є критично важливим для точності й ефективності систем моніторингу транзакцій.

Ще однією важливою особливістю обробки фінансових даних є їхній великомасштабний характер. У сучасних фінансових системах обробляються мільйони транзакцій щодня, що вимагає використання

високопродуктивних технологій для забезпечення своєчасного аналізу цих операцій. Обробка великих обсягів даних не лише підвищує вимоги до обчислювальних ресурсів, але й потребує спеціальних методів зберігання й обробки даних, таких як технології великих даних, потокова обробка та розподілені обчислення. Ці технології дозволяють аналізувати дані в реальному часі, що є ключовим для швидкого реагування на потенційні шахрайські дії.

Окрім того, одним із головних викликів у сфері виявлення шахрайства є обробка даних для незбалансованих класів. У фінансових транзакціях випадки шахрайства зазвичай є рідкісними порівняно з загальною кількістю операцій, що створює проблеми для багатьох алгоритмів машинного навчання. Якщо клас шахрайських операцій значно менший за легітимні транзакції, це може призвести до того, що моделі будуть схильні до переваги одного класу, ігноруючи або недостатньо реагуючи на інший. Для вирішення цієї проблеми застосовуються спеціальні методи [5], [6], [7], такі як ресемплінг даних, використання вагових коефіцієнтів або застосування алгоритмів, що спеціально розроблені для роботи з незбалансованими класами.

Отже, обробка фінансових даних для виявлення шахрайства вимагає врахування кількох важливих аспектів, таких як типи даних, забезпечення їх якості, робота з великими обсягами інформації та обробка незбалансованих класів. Ефективна робота з цими даними є ключем до точного виявлення шахрайських дій, що допомагає фінансовим установам мінімізувати ризики та підвищити рівень захисту своїх клієнтів.

1.3.2 Проблема класової незбалансованості даних

Проблема класової незбалансованості є одним із найсерйозніших викликів у виявленні фінансового шахрайства за допомогою машинного навчання. У більшості фінансових даних кількість шахрайських транзакцій

є значно меншою порівняно з кількістю чесних, що створює серйозний дисбаланс між класами. Це може призвести до того, що моделі машинного навчання будуть «схильні» до класифікації більшості транзакцій як чесних, що спричиняє високу частоту хибно-негативних результатів і знижує ефективність виявлення шахрайства.

Моделі, що навчаються на незбалансованих даних, часто орієнтовані на більшість, тобто на чесні транзакції, що призводить до ігнорування меншої, але критично важливої частини шахрайських операцій. Це означає, що навіть модель з високою загальною точністю може демонструвати низьку ефективність саме у виявленні шахрайських дій, оскільки вона може не надавати достатньої уваги рідкісним, але важливим шахрайським транзакціям. У результаті знижується здатність системи адекватно реагувати на загрози та своєчасно блокувати підозрілі операції.

Для вирішення цієї проблеми застосовуються різні методи корекції класової незбалансованості. Одним із підходів є ресемплінг, який включає або дублювання шахрайських транзакцій, over-sampling, або видалення частини чесних транзакцій, under-sampling. Такий підхід дозволяє збалансувати дані, однак може спричинити появу нових проблем, таких як перенавчання або втрата важливих даних. Також у випадку незбалансованих даних використовуються спеціалізовані метрики, як-от F1-міра або ROC-AUC, які допомагають більш точно оцінити здатність моделі виявляти шахрайські транзакції, що забезпечує більш адекватну оцінку результатів і дозволяє врахувати специфіку завдання.

Загалом, класова незбалансованість даних залишається значною проблемою у виявленні шахрайства, і її успішне подолання вимагає застосування спеціальних методів та оптимізації моделей, що дозволяє підвищити ефективність виявлення шахрайських дій у фінансових транзакціях.

1.3.3 Динаміка шахрайських схем і адаптація алгоритмів

Постійна еволюція шахрайських схем є значним викликом для систем виявлення шахрайства, оскільки шахраї безперервно адаптуються до нових методів захисту та створюють дедалі складніші схеми для обману систем. Зловмисники використовують різні техніки, включаючи соціальну інженерію, автоматизовані боти, підроблені акаунти та складні мережі для приховування своїх дій. Це ускладнює завдання фінансових установ, які повинні постійно оновлювати свої системи безпеки, щоб не лише виявляти вже відомі шахрайські схеми, але й прогнозувати нові, які ще не потрапили у бази даних і не мають чітких шаблонів.

Адаптація моделей машинного навчання до швидких змін у шахрайських схемах є складним завданням, оскільки алгоритми можуть бути ефективними лише до певного моменту, коли з'являються нові види загроз. Постійне оновлення моделей потребує доступу до актуальних даних та регулярного навчання на нових прикладах шахрайства, що є ресурсно-інтенсивним процесом. Крім того, шахрайські дії часто стають більш тонкими і складними, що вимагає від моделей здатності до самонавчання і швидкої адаптації, щоб бути готовими до нових викликів у режимі реального часу.

Методи виявлення, які покладаються лише на попередньо визначені правила або застарілі моделі, можуть втратити свою актуальність і ефективність, оскільки шахраї швидко змінюють тактику і можуть уникати виявлення. Тому зростає потреба у впровадженні адаптивних алгоритмів, здатних до самонавчання, які можуть підлаштовуватися до нових умов і виявляти патерни, що раніше не фіксувалися. Наприклад, методи глибокого навчання або навчання з підкріпленням, які мають можливість коригувати свої стратегії на основі отриманих результатів, є ефективними для виявлення нових типів шахрайства.

Постійна еволюція шахрайських схем підкреслює важливість створення гнучких і динамічних систем, здатних швидко реагувати на зміни. Це забезпечує більш ефективний захист фінансових установ, оскільки автоматизовані системи можуть не лише відстежувати нові схеми, але й передбачати можливі майбутні загрози, що підвищує загальний рівень безпеки у фінансовому середовищі.

1.4 Гібридні та ансамблеві підходи до виявлення шахрайства

Гібридні та ансамблеві підходи до виявлення шахрайства є перспективними методами, що дозволяють підвищити точність і надійність системи за рахунок комбінування декількох алгоритмів машинного навчання. Ці методи враховують складність і динамічний характер шахрайських схем, що постійно змінюються й адаптуються до нових умов.

Гібридні підходи об'єднують різні типи алгоритмів, зокрема класифікаційні методи й методи кластеризації, що дає змогу одночасно проводити ідентифікацію відомих шаблонів шахрайства і виявлення аномалій у транзакційних даних. Наприклад, один з компонентів гібридної системи може класифікувати дані на основі навчання з учителем, визначаючи транзакції як шахрайські або чесні, тоді як інший компонент, заснований на кластеризації, виявляє нові та незвичні патерни, які раніше не фіксувалися в системі. Такий підхід забезпечує вищу точність, оскільки він здатен реагувати на невідомі шахрайські схеми, а не лише на ті, що вже були визначені.

Ансамблеві методи, своєю чергою, включають поєднання кількох однакових або різних алгоритмів для створення більш стабільної та точної моделі. Одним із найпопулярніших ансамблевих методів є Random Forest, який використовує множину дерев рішень для підвищення точності класифікації. Інші методи, як-от градієнтний бустинг, об'єднують послідовність моделей, кожна з яких коригує помилки попередньої, що

підвищує загальну продуктивність системи. Ансамблеві методи дозволяють зменшити ймовірність помилкових класифікацій, що особливо важливо у сфері фінансового шахрайства, де навіть невеликі похибки можуть призвести до значних фінансових втрат.

Гібридні та ансамблеві підходи також мають здатність працювати з великими обсягами даних, що є важливим у фінансових системах. Використання декількох методів одночасно дозволяє системам швидше адаптуватися до нових шахрайських схем і підвищує точність виявлення. Окрім цього, комбінування різних підходів допомагає мінімізувати недоліки кожного окремого алгоритму, що робить гібридні та ансамблеві системи особливо ефективними в динамічному середовищі фінансових транзакцій.

Таким чином, гібридні та ансамблеві методи виявлення шахрайства є ефективними інструментами для покращення якості та точності класифікації шахрайських транзакцій. Завдяки своїй здатності поєднувати сильні сторони різних моделей, ці підходи забезпечують кращу адаптивність та підвищують загальний рівень безпеки фінансових систем.

1.5 Метрики оцінки методів виявлення шахрайства

Оцінка ефективності методів виявлення шахрайства є критично важливим етапом у побудові системи, оскільки від точності та надійності результатів залежить, наскільки ефективно система зможе виявляти шахрайські транзакції й мінімізувати хибно-позитивні та хибно-негативні результати. Для вимірювання успішності моделей машинного навчання використовуються спеціальні метрики, які дозволяють детально оцінити, наскільки модель правильно класифікує шахрайські та чесні транзакції. Це особливо важливо у випадках, коли кількість шахрайських транзакцій є невеликою порівняно з чесними, оскільки навіть невелика похибка може призвести до серйозних фінансових втрат.

Однією з ключових метрик є точність, яка показує загальну частку правильно класифікованих транзакцій. Однак для завдань виявлення шахрайства точність може бути недостатньо інформативною, оскільки висока кількість чесних транзакцій може завищити показник точності навіть тоді, коли шахрайські транзакції не виявляються належним чином. Тому частіше використовуються такі метрики, як точність позитивного класу, precision, що показує частку правильно виявлених шахрайських транзакцій серед усіх транзакцій, класифікованих як шахрайські, та повнота, recall, що вказує на здатність моделі виявляти шахрайські транзакції серед усіх фактично шахрайських операцій.

F1-міра є також важливою метрикою для оцінки методів виявлення шахрайства, оскільки вона враховує як precision, так і recall, забезпечуючи збалансовану оцінку ефективності моделі. Це особливо корисно, коли важливо уникнути як пропуску шахрайських дій, так і зайвого спрацьовування на чесні транзакції. Крім того, метрика ROC-AUC дозволяє оцінити загальну продуктивність моделі, вимірюючи здатність відрізняти шахрайські транзакції від чесних, що особливо важливо для порівняння різних моделей і вибору оптимальної.

Завдяки таким метрикам можна отримати детальне уявлення про сильні та слабкі сторони різних моделей машинного навчання. Такі показники допомагають оцінити, як моделі поводяться в умовах незбалансованих даних і наскільки вони здатні адаптуватися до складних і мінливих умов. Оцінка за допомогою відповідних метрик дозволяє не лише підвищити якість системи виявлення шахрайства, але й забезпечує точне налаштування моделі для роботи в умовах реальних фінансових транзакцій.

1.6 Постановка задачі дослідження

Основна мета цього дослідження полягає в детальному аналізі існуючих підходів до виявлення фінансового шахрайства за допомогою

методів машинного навчання, зокрема методів навчання з учителем, які зарекомендували себе як ефективні інструменти для класифікації транзакцій.

Дослідження включає глибокий огляд наукових джерел, що висвітлюють сучасні методи боротьби з шахрайством у фінансових транзакціях, а також експериментальне застосування найбільш ефективних алгоритмів машинного навчання на реальних транзакційних даних.

Окремим завданням є оцінити ефективність цих алгоритмів, порівнюючи різні методи класифікації для вибору оптимального підходу до виявлення шахрайства з урахуванням характерних особливостей таких даних, як класова незбалансованість, коли кількість шахрайських випадків є значно меншою, ніж кількість чесних операцій.

Особлива увага приділяється аналізу впливу попередньої обробки даних, вибору алгоритму та параметрів навчання на точність моделі. У процесі дослідження розглядаються як традиційні, так і сучасні методи машинного навчання, зокрема логістична регресія та Random Forest, що дозволяє оцінити їхню ефективність у реальному сценарії.

Крім того, дослідження спрямоване на виявлення сильних і слабких сторін кожної з моделей та визначення умов, за яких їх використання є найбільш доцільним у фінансовій сфері.

Таким чином, завдання дослідження охоплюють не лише всебічний аналіз теоретичної бази, що включає сучасні методи та підходи до класифікації фінансових даних, але й практичну перевірку алгоритмів, яка дозволяє визначити найбільш точні та надійні методи для виявлення шахрайських дій у фінансових транзакціях, що має вагоме значення для підвищення рівня безпеки платіжних систем. Результатом роботи має стати вибір оптимальної моделі для побудови системи виявлення шахрайства у сучасних фінансових установах, що мають на меті захищати транзакції від недоброчесних клієнтів.

2 ТЕОРЕТИЧНІ ДОСЛІДЖЕННЯ

2.1 Класифікація як підхід до виявлення шахрайства

Класифікація є одним із ключових підходів у машинному навчанні, який активно застосовується для вирішення задач виявлення шахрайства у фінансових транзакціях. Цей метод дозволяє автоматизувати процес аналізу даних, розподіляючи транзакції на певні категорії, такі як шахрайські та чесні. Завдяки своїй здатності швидко обробляти великі обсяги інформації та виявляти складні закономірності, класифікація стала основою для побудови систем моніторингу та запобігання шахрайству у фінансовій сфері.

У цьому підрозділі буде розглянуто роль класифікації у виявленні шахрайства, основні концепти, на яких ґрунтується цей підхід, а також ключові виклики, з якими стикаються розробники таких систем. Це створить базу для глибшого розуміння застосування класифікації у боротьбі з фінансовими злочинами, яка буде детально розкрита у наступних розділах.

2.1.1 Роль класифікації у виявленні шахрайства

Класифікація відіграє ключову роль у виявленні шахрайства у фінансових транзакціях завдяки своїй здатності автоматизувати процес ідентифікації підозрілих дій. Фінансові системи генерують величезні обсяги даних, які потребують швидкої обробки для мінімізації ризиків. У таких умовах традиційні підходи виявлення шахрайства, засновані на ручному аналізі чи фіксованих правилах, часто виявляються недостатньо ефективними.

Однією з основних переваг класифікації є її здатність знаходити складні закономірності в даних, які можуть свідчити про підозрілу активність. За допомогою навчання на історичних даних класифікаційні

моделі здатні адаптуватися до змінних умов і виявляти нові схеми шахрайства. Це особливо важливо, оскільки шахраї постійно вдосконалюють свої методи, намагаючись обійти існуючі системи захисту.

Крім того, класифікація забезпечує можливість аналізу великої кількості ознак, таких як географічні дані, час проведення операцій, частота транзакцій та інші параметри, які можуть бути важливими для виявлення шахрайства.

Роль класифікації також полягає в її універсальності. Вона може використовуватися як для виявлення відомих шахрайських схем, так і для ідентифікації нових патернів, які раніше не були зафіксовані. Завдяки цьому класифікація залишається одним із найбільш важливих інструментів у боротьбі з фінансовим шахрайством, дозволяючи не лише знижувати ризики, але й підвищувати довіру клієнтів до фінансових установ.

2.1.2 Основи класифікації у машинному навчанні

Класифікація у машинному навчанні є основою для вирішення багатьох завдань, зокрема виявлення шахрайства у фінансових транзакціях. Вона передбачає процес розподілу даних на певні категорії або класи на основі їхніх характеристик (рисунок 2.1).

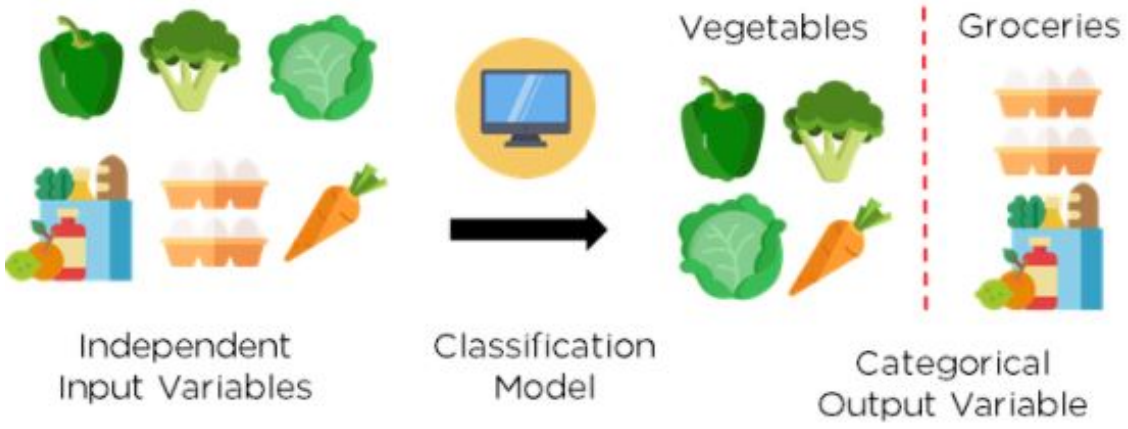

Рисунок 2.1 – Процес класифікації на прикладі продуктів

У контексті виявлення шахрайства це може бути класифікація транзакцій як «чесних» або «шахрайських». Такий підхід дозволяє ефективно аналізувати дані та знаходити патерни, що свідчать про аномальну поведінку.

Основний процес класифікації включає кілька етапів: збір і підготовка даних, вибір моделі, навчання моделі на мічених даних, її оцінювання та використання для прогнозування. Модель навчається на історичних даних, де кожен запис має мітку, яка вказує на його клас. Після навчання модель може застосовуватися для класифікації нових даних.

Однією з головних характеристик класифікаційних моделей є їхня здатність працювати з багатовимірними даними. У задачах виявлення шахрайства враховуються численні ознаки, такі як час, місце транзакції, частота операцій та суми, що дозволяє створювати більш точні та надійні моделі. Наприклад, вхідні дані можуть бути представлені у вигляді багатовимірного простору, де кожен вимір відповідає одній з характеристик транзакції, а класи визначаються на основі поділу цього простору.

Таким чином, основи класифікації у машинному навчанні забезпечують ефективний підхід до аналізу даних, що дозволяє автоматизувати процеси моніторингу та виявлення підозрілих транзакцій. Завдяки своїй здатності швидко адаптуватися до змін у поведінці користувачів і схемах шахрайства, класифікація залишається однією з найбільш затребуваних технік у цій сфері.

2.1.3 Виклики класифікації у задачах виявлення шахрайства

Класифікація у задачах виявлення шахрайства стикається з низкою викликів, які ускладнюють ефективне застосування навіть найсучасніших методів машинного навчання. Основні труднощі полягають у природі фінансових даних, динамічності шахрайських схем і необхідності

досягнення балансу між точністю класифікації та її реалістичним виконанням у режимі реального часу.

Одним із ключових викликів є класова незбалансованість даних. У більшості фінансових систем частка шахрайських транзакцій у загальному обсязі є надзвичайно малою. Це призводить до того, що моделі машинного навчання можуть бути схильні до надмірного класифікування даних як «чесних», залишаючи шахрайські дії невиявленими. Традиційні метрики, такі як точність, у таких випадках не дають повного уявлення про ефективність моделі, тому необхідно використовувати більш специфічні показники, наприклад, F1-міру або ROC-AUC.

Динамічність шахрайських схем є ще однією суттєвою проблемою. Шахраї постійно адаптують свої дії до нових методів захисту, розробляючи нові підходи для обходу систем виявлення. Це вимагає від класифікаційних моделей не лише швидкої адаптації до змін, але й здатності виявляти нові, раніше невідомі типи шахрайства. Методи, які базуються лише на історичних даних, часто втрачають актуальність через постійний розвиток загроз.

Ще одним викликом є великий обсяг даних, що обробляються у фінансових системах. Класифікація у режимі реального часу вимагає від систем здатності швидко та ефективно аналізувати тисячі транзакцій щосекунди. Це потребує значних обчислювальних ресурсів і оптимізованих алгоритмів, які можуть обробляти багатовимірні дані з мінімальними затримками.

Не менш важливим аспектом є необхідність забезпечення прозорості моделей класифікації. Фінансові установи часто зіштовхуються з питанням довіри до автоматизованих систем, тому моделі повинні бути зрозумілими для користувачів та регуляторів. Особливо це стосується більш складних методів, таких як глибокі нейронні мережі, які часто працюють як «чорні ящики», надаючи результати без чіткого пояснення, як вони були отримані.

Таким чином, виклики класифікації у виявленні шахрайства охоплюють як технічні аспекти, пов’язані з якістю даних і продуктивністю алгоритмів, так і організаційні питання, такі як прозорість та адаптивність моделей. Успішне подолання цих проблем вимагає використання новітніх технологій, регулярного оновлення моделей і комплексного підходу до побудови систем виявлення шахрайства.

2.2 Ключові методи класифікації для виявленні шахрайства

Методи класифікації є основою для створення ефективних систем виявлення шахрайства у фінансових транзакціях. Ці методи забезпечують можливість аналізувати великі обсяги даних, визначати аномальні транзакції та прогнозувати ризики шахрайства з високою точністю. У контексті машинного навчання класифікація використовується для розподілу транзакцій на дві основні категорії: «чесні» та «шахрайські», що дозволяє автоматизувати процес прийняття рішень і знижує потребу у втручанні людини.

Серед методів класифікації, які знаходять застосування у сфері виявлення шахрайства, є як прості, так і більш складні підходи. Вони охоплюють як традиційні статистичні моделі, так і сучасні алгоритми машинного навчання, які використовують потужність обчислень для роботи з багатовимірними даними. Кожен із методів має свої особливості, переваги та обмеження, що визначає доцільність його використання в різних умовах і для різних задач.

Методи класифікації можуть працювати як із лінійними, так і з нелінійними залежностями в даних, що дозволяє враховувати складні взаємозв’язки між характеристиками транзакцій. Вибір конкретного алгоритму залежить від специфіки проблеми, обсягу та якості даних, а також від цілей дослідження. У процесі створення моделей важливо

враховувати такі аспекти, як класова незбалансованість, точність прогнозів та здатність алгоритмів до генералізації.

Методи класифікації у виявленні шахрайства охоплюють широкий спектр підходів, кожен із яких має свої особливості та застосування. На рисунку (рисунок 2.2) представлено основні методи, які використовуються в системах машинного навчання для аналізу фінансових транзакцій: логістична регресія, Random Forest, метод опорних векторів, байєсові методи та нейронні мережі. Ці методи забезпечують ефективний інструментарій для розпізнавання шахрайських дій, дозволяючи враховувати як лінійні, так і нелінійні взаємозв'язки в даних, що робить їх ключовими елементами сучасних систем виявлення шахрайства.

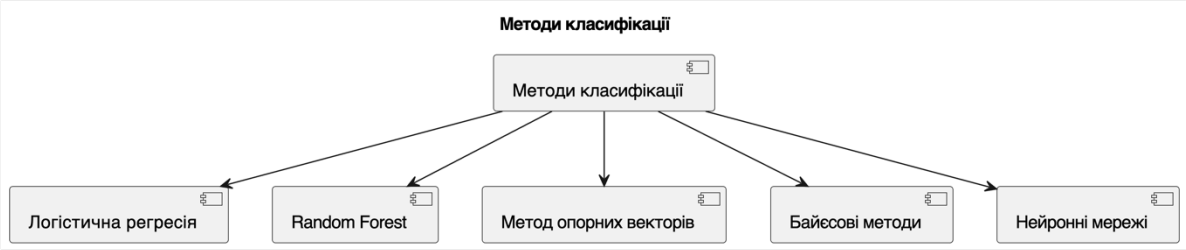

Рисунок 2.2 – Основні методи класифікації у виявленні шахрайств

Цей підрозділ присвячений огляду основних методів класифікації, які використовуються для виявлення шахрайства. Методи класифікації є невід'ємною частиною машинного навчання, що забезпечує автоматизацію процесу ідентифікації підозрілих операцій на основі аналізу великих обсягів даних. У подальших підрозділах буде розглянуто ключові методи, які показали високу ефективність у цій сфері, зокрема їхні переваги, недоліки та особливості застосування в умовах реальних фінансових потоків. Важливість аналізу цих методів обумовлена необхідністю вибору найбільш оптимальних підходів для підвищення виявлення шахрайства.

2.2.1 Логістична регресія

Логістична регресія є одним із найпоширеніших методів машинного навчання, який використовується для задач класифікації, зокрема для виявлення фінансового шахрайства. Її популярність зумовлена простотою реалізації, інтерпретованістю результатів та високою ефективністю у випадках, коли дані добре структуровані та мають лінійну залежність між вхідними змінними. Як ілюструє рисунок (рисунок 2.3), логістична регресія базується на використанні вхідних ознак X_1, X_2, X_3, які після застосування вагових коефіцієнтів θ_1, θ_2, θ_3, визначають імовірність приналежності до певного класу, наприклад, шахрайської або чесної транзакції.

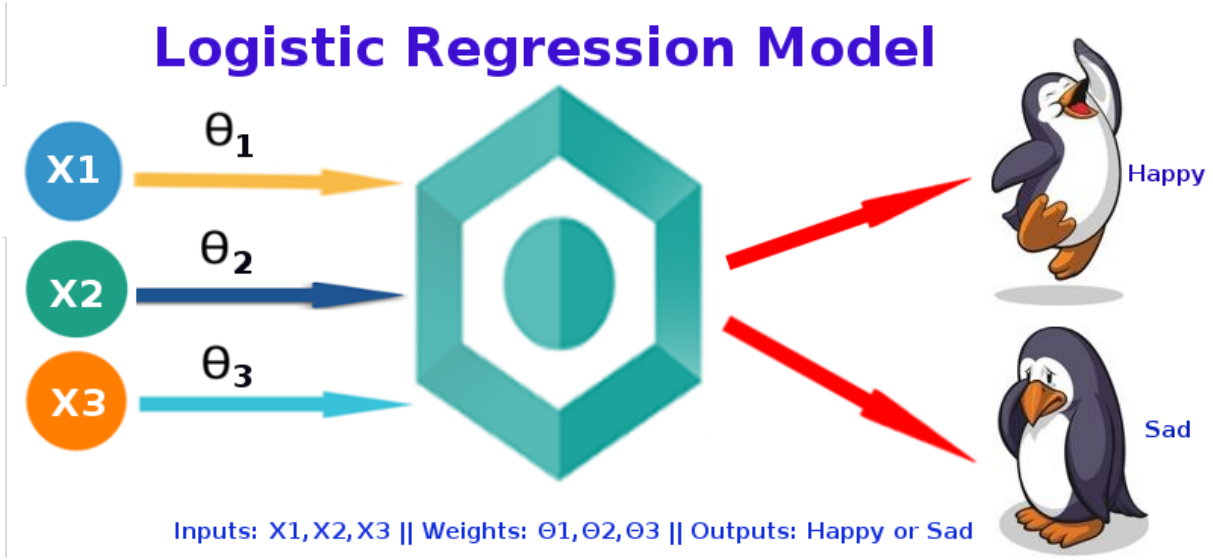

Рисунок 2.3 – Ілюстрація роботи логістичної регресії

Математична основа логістичної регресії полягає у використанні логістичної функції, сигмоїдальної, яка перетворює лінійну комбінацію вхідних ознак у значення, що лежить у діапазоні від 0 до 1. Це значення інтерпретується як ймовірність приналежності об'єкта до одного з двох класів. Для задач виявлення шахрайства клас із ймовірністю вище певного порогу, наприклад 0.5, може бути ідентифікований як шахрайський.

Основною перевагою логістичної регресії є її здатність працювати з невеликими та добре збалансованими наборами даних, де важлива інтерпретація результатів. Наприклад, у задачах виявлення шахрайства фінансові установи можуть використовувати модель для аналізу таких ознак, як сума транзакції, частота операцій або місцезнаходження користувача. Вагові коефіцієнти дозволяють зрозуміти, які фактори найбільше впливають на рішення моделі.

Однак логістична регресія має свої обмеження. Вона ефективна лише за наявності лінійної залежності між змінними, а її продуктивність може значно знижуватись у разі, якщо дані мають високу нелінійність або кореляцію між ознаками. Крім того, метод чутливий до класової незбалансованості, яка часто зустрічається у фінансових даних, де частка шахрайських транзакцій значно менша за частку чесних. Для вирішення цієї проблеми можуть застосовуватись техніки підготовки даних, зокрема ресемплінг або корекція ваг.

Ще одним викликом є необхідність врахування мультиколінеарності між ознаками, яка може спотворювати результати моделі. Для усунення цієї проблеми часто застосовують регуляризацію, яка допомагає зменшити вплив менш важливих ознак. Найпоширенішими методами регуляризації є L1 (Lasso) та L2 (Ridge), які додають штраф за високі значення коефіцієнтів у функцію втрат, тим самим підвищуючи стабільність моделі.

Застосування логістичної регресії для виявлення шахрайства у фінансових транзакціях має широкий спектр практичних реалізацій. Вона використовується для моніторингу транзакцій у реальному часі, оцінки ризиків і підтримки прийняття рішень.

Таким чином, логістична регресія залишається популярним методом класифікації завдяки своїй ефективності та інтерпретованості. Вона є базовим інструментом для аналізу фінансових даних у задачах виявлення шахрайства.

2.2.2 Random Forest

Random Forest є потужним і широко використовуваним методом машинного навчання, особливо для задач класифікації та регресії. Цей алгоритм належить до ансамблевих методів і базується на об'єднанні результатів кількох дерев рішень для покращення точності прогнозування. У контексті виявлення шахрайства Random Forest показав високу ефективність завдяки своїй здатності обробляти великі обсяги даних, що містять численні змінні, і забезпечувати стійкість до перенавчання.

Основна ідея Random Forest полягає у створенні ансамблю дерев рішень, які будуються на різних випадкових підмножинах даних та ознак. Кожне дерево навчається незалежно, і результат обчислюється шляхом голосування (для задач класифікації) або усереднення (для задач регресії). Цей підхід (рисунок 2.4) дозволяє компенсувати похибки окремих дерев, підвищуючи загальну продуктивність моделі.

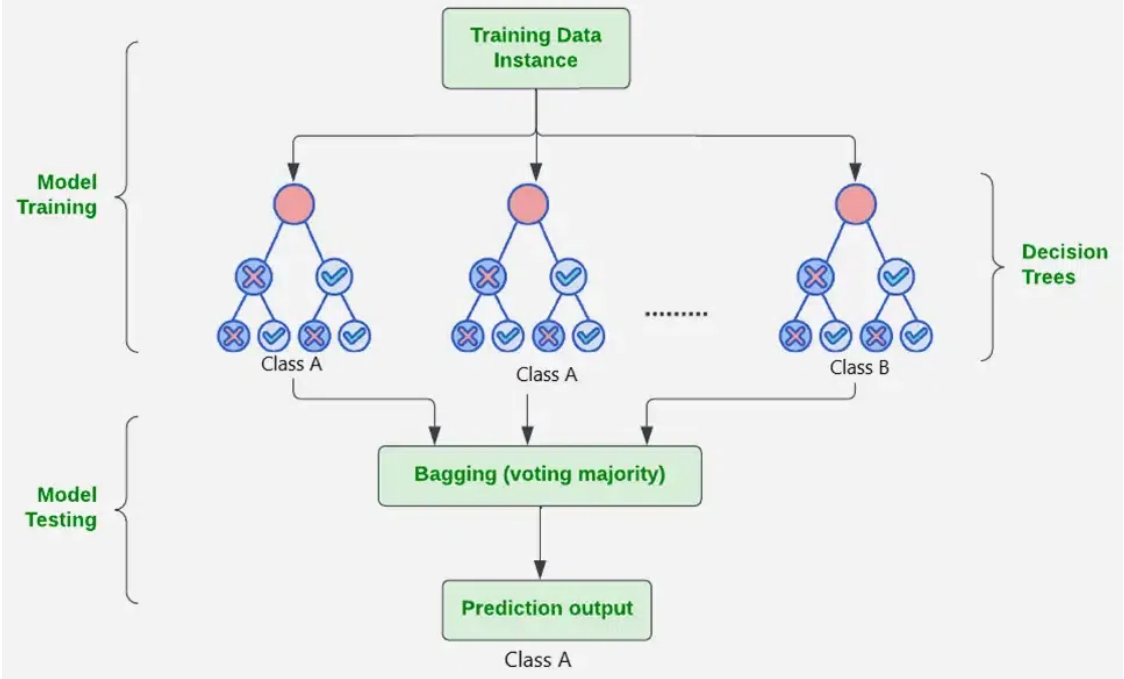

Рисунок 2.4 – Ілюстрація роботи Random Forest

Механізм роботи Random Forest складається з кількох ключових етапів. Спочатку вибирається підмножина даних за методом бутстрепу, random sampling with replacement. Потім, для кожного дерева, на кожному вузлі використовується випадковий набір ознак, щоб мінімізувати кореляцію між деревами. Такий підхід зменшує ризик перенавчання і дозволяє моделі враховувати широкий спектр патернів у даних.

Однією з основних переваг Random Forest є його здатність обробляти великі обсяги даних із численними ознаками, включаючи дані, що містять пропущені значення або сильну кореляцію між змінними. У задачах виявлення шахрайства це дозволяє аналізувати складні взаємозв'язки між такими характеристиками, як сума транзакції, час операції, географічне розташування та історія попередніх дій користувача.

Ще однією важливою властивістю Random Forest є можливість оцінки важливості ознак. Це означає, що модель може виявити, які з характеристик даних найбільше впливають на класифікацію транзакцій як шахрайських або чесних. Така функціональність є цінною для фінансових установ, оскільки дозволяє їм зосередитися на ключових аспектах моніторингу та вдосконалення систем безпеки.

Однак Random Forest також має певні обмеження. Наприклад, цей метод може бути менш ефективним для наборів даних із високим рівнем шуму або для задач із великою кількістю категорій, що може уповільнювати обчислення.

У контексті виявлення шахрайства Random Forest часто використовується для реального часу моніторингу фінансових транзакцій. Його здатність до швидкого навчання та передбачення робить цей метод ідеальним для інтеграції в системи моніторингу, які повинні забезпечувати точність та надійність у динамічному середовищі фінансових операцій. Random Forest є незамінним інструментом у боротьбі з шахрайством завдяки своїй гнучкості та здатності працювати з великими обсягами даних.

2.2.3 Метод опорних векторів

Метод опорних векторів (Support Vector Machines, SVM) є одним із найбільш популярних і ефективних методів машинного навчання, який широко застосовується для класифікації фінансових даних, зокрема для виявлення шахрайства. Основна ідея SVM полягає у пошуку гіперплощини, яка максимально розділяє дані на два класи, забезпечуючи найбільший можливий зазор між класами, що сприяє підвищенню точності класифікації.

На малюнку показано (рисунок 2.5), як SVM виконує класифікацію, визначаючи гіперплощину, що розділяє два класи (жовті кола та сині трикутники). Ключовим аспектом методу є опорні вектори – точки, які найближчі до гіперплощини, і визначають її положення. Завдяки максимізації зазору між цими точками, SVM створює стабільну та надійну модель для класифікації, яка добре узагальнює нові дані.

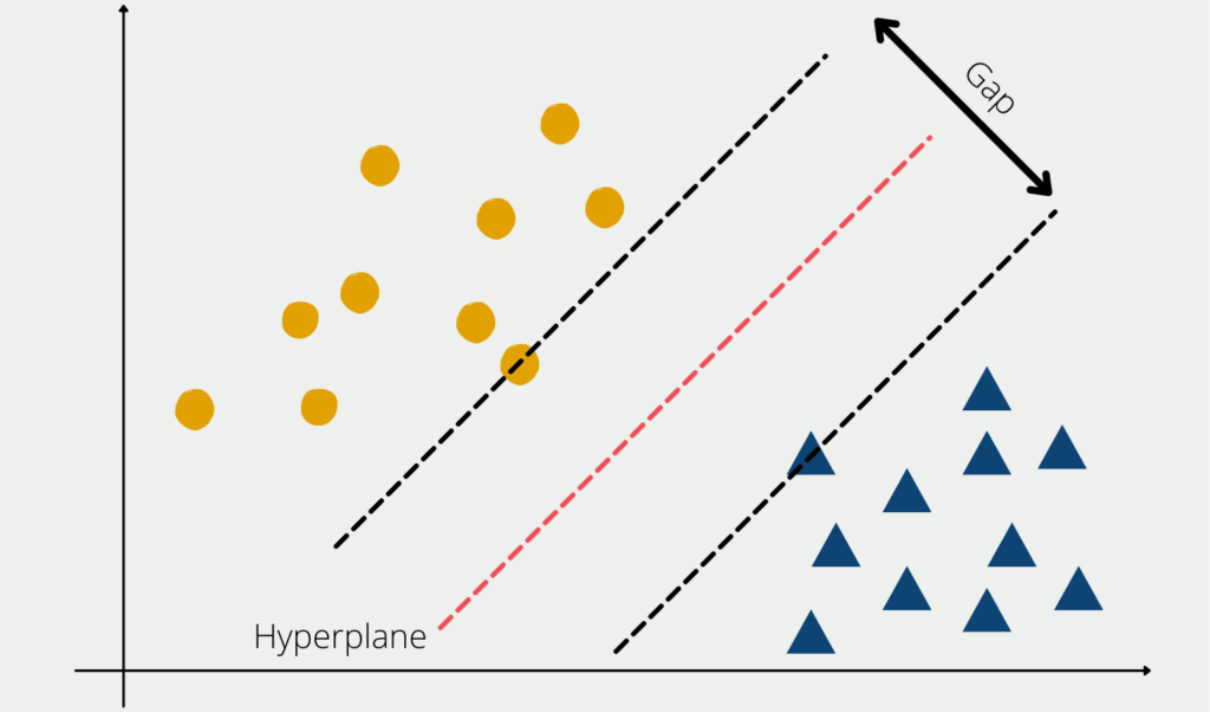

Рисунок 2.5 – Ілюстрація роботи методу опорних векторів

Метод SVM має низку важливих переваг, які роблять його ефективним і надійним у задачах класифікації. По-перше, він демонструє високу продуктивність у високовимірних просторах, що є ключовим фактором при аналізі фінансових транзакцій, які можуть характеризуватися великою кількістю змінних і параметрів. По-друге, SVM вирізняється гнучкістю, адже може використовувати ядрові функції для роботи з нелінійно роздільними даними. Наприклад, за допомогою ядрової функції Радіальної Базисної Функції цей метод здатен ідентифікувати складні та приховані патерни у даних, що особливо важливо у сфері виявлення шахрайства.

Однак, незважаючи на численні переваги, метод SVM має і певні обмеження, які слід враховувати під час його застосування. Одним із основних недоліків є чутливість до масштабування даних, що означає необхідність попередньої нормалізації чи стандартизації вхідних даних для забезпечення коректної роботи моделі. Крім того, метод може бути обчислювально затратним для великих наборів даних, адже процес побудови гіперплощини вимагає значних обчислювальних ресурсів. Це ускладнює його використання в реальних системах моніторингу.

У контексті фінансових систем метод опорних векторів активно застосовується для виявлення аномальних транзакцій, які можуть свідчити про шахрайські дії. Завдяки високій точності класифікації та здатності працювати із складними залежностями між характеристиками даних, SVM виявляється ефективним інструментом навіть у складних умовах.

Загалом, метод опорних векторів є потужним інструментом для аналізу фінансових даних і виявлення шахрайства. Його здатність до адаптації, висока точність і ефективність роботи зі складними патернами даних роблять SVM незамінним у сучасних системах моніторингу фінансових транзакцій, особливо в умовах постійного розвитку шахрайських схем.

2.2.4 Байєсовий метод

Метод Наївного Байєса [8] є одним із фундаментальних підходів у сфері класифікації, який базується на теоремі Байєса. Цей метод передбачає обчислення ймовірності того, що певний об'єкт належить до певного класу, ґрунтуючись на наявних даних про характеристики цього об'єкта.

Особливістю методу є його припущення про незалежність характеристик, що, хоча і є спрощенням, на практиці дозволяє досягти високої ефективності у багатьох задачах класифікації. Зображення ілюструє принцип роботи методу Наївного Байєса (рисунок 2.6), показуючи, як дані можуть бути розподілені між класами на основі обчислення ймовірностей.

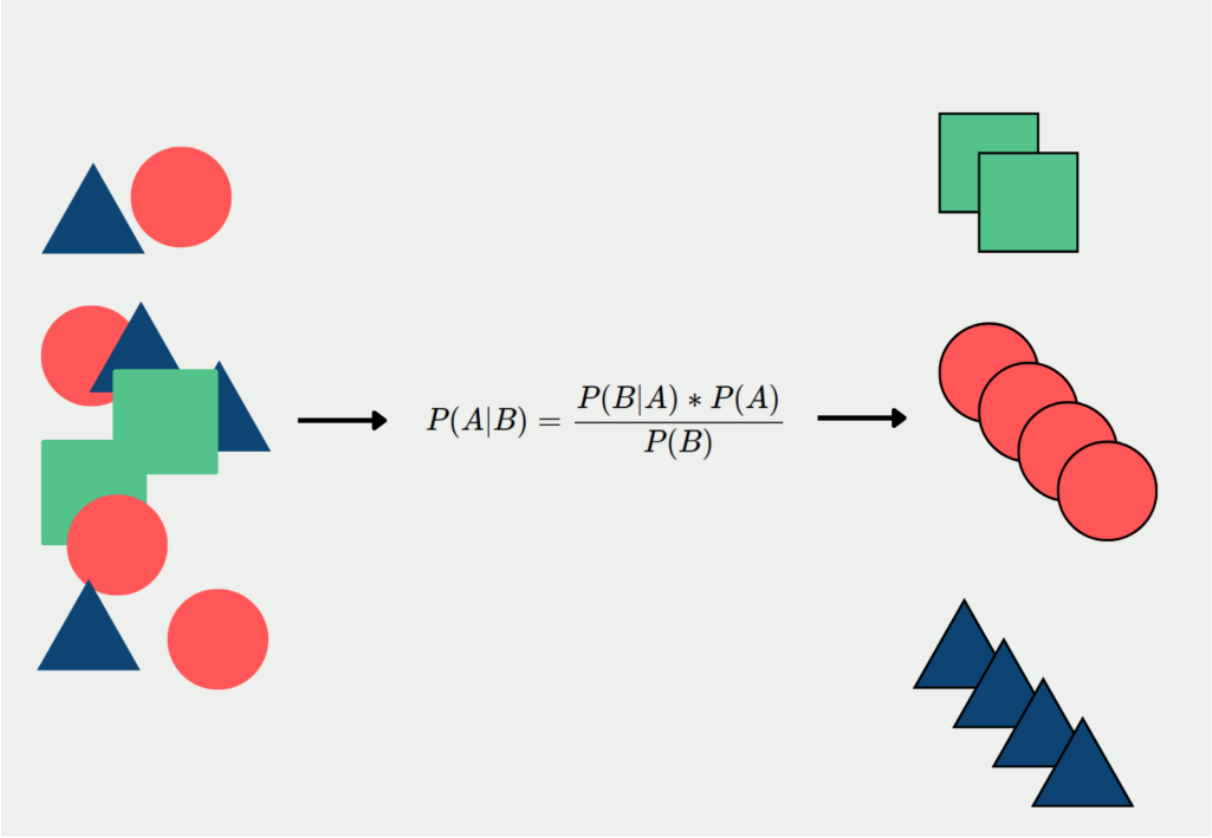

Рисунок 2.6 – Ілюстрація роботи методу Наївного Байєса

Метод Наївного Байєса базується на використанні теореми Байєса, яка дозволяє обчислювати ймовірність належності об'єкта до певного класу на

основі наявних даних про його характеристики. У контексті виявлення шахрайства цей підхід дозволяє класифікувати фінансові транзакції, оцінюючи ймовірність того, що вони є шахрайськими, залежно від таких параметрів, як час здійснення операції, сума транзакції або географічне місце виконання. Цей інструмент забезпечує системам моніторингу можливість використовувати статистичну інформацію для ефективного поділу транзакцій на категорії, ґрунтуючись на їхніх ймовірнісних характеристиках.

Основною перевагою методу Наївного Байєса є його швидкість і обчислювальна простота, що робить його ефективним для роботи з великими наборами даних. Крім того, метод добре працює навіть за умов обмеженої кількості даних для навчання. Це особливо важливо у фінансових системах, де можуть бути доступні лише обмежені дані про шахрайські транзакції.

Однак наївне припущення про незалежність характеристик може стати недоліком у задачах, де характеристики мають сильні кореляції. У таких випадках ефективність методу знижується, і для досягнення кращих результатів може знадобитися застосування більш складних моделей.

У сфері виявлення шахрайства метод Наївного Байєса використовується для аналізу транзакцій, допомагаючи визначати, які з них можуть бути підозрілими. Наприклад, аналіз частоти певних типів транзакцій або аномальних поведінкових патернів може допомогти системі виявляти потенційно шахрайські дії з високою точністю. Зображення, наведене вище, візуалізує, як метод працює з даними, класифікуючи їх на основі ймовірнісних обчислень.

2.2.5 Нейронні мережі

Нейронні мережі є одним із найпотужніших інструментів машинного навчання, який активно застосовується для виявлення фінансового

шахрайства. Завдяки своїй здатності моделювати складні взаємозв’язки між змінними та знаходити приховані патерни, вони стали незамінним методом у задачах класифікації, де традиційні підходи можуть бути недостатньо ефективними.

Основою нейронних мереж є структура, що нагадує біологічний мозок. Мережі складаються з нейронів, організованих у шари: вхідний шар, один або декілька прихованих шарів і вихідний шар. Кожен нейрон отримує сигнали з попереднього шару, обробляє їх і передає результати до наступного шару. Цей процес обробки даних дозволяє мережі поступово виділяти ключові ознаки та закономірності, що є критично важливими для виявлення шахрайства у великих наборах даних.

Особливо ефективними для роботи з фінансовими транзакціями є глибокі нейронні мережі, які включають велику кількість прихованих шарів. Завдяки цьому такі моделі здатні виявляти навіть найскладніші патерни у даних. Наприклад, вони можуть розрізняти шахрайські транзакції, що відрізняються не лише за окремими характеристиками, але й за їхньою загальною поведінковою структурою, такою як нетипова частота операцій чи невідповідність геолокації.

Нейронні мережі також мають значну гнучкість завдяки можливості використовувати різні архітектури, такі як згорткові нейронні мережі (CNN) для аналізу структурованих даних або рекурентні нейронні мережі (RNN) для обробки послідовностей, наприклад, історії транзакцій. Це робить їх універсальним інструментом для різних аспектів виявлення шахрайства.

Попри переваги, нейронні мережі також мають певні обмеження. По-перше, вони вимагають великих обсягів даних для ефективного навчання, що може бути проблемою у випадках обмеженої доступності мічених даних. По-друге, їхня складна структура ускладнює інтерпретацію результатів, що може бути важливим у фінансових системах, де потрібна прозорість у прийнятті рішень. Нарешті, навчання нейронних мереж є обчислювально

складним процесом, що потребує значних ресурсів, таких як GPU або хмарні обчислювальні платформи.

Незважаючи на ці виклики, нейронні мережі продовжують активно застосовуватися у виявленні шахрайства, завдяки їхній здатності адаптуватися до складних і динамічних середовищ. Завдяки цьому вони залишаються одним із провідних методів у боротьбі з фінансовими злочинами та підвищенні безпеки транзакцій.

2.2.6 Порівняння методів класифікації

Порівняння методів класифікації є ключовим етапом у виборі оптимального підходу до виявлення фінансового шахрайства. У контексті розглянутих методів, таких як логістична регресія, Random Forest, метод опорних векторів, байєсові методи та нейронні мережі, важливо оцінити їх ефективність, враховуючи специфіку фінансових даних та характер шахрайських транзакцій.

Логістична регресія є одним із найбільш простих і швидких методів класифікації, що дозволяє отримати прозорі та інтерпретовані результати. Вона добре працює на невеликих наборах даних і підходить для задач, де характеристики транзакцій мають лінійний зв’язок із результатом. Однак у випадках складних нелінійних патернів або великого обсягу даних її ефективність суттєво знижується.

Random Forest забезпечує високу точність класифікації завдяки об’єднанню результатів великої кількості дерев рішень. Метод відзначається стійкістю до перенавчання, а також здатністю працювати з великою кількістю характеристик. Проте його обчислювальна складність може бути проблемою для систем, що працюють у реальному часі.

Метод опорних векторів, завдяки використанню ядрових функцій, дозволяє ефективно працювати зі складними патернами у даних. Його перевагою є здатність до точного розмежування класів навіть у

високовимірних просторах. Водночас, цей метод має високі обчислювальні витрати, особливо на великих наборах даних, що може обмежувати його застосування у системах із великим обсягом транзакцій.

Байєсові методи пропонують швидкість і простоту у реалізації, що робить їх корисними для задач, де доступні малі набори даних. Однак їх ефективність значно залежить від коректності припущень про незалежність характеристик, що часто не відповідає реальним даним у фінансовій сфері.

Нейронні мережі, завдяки своїй здатності моделювати складні та багатовимірні взаємозв'язки, демонструють найвищу ефективність у задачах, де потрібна висока точність. Вони особливо корисні для роботи з великими наборами даних та виявлення складних аномалій. Водночас, їх застосування вимагає значних обчислювальних ресурсів та великих обсягів даних для навчання.

Вибір оптимального методу залежить від конкретних вимог системи, таких як обсяг транзакційних даних, швидкість обробки та точність виявлення шахрайства. Поєднання різних методів, наприклад ансамблевих моделей, може стати ефективним рішенням для забезпечення як високої точності, так і стійкості системи до змін у даних.

2.3 Ансамблеві підходи у класифікації

Ансамблеві підходи у класифікації є важливим напрямом машинного навчання, що дозволяє досягти більш високої точності та надійності моделей. Основна ідея ансамблевих методів полягає у поєднанні результатів кількох базових моделей для покращення загальної продуктивності системи класифікації. На відміну від використання однієї моделі, ансамблеві методи дозволяють враховувати різні підходи до аналізу даних, що робить їх стійкими до помилок окремих алгоритмів. Це особливо актуально у сфері виявлення шахрайства, де класифікація може бути складною через високу варіативність та приховані закономірності у фінансових даних.

Ансамблеві моделі можна умовно поділити на два основні підходи: методи, які зменшують дисперсію, такі як Bagging, і методи, які знижують похибку зміщення, такі як Boosting. Bagging створює кілька незалежних моделей, результати яких об'єднуються, наприклад, через середнє чи голосування, що дозволяє знизити варіативність і забезпечити більш стабільні результати. Boosting, навпаки, спрямований на поступове покращення моделі, навчання якої зосереджується на складних для класифікації прикладах. Ці підходи добре доповнюють один одного і широко застосовуються у фінансових системах для підвищення точності виявлення шахрайства.

Іншим популярним підходом є Stacking, який об'єднує вихідні дані кількох базових моделей за допомогою метамоделі. Це дозволяє враховувати сильні сторони кожного алгоритму, створюючи більш адаптивну систему класифікації. У задачах виявлення шахрайства Stacking може бути ефективним, оскільки він враховує різні аспекти патернів даних і забезпечує гнучкість в умовах мінливих схем шахрайства.

Таким чином, ансамблеві підходи займають центральне місце в сучасному машинному навчанні, пропонуючи потужні інструменти для роботи зі складними та нерівномірними даними. У наступних підрозділах будуть детально розглянуті основні ансамблеві методи, такі як Bagging, Boosting і Stacking, а також їхні переваги й обмеження.

2.3.1 Bagging

Bagging або Bootstrap Aggregating є одним із базових ансамблевих методів у машинному навчанні, що дозволяє підвищити точність та стабільність класифікаційних моделей. Основна ідея Bagging полягає у створенні декількох незалежних моделей на основі різних підвибірок навчальних даних і подальшому об'єднанні їхніх результатів для покращення продуктивності. Цей метод заснований на техніці бутстрепінгу,

коли з вихідного набору даних генеруються випадкові підвибірки із заміною, а кожна модель навчається на одній із цих підвибірок.

Bagging ефективно знижує дисперсію моделей, тобто зменшує їхню чутливість до шуму у навчальних даних. Завдяки цьому ансамблевий підхід стає більш стійким до перевчання, що особливо важливо для складних задач, таких як виявлення шахрайства. У цьому контексті Bagging допомагає забезпечити стабільність класифікаційних моделей, зменшуючи ризик пропуску шахрайських транзакцій через випадкові помилки.

У процесі Bagging кілька базових моделей (найчастіше дерева рішень) навчаються на різних підвибірках даних паралельно, а їхні результати об'єднуються шляхом голосування для задач класифікації або усереднення для задач регресії. Наприклад, якщо кілька дерев рішень у ансамблі класифікують транзакцію як шахрайську, система приймає це рішення на основі більшості голосів. Такий підхід (рисунок 2.7) забезпечує більш точну та надійну роботу моделі у порівнянні з використанням однієї моделі.

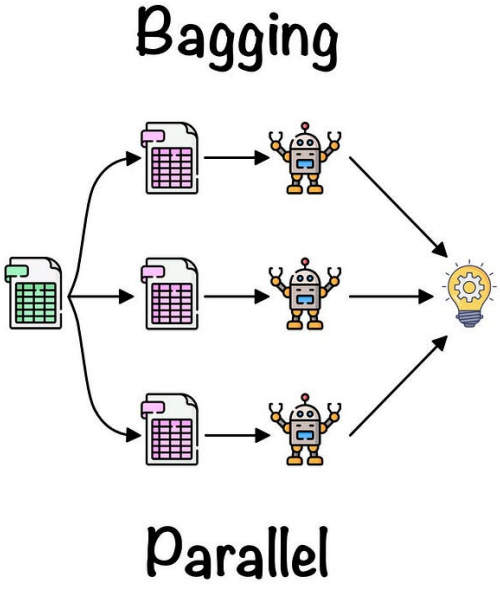

Рисунок 2.7 – Ілюстрація роботи Bagging

Завдяки своїй простоті та ефективності Bagging широко використовується в задачах виявлення шахрайства. Він допомагає знаходити приховані закономірності у даних, знижує ризик хибнопозитивних і хибнонегативних результатів, а також забезпечує адаптивність до мінливих умов у фінансових системах.

2.3.2 Boosting

Boosting є одним із ключових ансамблевих підходів у машинному навчанні, який орієнтований на покращення точності моделі шляхом послідовного навчання слабких моделей (базових алгоритмів) і коригування їхніх помилок. У порівнянні з Bagging, Boosting працює інакше: він навчає моделі послідовно, кожна нова модель фокусується на прикладах, які були неправильно класифіковані попередніми моделями. Основна мета Boosting полягає в створенні сильної моделі на основі комбінації кількох слабких моделей.

Процес Boosting починається з навчання базової моделі на вихідному наборі даних. Далі, на основі результатів цієї моделі, прикладам, які були неправильно класифіковані, призначається більша вага, щоб наступна модель могла краще враховувати ці помилки. Цей процес повторюється кілька разів, створюючи каскад моделей, кожна з яких спрямована на зменшення залишкової похибки попередньої. Остаточне рішення приймається шляхом об'єднання результатів усіх моделей, зазвичай через зважене голосування або сумування прогнозів.

На рисунку (рисунок 2.8) зображено основний принцип роботи цього методу. Базові моделі навчаються послідовно, де кожна наступна модель враховує помилки попередньої. У результаті комбінування прогнозів слабких моделей формується потужна модель, яка забезпечує високу точність класифікації.

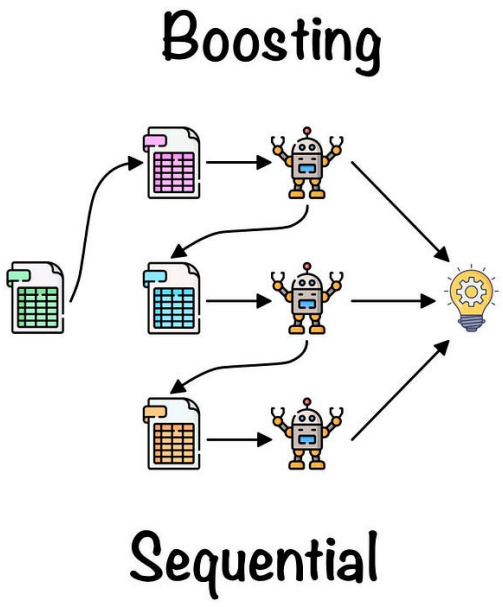

Рисунок 2.8 – Ілюстрація роботи Boosting

Boosting має кілька варіантів реалізації, серед яких найпопулярнішими є AdaBoost, Gradient Boosting і XGBoost. Ці алгоритми знаходять широке застосування у задачах виявлення шахрайства, оскільки вони дозволяють ефективно працювати з великими наборами даних і складними патернами. Зокрема, Gradient Boosting відзначається здатністю до обробки нелінійних залежностей, а XGBoost забезпечує високу продуктивність і адаптивність, що є критично важливим для реальних фінансових систем.

Переваги Boosting включають високу точність класифікації та здатність ефективно працювати з дисбалансом класів, що є частою проблемою у виявленні шахрайства. Однак метод може бути чутливим до шуму в даних і перевчання, якщо базові моделі є надто складними. Незважаючи на це, Boosting залишається одним із найефективніших інструментів для побудови адаптивних і надійних систем моніторингу фінансових транзакцій.

2.3.3 Stacking

Stacking є одним із потужних ансамблевих методів, який об'єднує передбачення декількох базових моделей для створення однієї, більш точної моделі. Основна ідея Stacking полягає в тому, щоб навчити «мета-навчальник» на основі прогнозів базових моделей, об'єднуючи їх результати для отримання кінцевого передбачення. Це дозволяє враховувати сильні сторони різних алгоритмів, підвищуючи загальну точність і стійкість системи класифікації.

Базовий принцип роботи Stacking наступний (рисунок 2.9). Спочатку дані подаються на вхід до кількох різних базових моделей, наприклад, нейронної мережі, методу опорних векторів та дерева рішень. Кожна модель створює свої прогнози, які потім використовуються як новий набір ознак для «мета-навчальника». Цей мета-навчальник, зазвичай реалізований у вигляді простої моделі, наприклад, лінійної регресії, об'єднує результати базових моделей і формує остаточний прогноз.

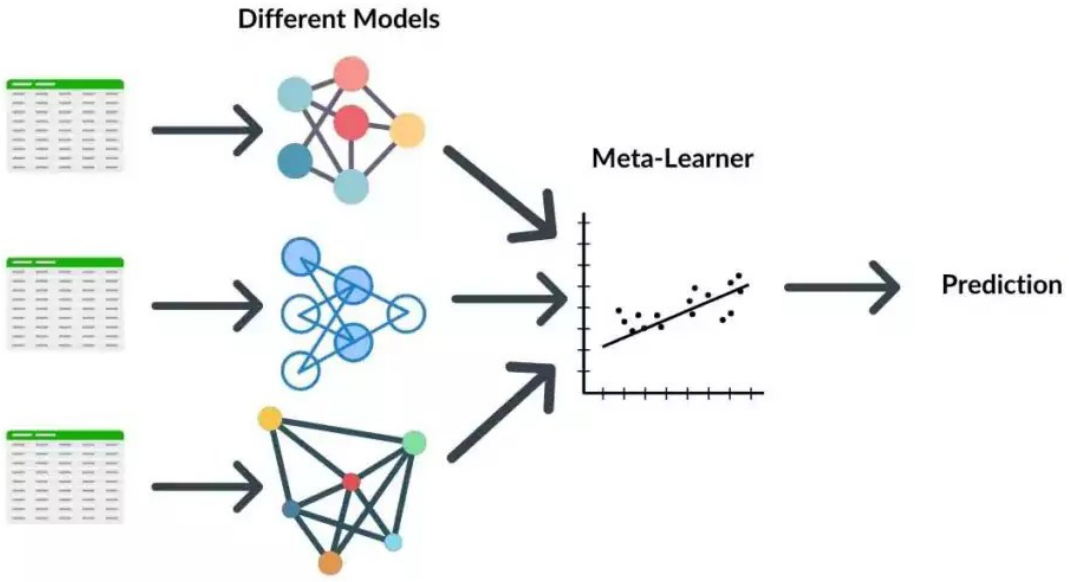

Рисунок 2.9 – Ілюстрація роботи Stacking

Однією з ключових переваг Stacking є його здатність враховувати різноманітність алгоритмів. Наприклад, один алгоритм може бути ефективним для виявлення певних типів шахрайства, тоді як інший краще справляється з іншими патернами. Поєднання їхніх результатів через мета-навчальника дозволяє створити більш гнучку і точну систему.

Stacking є особливо корисним у задачах виявлення шахрайства у фінансових транзакціях, де дані зазвичай мають складну структуру, а різні алгоритми можуть бути ефективними для аналізу різних аспектів даних. Наприклад, нейронні мережі можуть добре працювати із нелінійними залежностями, тоді як методи дерев рішень краще справляються із категоріальними ознаками.

Загалом, Stacking є ефективним інструментом для підвищення точності класифікації у фінансових системах, забезпечуючи більш точне і надійне виявлення шахрайства. Його здатність інтегрувати переваги різних алгоритмів робить його важливим компонентом сучасних систем моніторингу транзакцій.

2.4 Підходи для вирішення класової незбалансованості

Класова незбалансованість є поширеною проблемою у завданнях машинного навчання, зокрема у виявленні шахрайства у фінансових транзакціях. Вона виникає тоді, коли кількість прикладів одного класу, зазвичай нормальних або «чесних» транзакцій, значно перевищує кількість прикладів іншого класу, зазвичай шахрайських транзакцій. Такий дисбаланс може негативно вплинути на точність класифікації, оскільки модель має тенденцію ігнорувати менш представлений клас, зосереджуючись на переважаючому.

Для подолання цієї проблеми існує кілька підходів, які дозволяють збалансувати вибірку, тим самим підвищуючи ефективність роботи алгоритмів класифікації. Одним із підходів є зменшення кількості прикладів

у переважаючому класі, що відоме як Undersampling. Інший підхід полягає у збільшенні кількості прикладів менш представленого класу, який називається Oversampling. Серед сучасних методів особливу увагу привертає техніка Synthetic Minority Oversampling Technique (SMOTE), яка генерує нові синтетичні зразки для менш представленого класу, використовуючи дані, що вже існують.

Ці підходи мають свої переваги та обмеження, які слід враховувати під час вибору методології для вирішення конкретного завдання. У наступних підрозділах буде детально розглянуто кожен із цих методів, включаючи їхні особливості, переваги, недоліки та практичне застосування.

2.4.1 Undersampling

Undersampling є одним із популярних підходів для вирішення проблеми класової незбалансованості у задачах машинного навчання. Цей метод (рисунок 2.10) передбачає зменшення кількості прикладів переважаючого класу, щоб досягти більш рівномірного розподілу між класами у вибірці. У контексті виявлення шахрайства у фінансових транзакціях це може означати зменшення кількості записів про «чесні» транзакції, які зазвичай складають більшість, для забезпечення балансу із записами про шахрайські транзакції.

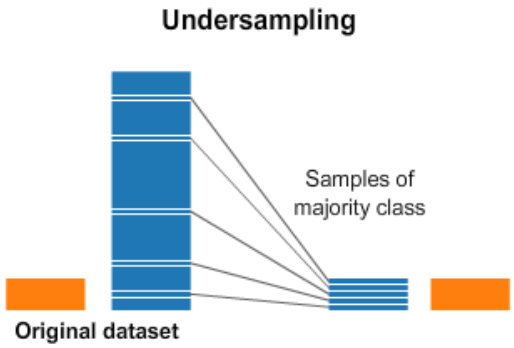

Рисунок 2.10 – Ілюстрація роботи Undersampling

Перевагою цього методу є те, що він зменшує обсяг даних, які необхідно обробляти, що може прискорити навчання моделей машинного навчання. До того ж, завдяки зменшенню розміру вибірки, Undersampling може допомогти уникнути перенавчання на надмірно великих даних переважаючого класу, які домінують у навчальній вибірці.

Однак метод Undersampling також має суттєві недоліки. Основною проблемою є те, що скорочення кількості прикладів переважаючого класу може призводити до втрати важливої інформації, яка могла б сприяти підвищенню точності моделі. У задачах виявлення шахрайства це може бути критичним, оскільки «чесні» транзакції можуть містити важливі патерни, що допомагають моделі краще ідентифікувати аномальні випадки. Крім того, у разі значного дисбалансу класів навіть після Undersampling залишаються виклики, пов'язані із забезпеченням балансу.

На практиці Undersampling часто використовується в комбінації з іншими підходами, такими як Oversampling чи застосування спеціалізованих алгоритмів, які враховують класову незбалансованість. Це дозволяє уникнути втрати ключової інформації та водночас зменшити вплив домінування переважаючого класу. Таким чином, Undersampling є корисним інструментом для підготовки даних, проте його використання вимагає ретельного аналізу та тестування.

2.4.2 Oversampling

Oversampling є підходом (рисунок 2.11) для вирішення проблеми класової незбалансованості, який передбачає збільшення кількості прикладів менш представленого класу, щоб зробити розподіл класів більш рівномірним. У контексті виявлення шахрайства цей метод дозволяє штучно збільшити кількість записів про шахрайські транзакції, які зазвичай становлять лише невелику частку від загальної кількості даних.

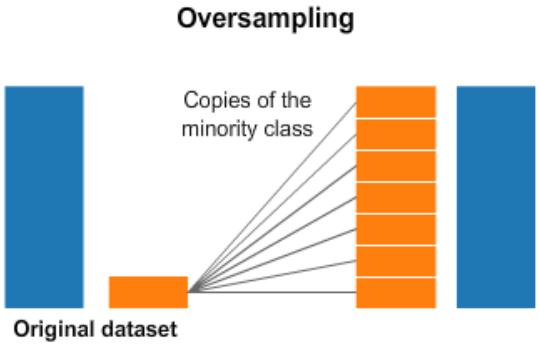

Рисунок 2.11 – Ілюстрація роботи Oversampling

Основна ідея Oversampling полягає у створенні копій існуючих зразків менш представленого класу або генерації нових синтетичних прикладів. Це дозволяє зберегти всі наявні дані та забезпечити баланс між класами без втрати інформації, як це може статися при використанні Undersampling. Наприклад, метод дублювання дозволяє простим чином копіювати зразки шахрайських транзакцій, тоді як більш складні методи, такі як SMOTE, генерують нові зразки шляхом інтерполяції між наявними даними.

Перевагою Oversampling є те, що він дозволяє забезпечити моделі доступ до більшої кількості даних менш представленого класу, що покращує її здатність виявляти аномалії та шахрайські транзакції. Це особливо важливо у фінансових системах, де точність у виявленні шахрайства є критичною. Крім того, цей метод не вимагає видалення жодних даних, що може бути важливим у ситуаціях, коли кожна транзакція містить цінну інформацію.

Однак Oversampling має свої обмеження. Наприклад, дублювання зразків може призводити до перенавчання моделі, коли вона занадто адаптується до конкретних прикладів із навчального набору. Це може знизити її здатність узагальнювати нові дані. Крім того, збільшення розміру даних може збільшити обчислювальні витрати та час навчання моделі.

На практиці Oversampling часто комбінується з іншими підходами, такими як Undersampling або спеціалізовані алгоритми, які враховують

незбалансованість класів. Це дозволяє досягти оптимального результату, зберігаючи баланс між точністю класифікації та ефективністю роботи моделі. У виявленні шахрайства Oversampling є корисним інструментом, який підвищує здатність моделі ідентифікувати аномальні транзакції в умовах значного дисбалансу класів.

2.4.3 SMOTE

SMOTE [9], [10], [11] є передовим методом вирішення проблеми класової незбалансованості, який спрямований на створення нових синтетичних прикладів для менш представленого класу. На відміну від простих підходів до Oversampling, які передбачають дублювання існуючих прикладів, SMOTE використовує алгоритм інтерполяції для створення нових точок даних, розташованих між існуючими зразками. Це дозволяє значно підвищити різноманітність даних у менш представленому класі, зменшуючи ризик перенавчання.

Основною перевагою SMOTE є його здатність уникати дублювання існуючих прикладів, що допомагає зменшити ризик перенавчання моделі. Крім того, цей метод дозволяє зберегти всю корисну інформацію, яка міститься у вихідних даних, водночас покращуючи баланс класів у навчальному наборі. У контексті виявлення шахрайства SMOTE виявляється особливо ефективним, оскільки дозволяє моделі краще розпізнавати аномалії, що характерні для шахрайських транзакцій.

Проте SMOTE має і свої обмеження. Наприклад, він може створювати зразки, які не повністю відображають реальні дані, особливо якщо вихідний набір даних містить значний рівень шуму. Крім того, метод чутливий до вибору параметрів, таких як кількість найближчих сусідів, що може впливати на якість синтетичних прикладів. Також SMOTE може збільшити обчислювальні витрати через зростання розміру навчального набору.

Процес роботи SMOTE (рисунок 2.12) полягає у виборі одного зразка з менш представленого класу та пошуку його найближчих сусідів у багатовимірному просторі ознак. Потім між цими зразками створюється новий синтетичний зразок, розташований на відрізку, що з'єднує їх. Цей підхід дозволяє створювати більш узагальнені дані, які краще відображають розподіл менш представленого класу, зберігаючи при цьому характерні особливості вихідного набору даних.

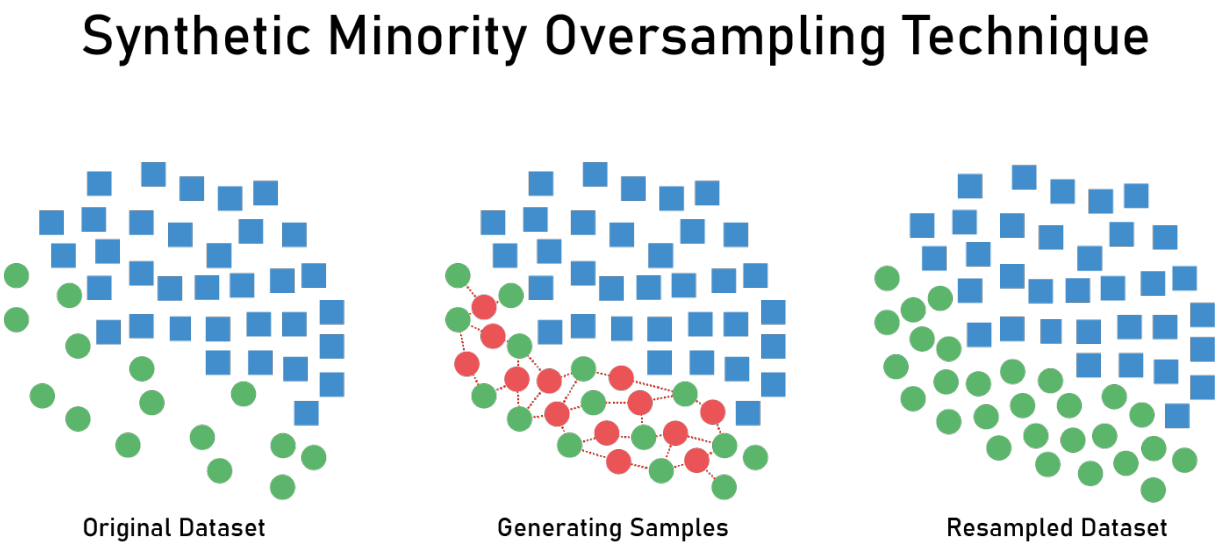

Рисунок 2.12 – Ілюстрація роботи SMOTE

На практиці SMOTE часто комбінується з іншими техніками обробки даних або використовується у поєднанні з алгоритмами класифікації для покращення ефективності моделі. У задачах виявлення шахрайства SMOTE допомагає моделі розпізнавати складні патерни в аномальних транзакціях, забезпечуючи точніше виявлення шахрайських дій навіть за значного дисбалансу класів. Це робить його одним із найпопулярніших методів у сучасних системах боротьби з шахрайством.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Опис застосованих технологій

У цьому підрозділі розглядаються основні технології та інструменти, що використовуються для вирішення задачі виявлення шахрайства у фінансових транзакціях за допомогою методів машинного навчання. Основною мовою програмування є Python, яка завдяки своїй гнучкості та багатофункціональності є оптимальним вибором для обробки великих обсягів даних та побудови складних моделей. Для роботи з даними використовуються бібліотеки Pandas та Seaborn. Pandas дозволяє ефективно маніпулювати таблицями та виконувати операції фільтрації, агрегації і трансформації даних, що значно полегшує процес підготовки даних до аналізу. Seaborn забезпечує можливість візуалізувати дані та результати, допомагаючи виявити закономірності та аномалії.

Для побудови моделей машинного навчання застосовується бібліотека Sklearn, яка містить широкий набір алгоритмів для класифікації, регресії та кластеризації. Sklearn дозволяє не тільки здійснювати налаштування моделей, але й проводити оцінку їхньої ефективності за допомогою таких метрик, як точність, recall, AUC-ROC. Для генерації та обробки синтетичних фінансових даних використовується інструмент PaySim, що дає змогу створювати реалістичні фінансові транзакції для навчання моделей без використання реальних даних. Використання цих технологій дозволяє забезпечити високу ефективність та точність при розв’язанні задачі виявлення шахрайських транзакцій.

3.1.1 Мова програмування Python

Python є основною мовою програмування, що використовується для розробки моделей машинного навчання в даному дослідженні. Завдяки

своїй гнучкості, зручності та великій кількості доступних бібліотек, Python забезпечує ефективну реалізацію алгоритмів для обробки даних, побудови моделей і їх оцінки. Мова має широке застосування у сфері наукових досліджень і машинного навчання завдяки бібліотекам, таким як Pandas, Sklearn та Seaborn, які використовуються для аналізу даних, побудови моделей та візуалізації результатів.

Python також підтримує інтеграцію з іншими технологіями та інструментами, що дозволяє ефективно вирішувати завдання в обробці великих даних та виконанні складних математичних операцій. Завдяки своїй простоті та читабельності коду, Python є зручним інструментом для швидкого прототипування моделей, що є важливим аспектом у задачах, пов'язаних із виявленням шахрайства в фінансових транзакціях.

3.1.2 Бібліотека Sklearn

Sklearn (Scikit-learn) є однією з найбільш популярних бібліотек Python для машинного навчання. Вона надає широкий набір інструментів для виконання задач класифікації, регресії, кластеризації, зниження вимірності та інших методів навчання. У дослідженні ця бібліотека використовується для побудови, налаштування та оцінки ефективності моделей для виявлення шахрайських транзакцій. Sklearn надає зручний інтерфейс для реалізації алгоритмів, таких як логістична регресія, випадковий ліс, метод опорних векторів (SVM), а також підтримує зручні функції для обробки даних, таких як крос-валідація, підбір гіперпараметрів та оцінка моделей.

Однією з основних переваг Sklearn є її простота у використанні, що дозволяє швидко інтегрувати алгоритми машинного навчання в робочий процес. Вона підтримує стандартні функції для масштабування та нормалізації даних, що є критично важливим для покращення ефективності моделей у випадку роботи з нерівномірними або масштабованими даними. Завдяки своїй універсальності та широкому набору алгоритмів, Sklearn є

важливим інструментом для розв'язання задач у сфері виявлення шахрайства.

3.1.3 Бібліотека Seaborn

Seaborn є потужною бібліотекою для візуалізації даних, яка побудована на основі Matplotlib і значно полегшує створення статистичних графіків у Python. Вона надає широкі можливості для візуалізації взаємозв'язків між різними змінними, побудови розподілів та детального аналізу даних. У даному дослідженні Seaborn використовувалася для побудови різноманітних графіків, таких як гістограми, теплові карти, коробкові діаграми та парні графіки, що дозволяють візуалізувати важливі закономірності і аномалії у фінансових транзакціях.

Однією з основних переваг Seaborn є її здатність автоматично оптимізувати зовнішній вигляд графіків і створювати зрозумілі візуалізації з мінімумом коду. Вона інтегрується з Pandas і дозволяє безпосередньо працювати з DataFrame, що спрощує процес підготовки даних для візуалізації. Завдяки Seaborn, у дослідженні вдалося ефективно представити статистичні дані, що допомогло виявити аномалії та взаємозв'язки між змінними, які були важливі для подальшого аналізу шахрайських транзакцій.

3.1.4 Бібліотека Pandas

Pandas є однією з основних бібліотек Python для обробки та аналізу даних. Вона надає потужні інструменти для роботи з таблицями даних, зокрема структуру даних DataFrame, яка дозволяє зручно маніпулювати, очищати, фільтрувати та агрегувати великі обсяги інформації. У цьому дослідженні Pandas використовується для підготовки даних, очищення набору та обробки відсутніх значень, а також для виконання різноманітних

операцій з даними, таких як зміна типів змінних, обчислення статистичних характеристик і агрегування даних.

Однією з основних переваг Pandas є її здатність ефективно працювати з великими наборами даних, що є критично важливим для задач, пов'язаних з фінансовими транзакціями, де обсяги інформації можуть бути значними. Завдяки простому та інтуїтивно зрозумілому синтаксису, Pandas дозволяє швидко обробляти дані та готувати їх до подальшого аналізу або використання в моделях машинного навчання. Вона забезпечує високопродуктивні інструменти для виконання операцій з даними та є важливою частиною в процесі підготовки фінансових транзакцій для побудови моделей виявлення шахрайства.

3.1.5 Симулятор PaySim

PaySim є інструментом для генерації синтетичних фінансових транзакцій, що імітує реальні операції в мобільних фінансових системах. Цей симулятор створює великі набори даних, які містять як шахрайські, так і звичайні транзакції, що дає змогу використовувати їх для тестування алгоритмів машинного навчання без необхідності працювати з конфіденційною інформацією. В основі PaySim лежать реальні дані про транзакції, що були зібрані в мобільних фінансових системах, але вони перетворені в синтетичні для збереження конфіденційності та безпеки.

У дослідженні PaySim використовується для генерації наборів даних, які потім використовуються для навчання і тестування моделей виявлення шахрайства. Оскільки шахрайські транзакції є рідкісними в реальних фінансових даних, генерація синтетичних даних за допомогою PaySim дає змогу забезпечити достатній обсяг шахрайських випадків для тренування моделей, що дозволяє перевірити ефективність алгоритмів у реалістичних умовах. Інструмент надає можливість створення різних сценаріїв транзакцій

і допомагає створювати збалансовані дані для моделювання фінансових шахрайств.

3.2 Опис згенерованого набору даних

У цьому підрозділі описано згенерований набір даних, який використовується для виявлення шахрайства в фінансових транзакціях. Набір даних створено за допомогою інструменту PaySim, який генерує синтетичні транзакції на основі реальних фінансових даних. Ці дані містять як нормальні транзакції, так і шахрайські операції, що дозволяє створити збалансовану модель для виявлення аномалій. Генерація синтетичних даних має важливу перевагу перед використанням реальних фінансових даних, оскільки це забезпечує конфіденційність і дозволяє проводити тестування на широких наборах без ризику порушення прав конфіденційності користувачів.

Набір даних складається з кількох мільйонів записів, кожен з яких містить відомості про одну фінансову транзакцію. Кожен запис має кілька характеристик, які детально описують операцію. Основні змінні включають step, що є цілим числом і вказує на тимчасовий крок, що дорівнює одній годині. Ця змінна є важливою, оскільки вона показує час здійснення транзакції в межах моделювання процесу обробки фінансових операцій. type – це категоріальна змінна, яка визначає тип транзакції, наприклад, CASH_OUT, PAYMENT, TRANSFER тощо, і допомагає класифікувати операції за їхньою природою, що важливо для подальшого аналізу шахрайських операцій. amount є числовою змінною, яка представляє суму транзакції, що може змінюватися в залежності від типу операції та слугує важливим індикатором при виявленні шахрайства. nameOrig – це ідентифікатор відправника транзакції, що дозволяє відстежувати транзакції, здійснені одним користувачем. nameDest є ідентифікатором отримувача транзакції, що, як і для відправника, дає можливість визначити отримувача

операції і може бути використано для виявлення аномалій, пов'язаних із шахрайськими схемами.

oldbalanceOrg – це баланс рахунку відправника до здійснення операції, що є важливою змінною для оцінки того, чи достатньо коштів для виконання транзакції. newbalanceOrg – це баланс рахунку відправника після здійснення операції, і аналіз зміни балансу допомагає виявляти аномалії, особливо якщо зміна балансу не відповідає сумі транзакції. oldbalanceDest вказує на баланс рахунку отримувача до здійснення операції і дозволяє відстежити зміни в балансі отримувача, що може вказувати на можливі аномалії в його рахунку. newbalanceDest є балансом рахунку отримувача після транзакції і, як і для відправника, ця змінна дозволяє відстежувати зміни в балансі отримувача і перевіряти їхню обґрунтованість. Нарешті, isFraud – це цільова змінна, яка позначає, чи є транзакція шахрайською, значення 1, або звичайною, значення 0, і вона є ключовою для задачі виявлення шахрайства, оскільки визначає, чи є транзакція шахрайською.

Крім цих основних змінних, набір даних містить інші ознаки, які можуть використовуватися для побудови моделі виявлення шахрайства. Важливо, що для задачі машинного навчання особливу увагу приділено класовій незбалансованості даних. У фінансових транзакціях шахрайські операції складають лише невелику частину всіх операцій, що робить задачу класифікації складною. Тому, для забезпечення належної якості моделі, дані часто проходять попередню обробку, включаючи балансування класів, щоб зменшити вплив цієї незбалансованості.

Згенерований набір даних також містить великі обсяги транзакцій, що дає змогу застосовувати методи машинного навчання для навчання моделей на реалістичних умовах. Набір даних включає інформацію як для звичайних, так і для шахрайських транзакцій, що дозволяє ефективно тестувати алгоритми на виявлення шахрайства в умовах класової незбалансованості.

Оскільки набір даних синтетичний, він є ідеальним для застосування в дослідженнях, де важлива безпека та конфіденційність даних, а також для

розробки та тестування алгоритмів виявлення шахрайства без ризику використання реальних фінансових даних.

3.3 Аналіз згенерованого набору даних

Аналіз згенерованого набору даних є важливим етапом у розробці моделей для виявлення шахрайства в фінансових транзакціях. Цей процес включає в себе комплексну перевірку даних на наявність аномалій, непослідовностей та інші фактори, які можуть вплинути на ефективність побудованих моделей. Правильний аналіз дозволяє забезпечити високоякісні вхідні дані для машинного навчання, що є основою для побудови точних і надійних алгоритмів. Оскільки фінансові транзакції можуть містити як звичайні операції, так і шахрайські, важливо ретельно вивчити розподіл класів, коректність значень та інші характеристики, щоб мінімізувати ризики помилкових висновків.

Одним з найважливіших аспектів є перевірка типів даних та їх відповідність очікуваним значенням. Невідповідність типів даних може спричинити помилки під час обробки та аналізу, що в свою чергу може призвести до зниження точності моделі. Аналіз змінних дозволяє не тільки виявити потенційні проблеми з даними, але й краще зрозуміти їх взаємозв'язки, що допомагає підготувати дані до подальшого використання в алгоритмах машинного навчання.

Перевірка на відсутні значення є ще одним критичним етапом. Відсутність даних або їх неправильне представлення може серйозно вплинути на результати моделювання. Важливо не лише виявити такі значення, а й визначити методи їх обробки, чи то через заповнення, видалення, чи інші техніки. Аналіз класової незбалансованості даних також є важливим кроком, оскільки велика частка звичайних операцій порівняно з шахрайськими транзакціями може призвести до того, що модель буде

навчатися на неправильних закономірностях, що не дозволяє адекватно виявляти рідкісні шахрайства.

Нарешті, видалення аномальних або непослідовних транзакцій, таких як транзакції з нульовими або від'ємними сумами, є необхідним для забезпечення чистоти та точності даних. Це дозволяє уникнути ситуацій, де аномалії можуть перекосити результат моделювання, і дає змогу зосередитися на реалістичних та коректних операціях для навчання моделі. Аналіз згенерованого набору даних є важливим етапом, що забезпечує основу для побудови високоякісної моделі виявлення шахрайства, здатної працювати в реальних умовах і ефективно обробляти великий обсяг фінансових транзакцій.

3.3.1 Перевірка типів даних

Перевірка типів даних є важливою частиною попереднього етапу обробки даних, оскільки вона забезпечує правильну інтерпретацію інформації та дозволяє уникнути потенційних помилок під час аналізу і побудови моделей машинного навчання. Типи даних визначають, як саме будуть оброблятися та використовуватися змінні в подальшому процесі. Наприклад, числові значення мають бути представлені в правильному форматі для виконання математичних операцій, а категоріальні змінні, такі як класи, повинні бути вірно закодовані для коректної класифікації. Якщо типи даних не відповідають очікуванням, це може призвести до помилок при виконанні операцій або навіть до некоректних результатів у побудові моделі. Тому на цьому етапі важливо перевірити, чи всі стовпці в наборі даних мають відповідний тип, щоб забезпечити точність подальших етапів обробки і моделювання. Для кожної змінної в наборі даних (рисунок 3.1) встановлений відповідний тип.

```
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 3.1 – Типи даних у наборі

Наприклад, змінні, що містять числові значення, такі як суми транзакцій та зміни балансу, мають тип float64, що є оптимальним для роботи з числами з плаваючою комою. Змінні, що містять текстову інформацію, такі як ідентифікатори відправників і отримувачів транзакцій, мають тип object, який використовується для рядкових значень. Крім того, для змінних, що містять цілі числа, таких як крок часу, а також змінні, що позначають шахрайські транзакції, встановлено тип int64, що є необхідним для коректного збереження цілих чисел.

Після вивчення початкових типів даних важливо переконатися, що всі змінні мають правильні типи для подальшої обробки. У лістингу 3.1 надано програмний код для конвертації типів даних. Наприклад, змінна isFraud, що позначає, чи є транзакція шахрайською, спочатку була інтерпретована як ціле число типу int64. Однак оскільки ця змінна є класовою і повинна бути категоріальною, вона була перетворена на тип object за допомогою методу astype('object'). Це важливий етап, оскільки правильна інтерпретація класових змінних необхідна для побудови моделі, яка буде коректно класифікувати транзакції як шахрайські або звичайні. Конвертація типів

даних дозволяє забезпечити коректність подальших етапів аналізу, знижуючи ризик помилок і підвищуючи ефективність машинного навчання.

Лістинг 3.1 – Програмний код, що конвертує зміну isFraud у об’єкт

```
# Convert class variables type to object
data['isFraud'] = data['isFraud'].astype('object')
```

Завдяки правильно виконаній перевірці і конвертації типів даних забезпечується коректність і надійність подальших етапів моделювання, що є критичним для задачі виявлення шахрайства у фінансових транзакціях.

3.3.2 Аналіз змінних у наборі

Аналіз змінних у наборі даних є важливим етапом у підготовці до побудови моделі для виявлення шахрайства. Цей етап допомагає зрозуміти структуру даних, виявити потенційні аномалії, а також підготувати змінні для подальшого використання в алгоритмах машинного навчання. На основі опису статистичних характеристик змінних можна зробити кілька важливих висновків.

Змінна type, що визначає тип транзакції, містить 5 унікальних значень. Найпоширенішим є значення CASH_OUT, яке зустрічається 2 237 500 разів, що свідчить про те, що більшість транзакцій у наборі даних пов’язані з видачею коштів. Це вказує на важливість цього типу операцій при аналізі фінансових транзакцій. Змінні nameOrig та nameDest, що містять ідентифікатори відправників і отримувачів транзакцій, мають значну кількість унікальних значень: 6 353 307 і 2 723 362 відповідно. Це показує, що дані містять велику кількість різних користувачів, що необхідно враховувати при побудові моделі, оскільки ідентифікатори можуть використовуватися для виявлення шахрайських схем.

Змінна amount, що вказує на суму транзакції, є числовою змінною і має великий діапазон значень, що варто враховувати при побудові моделі, оскільки сума може бути важливим індикатором при виявленні шахрайства. Змінна isFraud, яка позначає, чи є транзакція шахрайською, має два можливі значення – 0 (не шахрайська транзакція) та 1 (шахрайська транзакція). Важливою особливістю цієї змінної є значний дисбаланс класів, оскільки кількість звичайних транзакцій (0) значно переважає над кількістю шахрайських (1). Це створює додаткові труднощі при побудові моделі, оскільки модель може схилятися до прогнозування класу, що є домінуючим.

Змінна isFlaggedFraud, що вказує на те, чи була транзакція позначена як шахрайська, також містить два значення – 0 та 1, з більшими частками значення 0. Це свідчить про те, що більшість транзакцій не позначені як шахрайські, що є типовим для фінансових даних, де більшість операцій є звичайними, а шахрайства виявляються рідко.

Вивчення статистичних характеристик змінних у наборі даних дозволяє отримати розуміння їхнього розподілу та взаємозв'язків, що є необхідним для подальшої підготовки даних до моделювання. Такі характеристики, як кількість унікальних значень, найбільш поширені значення та їх частота, дають можливість виявити ключові патерни в даних, які будуть використовуватися для виявлення шахрайських транзакцій.

3.3.3 Перевірка на наявність відсутніх значень у даних

Перевірка на наявність відсутніх значень у даних є важливим етапом у підготовці набору для подальшого аналізу та моделювання. Відсутні значення можуть значно вплинути на якість результатів машинного навчання, оскільки більшість алгоритмів не здатні працювати з пропущеними значеннями без попередньої обробки. У цьому підрозділі описується процес перевірки на наявність відсутніх значень у

згенерованому наборі даних, що є необхідним для забезпечення коректності наступних етапів.

Для перевірки на відсутні значення у наборі даних використовується метод isnull() з бібліотеки Pandas, який дозволяє перевірити кожен стовпчик на наявність пропущених значень. У лістингу 3.2 наведено код, який визначає максимальну кількість пропущених значень у будь-якому з стовпців. Цей код використовує метод sum() для підрахунку відсутніх значень у кожному стовпці та визначає максимальну кількість пропусків серед усіх змінних, що дозволяє швидко оцінити рівень пропущених даних у наборі.

Лістинг 3.2 – Програмний код, що перевіряє наявність відсутніх значень у даних

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

Оскільки максимальна кількість відсутніх значень у будь-якому з стовпців дорівнює 0, можна зробити висновок, що набір даних не містить пропущених значень. Це є позитивним результатом, оскільки відсутність пропусків у даних дозволяє уникнути необхідності застосування методів заповнення або видалення відсутніх значень. Завдяки цьому, набір даних готовий до подальшого аналізу та моделювання без необхідності виконувати додаткові етапи обробки, що зберігає його цілісність і підвищує ефективність наступних етапів роботи.

3.3.4 Класова незбалансованість

Класова незбалансованість є важливим аспектом при роботі з набором даних, особливо в задачах класифікації, де одна з категорій значно

переважає. У випадку з набором даних про фінансові транзакції, класова незбалансованість є суттєвою проблемою, оскільки кількість шахрайських транзакцій значно менша за кількість звичайних операцій. Це може призвести до того, що моделі машинного навчання будуть переважно навчатися на класі, який зустрічається частіше, і не зможуть адекватно виявляти рідкісні, але важливі випадки шахрайства.

У таблиці 3.1 можна побачити, що змінна isFraud, яка позначає, чи є транзакція шахрайською, містить два значення: не шахрайська та 1. Частка транзакцій зі значенням 0 (звичайні операції) значно переважає, оскільки в наборі даних нараховується 6 354 407 таких записів. Водночас лише 8213 транзакцій мають значення 1 (шахрайські), що складає лише 0,13% від загальної кількості. Це підтверджує серйозну класову незбалансованість у даному наборі.

Таблиця 3.1 – Класова незбалансованість у наборі

Тип транзакції	Процент транзакцій
Не шахрайська	99.87
Шахрайська	0.13

Такий сильний дисбаланс може спричинити труднощі в навчанні моделей машинного навчання. Алгоритми можуть схилятися до того, щоб передбачати «негативний» клас (звичайні транзакції) з високою точністю, ігноруючи рідкісні, але важливі шахрайські транзакції. Це може призвести до низької повноти (recall) для класу шахрайських операцій, що є критично важливим для задачі виявлення шахрайства. Для ефективного вирішення цієї проблеми необхідно застосовувати методи боротьби з класовою незбалансованістю, такі як підвищення ваги меншості (шахрайських транзакцій), oversampling або undersampling.

Отже, виявлення і врахування класової незбалансованості є важливим етапом на шляху до побудови ефективних моделей для виявлення

шахрайських транзакцій, оскільки без коректної обробки цього аспекту моделі можуть бути неповними і неефективними в реальних умовах.

3.3.5 Видалення від'ємних та нульових транзакцій

Видалення від'ємних та нульових транзакцій є важливим етапом обробки даних, оскільки такі значення не мають сенсу в контексті фінансових операцій. Транзакції з від'ємними або нульовими сумами можуть виникати через помилки в даних або бути результатом некоректної генерації транзакцій, і тому вони можуть спотворити результати аналізу або моделювання. Для забезпечення коректності подальшої обробки даних необхідно видалити такі записи, оскільки вони не є релевантними для задачі виявлення шахрайства.

У лістингу 3.3 представлено код, що здійснює фільтрацію набору даних. Зокрема, цей код видаляє всі транзакції, сума яких дорівнює нулю. Використовуючи метод loc з бібліотеки Pandas фільтрує дані, залишаючи лише ті транзакції, у яких сума більша за нуль. Це дозволяє ефективно видаляти транзакції з нульовими сумами, які можуть бути результатом помилок у зборі даних або неправильного генерування транзакцій, і які не мають сенсу для подальшого аналізу або моделювання.

Лістинг 3.3 – Програмний код, що видаляє нульові транзакції

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

Видалення нульових транзакцій є необхідним етапом обробки фінансових даних, оскільки такі записи можуть спотворити результати аналізу або побудови моделі. Нульові суми не мають економічної цінності в контексті транзакцій і можуть ввести в оману алгоритми машинного навчання. Очищення набору даних від таких транзакцій допомагає

зменшити шум і підвищити точність моделі, оскільки вона буде навчатися на валідних і коректних даних. Водночас відсутність транзакцій з нульовими сумами дозволяє фокусуватися на реальних фінансових операціях, що позитивно позначається на здатності моделі правильно виявляти шахрайські транзакції.

3.3.6 Видалення неоднозначних транзакцій

Видалення неоднозначних транзакцій є критичним етапом обробки фінансових даних, оскільки такі записи можуть негативно вплинути на точність побудови моделі для виявлення шахрайства. У фінансових транзакціях важливо, щоб всі значення балансу були коректно зафіксовані, оскільки будь-яка невідповідність між початковими та кінцевими балансами може свідчити про помилки в даних або аномалії, що потребують додаткового аналізу. Виявлення таких транзакцій дозволяє очистити набір даних від потенційних артефактів і зосередитись на реальних, коректних операціях.

Згідно з проведеним аналізом, значна частина транзакцій містить неоднозначні або некоректно зафіксовані баланси. Зокрема, 47,23% транзакцій мають початковий баланс відправника, що дорівнює нулю. Це означає, що майже половина всіх операцій починаються з нульового балансу, що є аномалією, оскільки в реальних умовах рахунки відправників зазвичай мають позитивні залишки перед здійсненням операцій. Крім того, лише 0,6% транзакцій мають кінцевий баланс отримувача, рівний нулю, що також є підозрілим і може бути індикатором неправильних даних або шахрайських схем.

Що стосується коректності відображення балансів, то 93,72% транзакцій мають некоректно зафіксований баланс відправника після здійснення операції. Це означає, що у більшості випадків зміна балансу відправника не відповідає очікуваному результату після здійснення

транзакції, що є серйозною аномалією. Аналогічно, 42,09% транзакцій містять помилки в кінцевому балансі отримувача, що вказує на значну кількість транзакцій з неправильними даними про отримувача.

Для забезпечення якості набору даних та уникнення впливу цих аномалій на модель, необхідно видалити всі транзакції з неправильними або неоднозначними балансами. Це дозволяє очистити набір даних від транзакцій, які можуть спотворити результати аналізу, і зосередитись на коректних даних, що важливо для побудови надійної та ефективної моделі для виявлення шахрайських транзакцій. Видалення таких записів є важливим кроком у підготовці даних і підвищує точність моделювання, оскільки забезпечує, що модель працює лише з валідними операціями.

3.4 Аналіз шахрайських транзакцій

Аналіз шахрайських транзакцій є важливим етапом у розробці моделей для виявлення шахрайства в фінансових операціях. Для цього необхідно дослідити різні аспекти транзакцій, зокрема часові кроки, суми транзакцій та зміни балансових показників відправників і отримувачів. Ретельний аналіз цих аспектів дозволяє виявити закономірності, що можуть допомогти у побудові точних і ефективних класифікаційних моделей для виявлення шахрайства.

По-перше, шахрайські транзакції майже рівномірно розподілені по всіх часових кроках, тоді як звичайні транзакції, як правило, мають піки в певні моменти часу. Це може свідчити про різний характер поведінки при здійсненні звичайних та шахрайських операцій.

По-друге, сума транзакцій у шахрайських операціях не має чітко вираженої залежності від інших характеристик і варіюється в межах великих та малих значень. Для звичайних транзакцій можна спостерігати певну тенденцію до більших сум в порівнянні з шахрайськими операціями, хоча ця різниця не є настільки значною.

По-третє, балансові показники можуть служити важливим індикатором для виявлення шахрайства. Наприклад, у звичайних транзакціях початковий баланс відправника часто дорівнює нулю, а кінцевий баланс отримувача має бути позитивним. Однак у шахрайських операціях ці показники можуть мати інші аномальні значення, що може служити сигналом для моделі.

Шахрайські транзакції мають рівномірний розподіл по часових кроках, що вказує на їхнє випадкове виникнення в будь-який момент часу. Це дозволяє припустити, що шахрайські операції можуть відбуватися в будь-який час і не залежать від часу доби чи специфічних періодів. В той же час, звичайні транзакції зазвичай мають піки в певні часові періоди, що свідчить про більш організовану та передбачувану поведінку користувачів.

Такий розподіл може бути важливим для тренування моделей, оскільки він дозволяє виділити цей фактор як потенційний диференціатор між шахрайськими та звичайними транзакціями. Якщо модель буде навчена з урахуванням рівномірного розподілу шахрайських операцій по всіх часових кроках, це може допомогти точно класифікувати транзакції як шахрайські чи звичайні.

Сума транзакції є ще одним важливим аспектом, який варто враховувати при аналізі шахрайських та звичайних операцій. Аналіз розподілу суми транзакцій показує, що шахрайські транзакції не мають чіткої тенденції до вищих чи нижчих сум, що ускладнює використання цієї характеристики для класифікації. В той же час, для звичайних транзакцій спостерігається певна закономірність – сума таких операцій, як правило, є більшою, що дозволяє частково відрізняти їх від шахрайських.

Однак, через варіативність сум у шахрайських транзакціях, не можна зробити однозначний висновок про відмінності між двома категоріями на основі лише цієї характеристики. Це підкреслює важливість комплексного підходу до аналізу, де сума транзакції є лише одним з багатьох параметрів, що визначають класифікацію.

Важливу роль в аналізі шахрайських транзакцій відіграють балансові показники відправників і отримувачів. Як показано у лістингу 3.4 для виявлення можливих шахрайських операцій використовуються неточності в балансах відправника і отримувача. Зокрема, для відправника розраховується різниця між старим балансом та новим балансом після здійснення операції, враховуючи суму транзакції. Якщо ця різниця значно відрізняється від очікуваної, це може вказувати на можливі помилки або маніпуляції.

Лістинг 3.4 – Програмний код, що визначає неточності балансу для відправника, так і для одержувача

```
    # Defining inaccuracies in originator and recipient balances
    data['origBalance_inacc'] = (data['oldbalanceOrg'] - data['amount']) - data['newbalanceOrig']
    data['destBalance_inacc'] = (data['oldbalanceDest'] + data['amount']) - data['newbalanceDest']
```

У результаті цього кроку отримуються нові змінні, які вказують на неточності в балансах відправника та отримувача. Шахрайські транзакції мають більші й позитивні неточності в балансі одержувача (як правило), на відміну від звичайних транзакцій, де ці неточності можуть бути негативними через помилки в обробці або недостовірні дані, що зустрічається у реальному житті.

Згідно з проведеним аналізом, у шахрайських транзакціях початковий баланс відправника є нульовим лише 0,3% часу, у порівнянні з 47% випадків у звичайних транзакціях. Це може служити важливим показником для побудови класифікаційної моделі, оскільки така суттєва різниця в поведінці балансу відправника є чітким диференціатором між двома категоріями транзакцій.

3.5 Підготовка даних для побудови моделі

Підготовка даних для побудови моделі є важливим етапом у розробці будь-якої системи машинного навчання, оскільки якість вхідних даних безпосередньо впливає на ефективність і точність моделі. На цьому етапі здійснюється обробка сирих даних, що включає в себе видалення непотрібних змінних, кодування категоріальних змінних, нормалізацію даних і розподіл їх на тренувальні та тестові набори. Це дозволяє привести дані до формату, який підходить для використання в алгоритмах машинного навчання, а також мінімізувати потенційні помилки і неточності, які можуть виникнути в процесі навчання моделі. Правильна підготовка даних допомагає зберегти важливу інформацію і оптимізує процес навчання, що призводить до підвищення точності та надійності результатів.

Кожен із підпунктів підготовки даних має важливе значення для забезпечення коректності моделі. Видалення непотрібних змінних дозволяє зменшити розмірність набору даних, що полегшує навчання моделі, а також знижує ризик перенавчання. Кодування категоріальних змінних необхідне для того, щоб алгоритми машинного навчання могли коректно працювати з такими даними. Нормалізація даних дозволяє привести числові ознаки до спільного масштабу, що важливо для багатьох моделей, оскільки без цього дані з великими значеннями можуть домінувати в процесі навчання. Розподіл даних на тренувальний та тестовий набори дозволяє оцінити ефективність моделі на нових, невідомих даних, що є критично важливим для перевірки її узагальнюючої здатності.

3.5.1 Видалення непотрібних змінних

Видалення непотрібних змінних є важливим етапом підготовки даних для побудови моделі машинного навчання. У процесі обробки даних часто виникає необхідність усунути змінні, які не несуть корисної інформації для

вирішення поставленої задачі або які можуть призвести до зайвих обчислювальних витрат та складності моделі. У даному випадку було прийнято рішення видалити змінні nameOrig та nameDest, оскільки вони містять ідентифікатори відправника та отримувача транзакцій. Ці змінні не впливають на виявлення шахрайства, оскільки не несуть прямої інформації про характер транзакції чи її суму

Нижче наведено код у лістингу 3.5, де виконується видалення цих змінних за допомогою методу drop() з бібліотеки Pandas. Код видаляє стовпці nameOrig і nameDest з набору даних, де параметр axis = 1 вказує на те, що потрібно видалити стовпці, а не рядки. Цей крок дозволяє зменшити розмірність даних, зберігаючи лише ті змінні, які мають значення для моделювання і безпосередньо впливають на класифікацію шахрайських транзакцій.

Лістинг 3.5 – Програмний код, що видаляє ім’я відправника та одержувача

```
# Removing name columns
data = data.drop(['nameOrig', 'nameDest'], axis=1)
```

3.5.2 Кодування категоріальних змінних

Кодування категоріальних змінних є важливим етапом підготовки даних для машинного навчання. Більшість алгоритмів не можуть безпосередньо працювати з категоріальними змінними, оскільки вони призначені для обробки числових значень. Тому категоріальні змінні повинні бути перетворені на числовий формат, щоб моделі могли коректно працювати з ними. Один із найбільш поширених методів кодування категоріальних змінних – це one-hot encoding, який створює нові змінні для кожного унікального значення категоріальної змінної, де для кожної категорії присвоюється бінарне значення (0 або 1).

У лістингу 3.6 показано, як застосовувати one-hot encoding до змінної type, яка вказує на тип транзакції. За допомогою методу pd.get_dummies() створюються нові бінарні змінні для кожного унікального значення змінної type. Це дозволяє перетворити категоріальну змінну на числові значення, що можна використовувати в моделях машинного навчання. Метод get_dummies() автоматично створює стовпці для кожного типу транзакції та заповнює їх значеннями 0 або 1 в залежності від того, чи належить конкретна транзакція до цього типу.

Лістинг 3.6 – Програмний код, що кодує категоріальну зміну

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

Застосування цього методу кодування дозволяє ефективно інтегрувати категоріальні дані у моделі машинного навчання, забезпечуючи їх коректну обробку. Важливість такого кроку полягає в тому, що алгоритми навчання можуть працювати з такими змінними без втрати інформації та зберігаючи їх значення для класифікації транзакцій. В результаті отримуються числові представлення категоріальних змінних, що покращує здатність моделі виявляти закономірності та робити точні прогнози.

3.5.3 Нормалізація даних

Нормалізація даних є важливим етапом у підготовці набору даних для машинного навчання, особливо коли використовуються алгоритми, чутливі до масштабу змінних. Деякі моделі, зокрема, методи, засновані на відстанях, наприклад, метод опорних векторів або k-найближчі сусіди, можуть бути значно спотворені через різницю в масштабах змінних. Нормалізація дозволяє привести всі числові ознаки до одного масштабу, зазвичай у

діапазон [0, 1] або зі стандартним відхиленням, що дозволяє моделі працювати більш ефективно і знижує ймовірність перенавчання.

У лістингу 3.7 продемонстровано, як нормалізувати числові ознаки в наборі даних за допомогою StandardScaler з бібліотеки Scikit-learn. У цьому коді спочатку створюється об'єкт std_scaler, який виконує стандартизацію даних. За допомогою методу fit_transform() дані нормалізуються, і результат зберігається в новому датафреймі data_scaled. Важливо, що після нормалізації набір даних містить ті самі стовпці, що й оригінальний набір, за винятком змінної isFraud, яку залишають без змін, оскільки вона є цільовою змінною і не потребує нормалізації.

Лістинг 3.7 – Програмний код, що нормалізує набір даних

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

Нормалізація даних є необхідним кроком, оскільки вона дозволяє усунути вплив різних масштабів числових змінних на процес навчання моделей. Це забезпечує більш збалансоване представлення всіх ознак, дозволяючи моделі коректно оцінювати важливість кожної змінної. Завдяки цьому етапу моделі можуть навчатися швидше та з більшою точністю, особливо коли дані мають великий діапазон значень.

3.5.4 Розподіл даних на тренувальний та тестовий набори

Розподіл даних на тренувальний та тестовий набори є критично важливим етапом у процесі побудови моделей машинного навчання. Цей

крок дозволяє перевірити здатність моделі узагальнювати на нових, невідомих даних, що є важливим для оцінки її ефективності. Тренувальний набір використовується для навчання моделі, тоді як тестовий набір необхідний для перевірки її точності та здатності робити прогнози на реальних даних, з якими модель не стикалася під час навчання. Стандартною практикою є розподіл даних на тренувальний набір, як правило, 70-80% від всіх даних, і тестовий набір 20-30%.

У лістингу 3.8 показано, як здійснюється цей процес за допомогою функції train_test_split() з бібліотеки Scikit-learn. Спочатку дані розподіляються на дві групи: X (ознаки) та y (цільова змінна isFraud). Після цього дані розділяються на тренувальний та тестовий набори за допомогою функції train_test_split(), де 30% даних використовуються для тестування, параметр test_size=0.3, а решта 70% – для навчання моделі. Крім того, застосовується LabelEncoder для кодування цільової змінної isFraud, щоб перетворити її в числові значення, оскільки багато моделей машинного навчання не працюють з категоріальними значеннями без попереднього перетворення.

Лістинг 3.8 – Програмний код, що розподіляє дані на тренувальний та тестовий набори

```
    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())
```

Правильний розподіл даних на тренувальний та тестовий набори дозволяє перевірити, наскільки добре модель може працювати з новими даними, і оцінити її ефективність без ризику перенавчання. Цей етап є важливим для оцінки реальної здатності моделі до узагальнення, що є критичним для її застосування в реальних умовах, де дані можуть відрізнятися від тих, що використовувалися для тренування.

3.6 Опис побудови та навчання моделей класифікації

У цьому підрозділі описано процес побудови та навчання двох моделей класифікації, які використовуються для виявлення шахрайства у фінансових транзакціях. Були використані для цього завдання наступні моделі – логістична регресія та Random Forest. Обидві ці моделі є популярними інструментами в задачах класифікації, і кожна з них має свої переваги в залежності від характеристик даних і вимог до точності. Кожна з цих моделей була побудована та навчена з використанням стандартних підходів та бібліотек Python, зокрема, scikit-learn, що забезпечує зручний інтерфейс для роботи з алгоритмами машинного навчання.

Перше, що потрібно зробити для побудови моделі, це визначити модель і налаштувати параметри для навчання. У лістингу 3.9 показано, як відбувається ініціалізація двох моделей: для логістичної регресії використовується клас LogisticRegression(), а для Random Forest – клас RandomForestClassifier(). Це базові класифікатори, які добре підходять для задачі класифікації, де потрібно розподілити транзакції на шахрайські та звичайні. Логістична регресія є лінійною моделлю, яка добре працює при наявності лінійних залежностей між ознаками, а Random Forest є ансамблевим методом, який побудований на основі дерева рішень і є дуже потужним у випадках, коли дані містять складні взаємозв’язки.

Лістинг 3.9 – Ініціалізація моделей для класифікації та налаштування крос-валідації

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

Для обох моделей налаштовується перехресна валідація через клас StratifiedKFold, що дозволяє розбити набір даних на 5 частин (фолдів). Використання стратифікованої крос-валідації гарантує, що кожен фолд містить пропорційне співвідношення між класами (шахрайські та звичайні транзакції), що важливо при класовій незбалансованості, де одна категорія, звичайні транзакції, значно переважає. Це дозволяє моделі вчитися на більш збалансованих даних, що сприяє кращому узагальненню. Параметр scr вказує на використання метрики recall (повнота) для оцінки моделі під час крос-валідації, що є важливим, оскільки ми хочемо максимально виявляти шахрайські транзакції.

Задача побудови моделі класифікації є складною через високу класову незбалансованість та необхідність забезпечення точності виявлення рідкісних шахрайських транзакцій. Тому важливою частиною цього етапу є правильне налаштування крос-валідації і вибір відповідної метрики для оцінки результатів.

У нашому випадку, для оцінки ефективності моделей вибрано метрику recall (повнота), оскільки вона дозволяє максимально точно виявляти шахрайські транзакції, що є основною метою цього дослідження. На цьому етапі були побудовані дві моделі: логістична регресія та Random Forest, які будуть оцінюватися за допомогою крос-валідації, і результати їхнього порівняння будуть представлені в наступному розділі, що стосується експериментальних досліджень.

3.6.1 Логістична регресія

Логістична регресія є одним з найпоширеніших методів класифікації, який використовується для прогнозування ймовірності належності спостереження до одного з двох класів. Вона є лінійною моделлю, яка намагається знайти таку гіперплощину в простору ознак, що оптимально розділяє два класи, використовуючи логарифмічну функцію для оцінки ймовірності. Логістична регресія є дуже ефективною для задач бінарної класифікації, де класові мітки є дискретними (наприклад, шахрайство чи не шахрайство), і має декілька переваг, зокрема простоту в реалізації та інтерпретації результатів. Однак вона може показати не найкращі результати на складних, нелінійних даних, де зв'язки між ознаками і класами є складнішими.

Для оцінки моделі логістичної регресії використовувалася перехресна валідація, що дозволяє отримати більш точні результати, оскільки модель тестується на різних підмножинах даних. У лістингу 3.10 показано, як здійснюється оцінка моделі за допомогою функції cross_val_score. Ця функція виконує перехресну валідацію на навчальних даних, розбитих на 5 фолдів за допомогою StratifiedKFold. Кожен фолд використовується для тестування моделі, при цьому навчання відбувається на решті даних. Параметр scoring був заданий як recall, що дозволяє оцінити здатність моделі виявляти шахрайські транзакції, оскільки саме ця метрика є критично важливою для задачі виявлення рідкісних подій, таких як шахрайства.

Лістинг 3.10 – Оцінка моделі логістичної регресії за допомогою перехресної валідації

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

Для оцінки ефективності моделі було обрано метрику recall (повнота), оскільки важливо виявляти якомога більше шахрайських транзакцій. Застосування перехресної валідації дозволяє отримати не тільки точність моделі, а й оцінити її стабільність на різних частинах даних.

Таким чином, після виконання крос-валідації ми отримуємо значення recall для кожного з фолдів, що дозволяє оцінити стабільність і ефективність моделі на різних частинах навчальних даних. Застосування перехресної валідації дає змогу уникнути проблеми перенавчання, оскільки модель оцінюється на всіх даних, а не лише на тренувальному наборі. Отримані результати будуть використані для подальшого аналізу і порівняння ефективності моделі з іншими методами класифікації, такими як Random Forest, у наступному розділі.

3.6.2 Random Forest

Текст Random Forest – це ансамблевий метод, що використовує декілька дерев рішень для здійснення класифікації або регресії. Кожне дерево приймає рішення на основі підмножини ознак та випадкових зразків даних, що дозволяє зменшити ризик перенавчання, властивий окремим деревам. Random Forest є потужним методом для задач класифікації, особливо коли є складні взаємозв'язки між ознаками, або коли дані містять шум. Однією з основних переваг Random Forest є його здатність добре працювати навіть з високою кількістю ознак та обробляти великий обсяг даних. Він також не вимагає великих зусиль для налаштування гіперпараметрів, оскільки модель автоматично визначає важливість ознак та мінімізує помилки. В результаті Random Forest є одним з найбільш використовуваних методів у практиці машинного навчання.

Оцінка моделі Random Forest здійснювалася за допомогою перехресної валідації, що дозволяє ефективно перевірити її здатність до узагальнення. У лістингу 3.11 показано, як проводиться оцінка моделі

Random Forest на навчальних даних, використовуючи функцію cross_val_score з бібліотеки scikit-learn. Ця функція автоматично виконує перехресну валідацію, розбиваючи дані на 5 фолдів, при цьому кожен фолд використовується для тестування, а решта – для навчання моделі. Важливою частиною цього процесу є вибір метрики для оцінки моделі, і в даному випадку було обрано recall (повнота), що дозволяє оцінити здатність моделі правильно виявляти шахрайські транзакції, які є рідкісними і важливими для розв'язуваної задачі.

Функція cross_val_score повертає оцінки повноти для кожного з фолдів, що дозволяє отримати загальну картину ефективності моделі. Використання стратифікованої перехресної валідації зберігає пропорцію класів у кожному з фолдів, що важливо при класовій незбалансованості. Таким чином, модель Random Forest оцінюється на всіх даних, що дозволяє забезпечити більш точні й стабільні результати, ніж при використанні лише одного набору для тренування та тестування.

Лістинг 3.11 – Оцінка моделі Random Forest за допомогою перехресної валідації

```
sc_rf = cross_val_score(model_rf, X_train_original, y_train_original, cv=skf,scoring=scr)
```

Цей підхід дає змогу уникнути проблеми перенавчання і оцінити, наскільки модель здатна працювати з новими, невідомими даними. Після виконання перехресної валідації отримані результати дають змогу порівняти ефективність Random Forest з іншими методами, такими як логістична регресія, що буде зроблено в наступному розділі.

4 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

4.1 План експериментів

У цьому розділі представлено експериментальне дослідження побудованих моделей машинного навчання, спрямоване на оцінку їх ефективності у виявленні шахрайських транзакцій. Моделі логістичної регресії та Random Forest були попередньо реалізовані на основі підготовленого датасету, який містить транзакції з мітками про наявність шахрайства. Подальше експериментальне дослідження включає оцінку якості класифікації моделей із застосуванням стратифікованої крос-валідації, а також перевірку їх стійкості до проблеми класової незбалансованості.

Основна мета експериментів полягає у визначенні продуктивності кожної моделі як до, так і після обробки дисбалансу класів, з подальшим тюнінгом гіперпараметрів для досягнення оптимальних результатів. Особливу увагу буде приділено ключовим метрикам, таким як точність, повнота, F1-міра, ROC AUC та аналізу важливості ознак, що дозволить зробити обґрунтовані висновки щодо придатності моделей до використання в системах реального часу.

4.2 Оцінка продуктивності моделей

Оцінка продуктивності моделей є ключовим етапом експериментального дослідження, оскільки саме вона дозволяє зробити обґрунтовані висновки щодо ефективності кожного з алгоритмів у контексті задачі виявлення шахрайських транзакцій. На цьому етапі аналізується здатність моделей правильно класифікувати як шахрайські, так і звичайні операції, з використанням основних метрик якості: точності (Precision) та повноти (Recall). Для забезпечення об’єктивності результатів було

застосовано стратифіковану крос-валідацію з п’ятьма фолдами, яка дозволяє зберігати співвідношення класів у кожному підмножині даних, що особливо важливо в умовах вираженого класового дисбалансу.

Першою була протестована модель логістичної регресії, яка є базовим підходом до бінарної класифікації. Як видно з таблиці 4.1, ця модель досягає високої точності на тренувальних даних – 91.03%, але демонструє значно нижчу повноту – лише 50.88%. Це означає, що хоча більшість передбачень моделі були правильними, вона не змогла ідентифікувати велику частину шахрайських операцій, що вкрай критично для задачі виявлення шахрайства. На тестових даних ситуація подібна: точність залишається на високому рівні (90.12%), але повнота становить лише 51.67%, що вказує на низьку чутливість до шахрайського класу. Ці результати свідчать про те, що модель логістичної регресії переважно орієнтується на основний, більший клас – звичайні транзакції – і не здатна ефективно виявляти рідкісні випадки шахрайства.

Таблиця 4.1 – Порівняння результатів Logistic Regression та Random Forest

Модель	Precision (Train)	Recall (Train)	Precision (Test)	Recall (Test)
Logistic Regression	91.03%	50.88%	90.12%	51.67%
Random Forest	100%	99.84%	100%	99.79%

На відміну від цього, модель Random Forest продемонструвала майже ідеальні результати як на тренувальних, так і на тестових даних. Точність на тренувальній вибірці склала 100%, а повнота – 99.84%. Це означає, що модель змогла коректно класифікувати практично всі транзакції, включаючи майже всі шахрайські. На тестових даних результати залишаються на тому ж високому рівні: 100% точності та 99.79% повноти. Така висока продуктивність моделі свідчить про її здатність ефективно

навчатися навіть в умовах класової незбалансованості та складності виявлення аномалій. З огляду на ці результати можна зробити висновок, що ансамблевий підхід Random Forest краще адаптується до даних та здатен виявляти шахрайство значно точніше, ніж логістична регресія.

Загалом, оцінка моделей засвідчила значну перевагу Random Forest над логістичною регресією за всіма ключовими метриками. Особливо важливою є саме повнота – здатність моделі виявляти якомога більше шахрайських транзакцій, не залишаючи їх непоміченими. У реальних фінансових системах така здатність має вирішальне значення, адже невиявлені шахрайські операції можуть призвести до серйозних втрат.

Отже, на цьому етапі експериментального дослідження можна впевнено стверджувати, що Random Forest є більш надійним інструментом для побудови систем виявлення шахрайства, особливо в умовах, наближених до практичних.

4.3 Вирішення проблеми класового дисбалансу

У процесі побудови моделей машинного навчання для виявлення фінансового шахрайства ключовою проблемою є значна класова незбалансованість даних. У використаному наборі транзакцій шахрайські операції становлять лише незначну частку від загальної кількості записів, тоді як звичайні транзакції домінують. Такий розподіл класів призводить до того, що більшість моделей схильні ігнорувати рідкісний клас і концентруватися на правильній класифікації основного, що формально покращує точність, але критично знижує здатність виявляти аномальні транзакції. Щоб усунути цей ефект і забезпечити моделі можливість ефективного навчання на обох класах, було вирішено застосувати техніку undersampling.

Метод undersampling полягає у зменшенні кількості зразків основного класу до рівня кількості зразків меншості (шахрайського класу). Як

показано у лістингу 4.1, спочатку було виділено індекси всіх шахрайських транзакцій у тренувальному наборі, після чого за допомогою функції np.random.choice випадковим чином вибрано таку саму кількість записів із класу звичайних транзакцій. Ці два масиви індексів були об’єднані, сформувавши нову збалансовану вибірку, яка згодом використовувалась для навчання моделей.

Лістинг 4.1 – Програмний код, який збалансовує тренувальну вибірку методом undersampling

```
# Undersampling the training dataset
fraud_indices_train = np.where(y_train_original == 1)[0]
non_fraud_indices_train = np.where(y_train_original == 0)[0]
undersample_non_fraud_indices_train =
np.random.choice(non_fraud_indices_train,len(fraud_indices_train),replace = False)
undersample_non_fraud_indices_train =
np.array(undersample_non_fraud_indices_train)
undersample_indices_train =
np.concatenate([fraud_indices_train,undersample_non_fraud_indices_train])
X_train_undersample =
X_train_original.loc[X_train_original.reset_index(drop=True).index.isin(undersample_indices_train),:]
y_train_undersample = y_train_original[undersample_indices_train.tolist()]
```

Після застосування методу undersampling було сформовано тренувальний набір, що містив 11526 записів. Така кількість обумовлена тим, що саме стільки прикладів шахрайських транзакцій було наявно у вихідному наборі.

4.4 Налаштування гіперпараметрів моделей

Після формування збалансованого тренувального набору методом undersampling було виконано тюнінг гіперпараметрів для логістичної регресії з метою пошуку найкращої конфігурації моделі. Як відомо, логістична регресія має два основних параметри, що підлягають налаштуванню: тип регуляризації чи параметр penalty, де l1 означає Lasso, а l2 – Ridge, та коефіцієнт регуляризації, параметр C, обернений до сили штрафу. У межах дослідження було протестовано різні комбінації цих параметрів.

Найкращий результат для логістичної регресії після застосування undersampling був досягнутий при використанні штрафу типу l1 та значенні C = 100, що забезпечило recall на рівні 44.16%. Інші конфігурації показували схожі або гірші значення. Навіть при найкращому налаштуванні, модель логістичної регресії не змогла подолати поріг у 50% для метрики recall, що вказує на її обмеженість у виявленні шахрайських транзакцій. У той час як Random Forest, навіть без додаткового тюнінгу, продемонстрував recall, близький до 100%. Це підкреслює перевагу ансамблевого підходу в контексті аналізу фінансових транзакцій і підтверджує його доцільність як базової моделі для таких задач.

4.5 Результати експериментів

Після завершення порівняльного аналізу моделей та проведення серії експериментів з обробкою класового дисбалансу й тюнінгом гіперпараметрів, було визначено найкращу модель для задачі виявлення шахрайських транзакцій – Random Forest. На відміну від логістичної регресії, яка демонструвала обмежені показники recall навіть після налаштування регуляризації, модель Random Forest забезпечила стабільно високі результати без глибокого налаштування, що свідчить про її природну

здатність ефективно працювати з незбалансованими та складними даними. У рамках експерименту були підібрані оптимальні параметри кількості дерев (n_estimators) та максимальної глибини (max_depth), що дозволило досягти повноти класифікації на рівні 99.79% і точності 100%. Це підтверджує, що модель змогла практично безпомилково класифікувати як шахрайські, так і звичайні транзакції.

Для кращого розуміння прийняття рішень моделлю Random Forest було проаналізовано важливість вхідних ознак (рисунок 4.1), що дозволило оцінити вплив кожної змінної на прийняття рішень. Найбільш впливовою ознакою виявилась newbalanceOrig, яка мала найвищу відносну важливість серед усіх ознак, тобто баланс відправника після транзакції, що є логічним маркером потенційно шахрайських операцій. Також високий вплив мають oldbalanceOrg (початковий баланс відправника) та amount (сума транзакції). Це свідчить про те, що фінансові характеристики учасників транзакції мають вирішальне значення для виявлення аномалій.

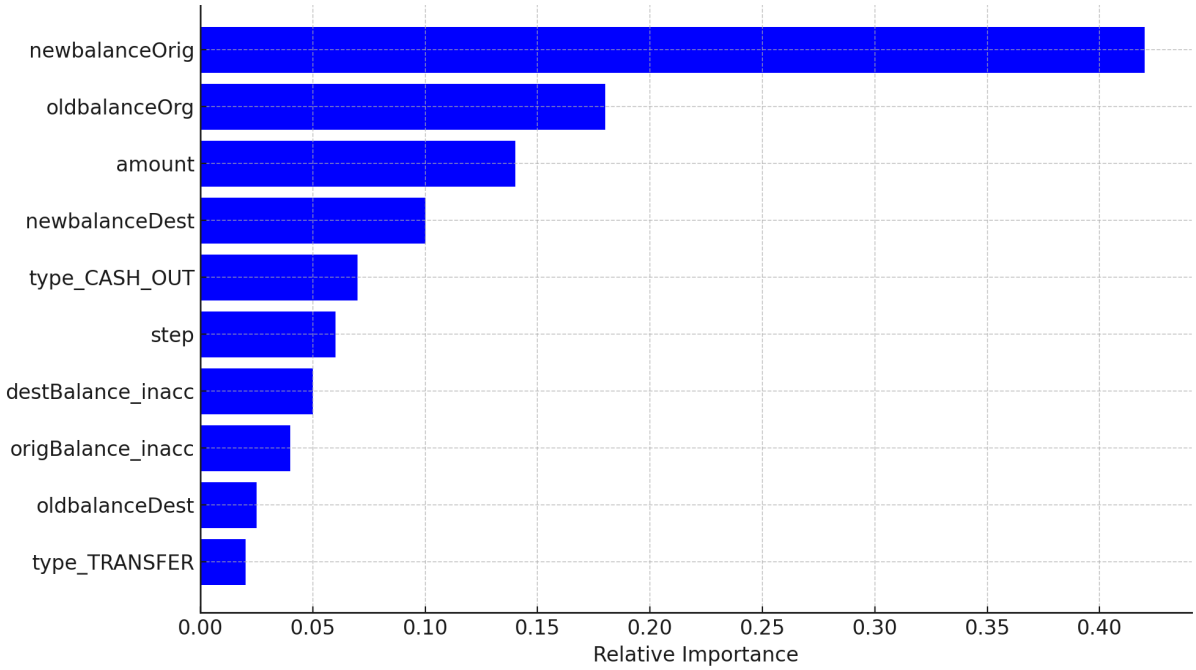

Рисунок 4.1 – Важливість ознак для моделі Random Forest

Крім того, ефективність моделі Random Forest було підтверджено за допомогою ROC-кривої (рисунок 4.2), яка демонструє практично ідеальне розділення класів. Лінія ROC проходить уздовж верхнього краю графіка, що відповідає значенню площі під кривою (AUC) рівному 1.00. Це означає, що модель з високою впевненістю ідентифікує всі позитивні випадки, не допускаючи хибних позитивів. Такий результат є ознакою надзвичайно високої здатності моделі до узагальнення при роботі з новими даними.

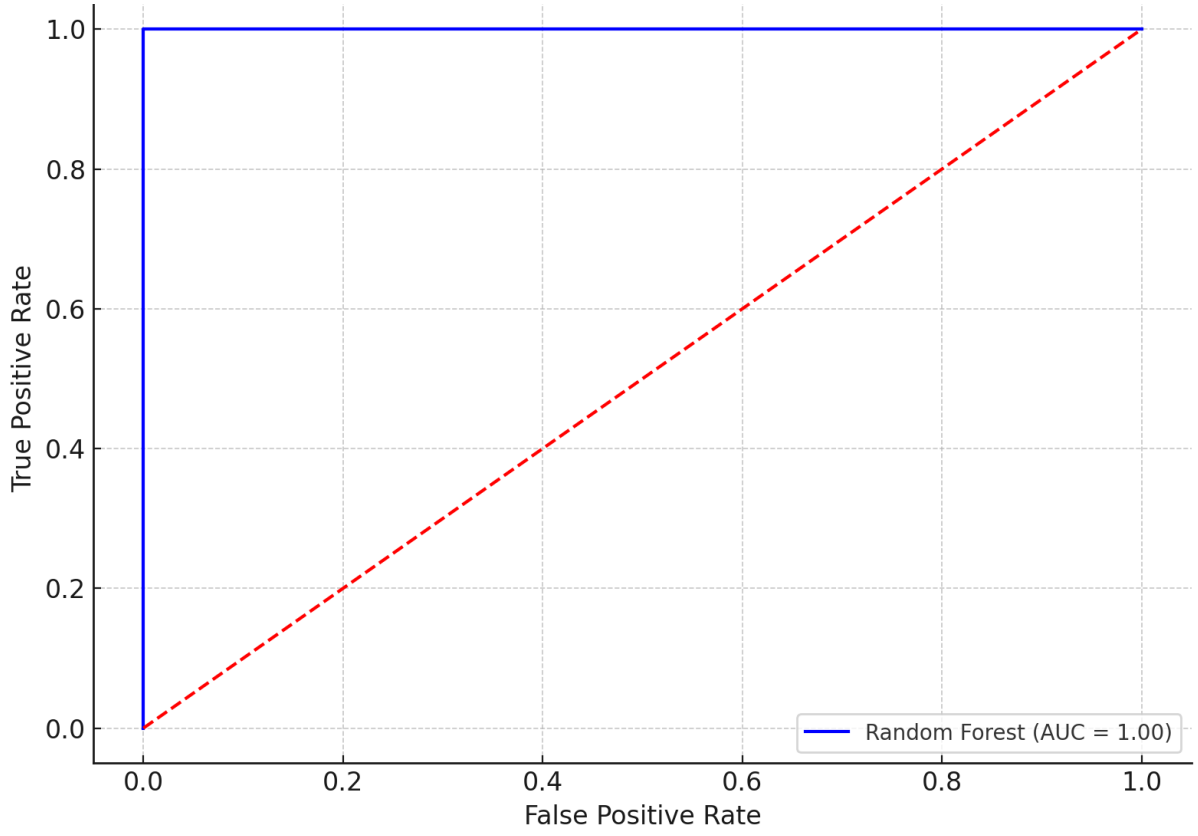

Рисунок 4.2 – ROC-крива для моделі Random Forest

Отже, модель Random Forest не лише показала найвищі кількісні результати, але й підтвердила свою ефективність через якісну інтерпретацію важливості ознак та ідеальну ROC-криву, що робить її найкращим кандидатом для практичного застосування в системах виявлення шахрайства.

ВИСНОВКИ

У межах виконання кваліфікаційної роботи було досягнуто поставлену мету – здійснено комплексне дослідження підходів до виявлення фінансового шахрайства за допомогою методів машинного навчання з акцентом на методи навчання з учителем. Робота включала як теоретичну, так і практичну частину, що забезпечило повноту аналізу проблеми та дозволило здійснити обґрунтоване порівняння ефективності різних алгоритмів класифікації. Було проаналізовано основні виклики у сфері боротьби з шахрайством, зокрема динамічну змінюваність шахрайських схем, високу частоту обробки транзакцій та складність фінансових даних, зокрема їхню незбалансованість.

Практична частина роботи реалізована з використанням синтетичного набору даних, згенерованого симулятором PaySim, який моделює реальні мобільні транзакції з чітким поділом на шахрайські та чесні операції. Було виконано повну обробку даних: аналіз структури, очищення від некоректних записів, нормалізація числових полів, кодування категоріальних ознак, а також поділ на тренувальний та тестовий набори. Особливу увагу було приділено боротьбі з класовою незбалансованістю, для підвищення якості моделі було застосовано undersampling негативного класу на етапі навчання.

У роботі реалізовано два алгоритми класифікації: логістичну регресію та Random Forest. Перша модель виступила як базова, оскільки є простою у реалізації та добре інтерпретується, але має обмежену здатність до виявлення нелінійних залежностей. Друга модель – Random Forest – є прикладом ансамблевого підходу, що поєднує переваги багатьох дерев рішень, забезпечуючи високу точність, стійкість до шуму та кращу генералізацію. За результатами моделювання було отримано значне покращення як кількісних, так і якісних показників при використанні Random Forest. Зокрема, ця модель демонструвала вищі значення метрик

точності, повноти, F1-міри та площі під кривою ROC, а також ефективніше виявляла шахрайські транзакції без суттєвого збільшення кількості хибних спрацьовувань.

Таким чином, у процесі виконання дипломної роботи було повністю реалізовано поставлені завдання: здійснено аналітичний огляд сучасних підходів до боротьби з шахрайством, побудовано та протестовано дві моделі класифікації, проведено порівняльний аналіз їх ефективності та сформульовано висновки щодо доцільності використання кожного з методів у залежності від контексту. На основі проведених експериментів можна стверджувати, що ансамблеві методи, зокрема Random Forest, є більш ефективними для виявлення шахрайства у фінансових транзакціях, особливо при роботі з високовимірними, незбалансованими та потенційно шумними даними.

Результати дослідження свідчать про доцільність подальшого розвитку систем виявлення шахрайства на базі машинного навчання. У подальшій роботі доцільно розширити порівняння моделей шляхом включення до аналізу інших алгоритмів, зокрема градієнтного бустингу, нейронних мереж, а також методів глибокого навчання. Перспективним напрямком є також застосування технік синтетичного збалансування класів, таких як SMOTE, що дозволить уникнути втрати важливої інформації при undersampling. Крім того, доцільно розглядати сценарії потокової обробки даних у режимі реального часу, що дозволить використовувати побудовані моделі у продакшн-середовищах банківських і платіжних систем.

Загалом, результати дослідження підтвердили, що сучасні алгоритми машинного навчання здатні ефективно вирішувати складні прикладні завдання виявлення шахрайства, а використані підходи можуть бути масштабовані та адаптовані для застосування в реальних фінансових установах. Отримані висновки створюють підґрунтя для подальших наукових досліджень та практичних розробок у сфері фінансової кібербезпеки.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. A comprehensive survey of data mining-based fraud detection research / C. Phua et al. *Artificial Intelligence Review*. 2010. Vol. 34, no. 1. P. 1–14. URL: https://arxiv.org/pdf/1009.6119 (date of access: 16.01.2025).

2. A Survey of Credit Card Fraud Detection Techniques: Data and Technique Oriented Perspective. *arXiv.org*. URL: https://arxiv.org/pdf/1611.06439 (date of access: 08.04.2025).

3. Bolton R. J., Hand D. J. Statistical fraud detection: A review. *Statistical Science*. 2002. Vol. 17, no. 3. P. 235–255.

4. Breiman L. Random forests. *Machine Learning*. 2001. Vol. 45, no. 1. P. 5–32.

5. Cortes C., Vapnik V. Support-vector networks. *Machine Learning*. 1995. Vol. 20, no. 3. P. 273–297.

6. Feature engineering strategies for credit card fraud detection / A. C. Bahnsen et al. *Expert Systems with Applications*. 2016. Vol. 51. P. 134–142.

7. He H., Garcia E. A. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*. 2009. Vol. 21, no. 9. P. 1263–1284.

8. Jha S., Guillen M., Westland J. C. Employing transaction aggregation strategy to detect credit card fraud. *Expert Systems with Applications*. 2012. Vol. 39, no. 16. P. 12650–12657.

9. Schmidhuber J. Deep learning in neural networks: An overview. *Neural Networks*. 2015. Vol. 61. P. 85–117.

10. Shen A., Tong R., Deng Y. Application of classification models on credit card fraud detection. *Service Systems and Service Management*. 2007. P. 1–4.

11. SMOTE: Synthetic minority over-sampling technique / N. V. Chawla et al. *Journal of Artificial Intelligence Research*. 2002. Vol. 16. P. 321–357.

12. Survey of fraud detection techniques / Y. Kou et al. *IEEE International Conference on Networking, Sensing and Control*. 2004.

13. Transaction aggregation as a strategy for credit card fraud detection / C. Whitrow et al. *Data Mining and Knowledge Discovery*. 2009. Vol. 18, no. 1. P. 30–55.

14. West J., Bhattacharya M. Intelligent financial fraud detection: A comprehensive review. *Computers & Security*. 2016. Vol. 57. P. 47–66.

15. Wang S., Zhou X. Deep learning-based methods for credit card fraud detection: A review. *IEEE Access*. 2022. Vol. 10. P. 56893–56905. URL: https://doi.org/10.1109/ACCESS.2022.3178964 (date of access: 02.03.2025).

16. Fraud detection for mobile payment systems using hybrid ensemble learning / X. Liu et al. *Journal of Financial Crime*. 2023. URL: https://doi.org/10.1108/JFC-11-2022-0253 (date of access: 18.04.2025).

17. Leveraging unsupervised pre-training and supervised fine-tuning for fraud detection / F. Carcillo et al. *IEEE Transactions on Neural Networks and Learning Systems*. 2021. URL: https://doi.org/10.1109/TNNLS.2020.2993373 (date of access: 25.03.2025).

18. Sebastião H., Gama J. Fraud detection in online banking transactions using dynamic ensemble selection. *Expert Systems with Applications*. 2021. URL: https://doi.org/10.1016/j.eswa.2021.115570 (date of access: 01.05.2025).

19. Nami M. R., Shajari M. Cost-sensitive learning for class-imbalanced credit card fraud detection. *Computers & Security*. 2020. URL: https://doi.org/10.1016/j.cose.2020.101759 (date of access: 14.02.2025).

20. Aggarwal C. C. Outlier Analysis. New York : Springer, 2017. 466 p.

21. Bishop C. M. Pattern Recognition and Machine Learning. New York : Springer, 2006. 738 p.

22. Géron A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. Sebastopol : O'Reilly Media, 2019. 819 p.

23. Han J., Kamber M., Pei J. Data Mining: Concepts and Techniques. Boston : Morgan Kaufmann, 2011. 744 p.