СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ
Навчально-науковий інститут (факультет) інформаційних технологій та електроніки
Кафедра інформаційних технологій та програмування

# Пояснювальна записка

до магістерської дипломної роботи
магістр
(освітньо-кваліфікаційний рівень)
на тему Методи машинного навчання для багаторівневого аналізу фінансових транзакцій та виявлення шахрайства

Виконав: студент 2 курсу, групи ІСТ-24зм
126 «Інформаційні системи та технології»
(шифр і назва спеціальності)
Купрієвич І. П.
(прізвище та ініціали)
Керівник Лифар В. О.
(прізвище та ініціали)
Рецензент Меняйленко О.С.
(прізвище та ініціали)

Київ – 2025 року

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ ДО МАГІСТЕРСЬКОЇ ДИПЛОМНОЇ РОБОТИ

СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Навчально-науковий інститут (факультет) інформаційних технологій та електроніки

Кафедра інформаційних технологій та програмування

Освітньо-кваліфікаційний рівень магістр

Спеціальність 126 «Інформаційні системи та технології»

(шифр і назва спеціальності)

ЗАТВЕРДЖУЮ
Завідувач кафедри ІТП
д.т.н., проф., Захожай О. І.
(підпис)
«____»_____2025р.

ЗАВДАННЯ
на магістерську дипломну роботу студенту
Купрієвич Інні Павлівні
(прізвище, ім'я, по батькові)

1. Тема роботи Методи машинного навчання для багаторівневого аналізу фінансових транзакцій та виявлення шахрайства
керівник роботиЛифар Володимир Олексійович, д.т.н., доц.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу
від «08» 12 2025року №241/17.03
2. Строк подання студентом роботи 15.12.2025
3. Вихідні дані до роботи: Матеріали науково-дослідної практики, науково-методична література; дані інтернет-мережі .
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
   4.1 Вступ
   4.2 Аналіз проблеми дослідження
   4.3 Провести порівняльний аналіз.
   4.4 Виконати тестування, налагодження та оцінку ефективності роботи програмного продукту.
   4.5 Висновки
   4.6 Перелік використаних джерел
5. Перелік графічного матеріалу (з точним значенням обов'язків креслень)

6. Консультанти розділів проекту (роботи)

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

7. Дата видачі завдання 18 жовтня 2025

**КАЛЕНДАРНИЙ ПЛАН**

| № з\п | Назва етапів дипломної роботи | Строк виконання етапів роботи | Примітка |
|---|---|---|---|
| 1 | Дослідження предметної галузі | 25.10.2 – 28.10.25 | |
| 2 | Пошук та аналіз існуючих рішень | 29.10.25 – 04.11.25 | |
| 3 | Аналіз проблеми дослідження | 05.11.25 – 15.11.25 | |
| 5 | Розробка інформаційно-аналітичної моделі | 16.11.25 – 25.11.25 | |
| 6 | Тестування | 25.11.25 – 02.12.25 | |
| 7 | Оформлення пояснювальної записки | 02.12.25 – 05.12.25 | |
| 8 | Підготовка та подання магістерської роботи до захисту | 06.12.25 – 06.12.25 | |

Студент_____ Купрієвич І.П.
(підпис) (прізвище та ініціали)

Керівник роботи_____ Лифар В.О.
(підпис) (прізвище та ініціали)

# РЕФЕРАТ

Магістерська дипломна робота: 57 стор., 16 рис., 2 таб., 12 джерел

У межах дослідження здійснено повний цикл аналізу, побудови та експериментальної перевірки каскадної моделі виявлення шахрайських транзакцій у фінансових системах з високим рівнем дисбалансу даних. Проведено огляд типів шахрайських операцій, проаналізовано сучасні підходи до боротьби з фінансовими зловживаннями, а також проведено оцінювання потенціалу методів машинного навчання для автоматизованого визначення аномальної платіжної активності.

Розроблено та реалізовано прототип системи ідентифікації шахрайських транзакцій на основі поетапної (каскадної) архітектури, що включає три послідовних рівні: первинне виявлення нетипової активності, уточнення параметрів ризикових транзакцій та фінальну класифікацію з використанням бустингового алгоритму. Запропонована модель забезпечує зниження кількості хибнопозитивних рішень і мінімізацію обсягів ручної верифікації транзакцій.

Проведено технічну реалізацію системи у середовищі Python із використанням бібліотек pandas, numpy, scikit-learn та CatBoost. Реалізовано модулі попередньої обробки, балансування даних, навчання класифікаційних алгоритмів і тестування результатів. Отримано дев’ять графічних візуалізацій роботи моделі, на основі яких проведено оцінювання якості детектування шахрайських операцій на кожному рівні каскаду.

За підсумками тестування підтверджено стабільність роботи системи, здатність до адаптації до поведінкових змін користувачів і збереження високих показників точності у випадках з мінімальними аномаліями. Результати дослідження засвідчили можливість впровадження побудованої

архітектури в реальні платіжні системи та цифрові фінансові сервіси для автоматизованого контролю транзакцій і запобігання шахрайству.

Розроблена система може використовуватися як практичний інструмент для банківських структур, платіжних сервісів, електронної комерції, а також у навчальних цілях для дослідження принципів побудови аналітичних моделей виявлення шахрайських транзакцій і перевірки алгоритмів машинного навчання в умовах дисбалансованих вибірок.

Ключові слова: шахрайські транзакції, машинне навчання, класифікація, каскадна архітектура, фінансовий моніторинг, аномалії, CatBoost, Random Forest, Python, виявлення шахрайства.

# Зміст

# ВСТУП

Актуальність теми. Стрімкий розвиток цифрової економіки, електронних платіжних сервісів та дистанційних фінансових операцій призвів до суттєвого збільшення обсягу транзакцій, що здійснюються в автоматизованих інформаційних системах. Зростання обсягів онлайн-платежів безпосередньо впливає на підвищення рівня фінансових ризиків та інтенсифікацію шахрайських схем, спрямованих на несанкціоноване отримання доступу до коштів користувачів, зміну параметрів платіжних операцій або викрадення конфіденційних даних.

Динамічне оновлення методів шахрайства, поява нових сценаріїв підміни особистості, автоматизованих атак та зловживань платіжними сервісами, а також використання спеціалізованих програмних модулів для прихованого відстеження фінансової активності створює критичні загрози для економічної безпеки державних і комерційних структур. У цих умовах питання оперативного виявлення аномальної поведінки стає фундаментальною складовою функціонування будь-якої платіжної системи.

Виявлення транзакцій із ознаками шахрайства потребує комплексного застосування інструментів фінансового моніторингу, поведінкового аналізу та алгоритмів штучного інтелекту. Традиційні методи ручного контролю та класичні статистичні процедури вже не забезпечують належної якості виявлення загроз, оскільки не здатні обробляти великі обсяги інформації в реальному часі та адаптуватися до безперервних змін у моделях поведінки шахраїв.

Головна мета магістерської роботи полягає в дослідженні моделей виявлення шахрайських транзакцій і розробці каскадної архітектури класифікації, здатної ідентифікувати відхилення у фінансових операціях за допомогою методів машинного навчання.

Для досягнення поставленої мети визначено такі завдання:

дослідити класифікацію типів шахрайських транзакцій та їх характеристик;

проаналізувати сучасні алгоритми машинного навчання, що застосовуються для виявлення аномальної поведінки у фінансових системах;

розглянути методи протидії шахрайству у високонавантажених платіжних потоках;

сформувати каскадну архітектуру моделі класифікації з поетапним звуженням множини транзакцій;

реалізувати систему аналізу транзакцій із використанням Python та бібліотек машинного навчання;

провести тестування розробленої моделі та оцінити її точність і стабільність в умовах дисбалансованих даних.

Система моніторингу транзакцій має бути зорієнтована на такі ключові аспекти:

контролювання платіжної активності користувачів у реальному часі;

виявлення нестандартної поведінки та транзакційних аномалій;

протидію фінансовому шахрайству та несанкціонованому доступу;

забезпечення мінімального рівня хибнопозитивних блокувань;

підтримку аналітичних механізмів для підтвердження або відхилення транзакцій;

автоматизацію моделі перевірки з можливістю оперативної перенастройки;

адаптивне навчання з використанням актуальних транзакційних даних.

Об’єкт дослідження — процес виявлення та класифікації транзакцій із ознаками шахрайства в електронних платіжних системах.

Предмет дослідження — методи побудови каскадних моделей класифікації та інструменти машинного навчання для автоматизованого виявлення аномальної платіжної активності.

Розв’язання поставлених завдань у межах даної магістерської роботи

дає змогу підвищити якість транзакційного контролю, знизити ризики фінансових втрат і забезпечити безперебійне функціонування електронних платіжних платформ в умовах постійного зростання шахрайських атак та кіберзагроз.

.

# 1 АНАЛІЗ ПРОБЛЕМИ ДОСЛІДЖЕННЯ

## 1.1 Загальна характеристика шахрайських транзакцій

Шахрайські транзакції становлять одну з ключових загроз для сучасних фінансових систем, оскільки вони пов’язані з умисним викривленням даних або обманом з метою отримання незаконної вигоди. З розвитком цифрових технологій та зростанням кількості безготівкових операцій масштаби таких правопорушень істотно збільшуються, що ускладнює завдання їх своєчасного виявлення та запобігання.

За своєю природою шахрайські транзакції охоплюють широкий спектр незаконних дій: від використання викрадених платіжних реквізитів до складних схем маніпуляцій у фінансових операціях. Головною метою зловмисників є отримання доступу до фінансових ресурсів або ухилення від відповідальності за рахунок спотворення дійсних даних. Найпоширеніші форми шахрайства включають крадіжку особистих даних, фішинг, маніпуляції з електронними платежами, мультиаккаунтинг та створення синтетичних ідентичностей [1].

Крадіжка особистих даних проявляється у несанкціонованому доступі до конфіденційної інформації: номерів банківських карток, паролів, персональних даних клієнтів. Значна частина таких випадків здійснюється за допомогою фішингових методів — підроблених сторінок чи електронних листів, створених із метою введення користувача в оману. Шахрайство з платежами охоплює дії, спрямовані на викривлення або підміну платіжних операцій з метою незаконного переказу коштів. До цього ж типу належить використання неавтентичних облікових записів, коли шахраї створюють множинні профілі або вигадані особи для доступу до фінансових послуг чи кредитних ресурсів.

Наявність таких видів шахрайства має суттєвий негативний вплив на

фінансові установи та їх клієнтів. Вона призводить до фінансових збитків, зниження довіри до електронних сервісів і створює додаткові ризики для стабільності фінансової інфраструктури. Крім того, швидкість і обсяги сучасних транзакцій ускладнюють використання виключно традиційних підходів до виявлення порушень, оскільки ручні методи контролю не здатні оперативно реагувати на динамічні та комплексні шахрайські схеми.

У такому контексті важливо впроваджувати сучасні автоматизовані підходи, зокрема моделі машинного навчання, які здатні аналізувати великі масиви даних у реальному часі. Завдяки здатності виявляти приховані закономірності між параметрами транзакцій — сумою, часом, географічним розташуванням, історією операцій та іншими характеристиками — такі системи істотно підвищують ефективність боротьби з шахрайством. Важливою їх перевагою є адаптивність, що дозволяє швидко реагувати на появу нових сценаріїв незаконних дій.

Таким чином, питання виявлення шахрайських транзакцій є критичним для забезпечення стабільної роботи фінансових установ, збереження коштів споживачів і підтримання довіри до цифрових фінансових сервісів. Використання сучасних технологій аналізу даних стає необхідною умовою для ефективної протидії зростаючим ризикам фінансового шахрайства.

## 1.2 Методи та технології протидії фінансовому шахрайству

Системи протидії фінансовому шахрайству постійно розвиваються, оскільки масштаби та складність незаконних дій з боку зловмисників зростають разом із цифровізацією фінансових операцій. Ефективне виявлення шахрайських транзакцій вимагає комплексного підходу, який поєднує організаційні заходи, аналітичні методи та сучасні технологічні рішення.

Одним із традиційних підходів є правилові системи (rule-based

systems), що ґрунтуються на заздалегідь визначених умовах та шаблонах поведінки. Наприклад, транзакції з нетиповими параметрами - перевищенням лімітів, незвичним географічним місцем або здійсненням операцій у короткий проміжок часу - можуть автоматично позначатися як підозрілі. Перевагою таких систем є простота реалізації та прозорість логіки, проте вони мають низьку ефективність проти нових або складних сценаріїв шахрайства, які не були враховані під час створення правил [2].

Більш сучасним і гнучким підходом є застосування методів машинного навчання, що дозволяють аналізувати великі обсяги даних та визначати приховані закономірності, властиві як легітимним, так і шахрайським операціям. Технології машинного навчання забезпечують адаптивність системи в умовах постійних змін, а також здатні виявляти як відомі, так і нові види шахрайських дій.

Серед найбільш поширених моделей машинного навчання у сфері фінансової безпеки використовують:

дерева рішень, що надають інтерпретовану структуру класифікації транзакцій за різними ознаками;

ансамблеві методи, такі як Random Forest або Gradient Boosting, які підвищують точність за рахунок об’єднання результатів кількох моделей;

алгоритми для роботи з дисбалансованими даними, оскільки у типових наборах транзакцій частка шахрайських операцій суттєво менша за частку легітимних;

методи кластеризації та виявлення аномалій, що дозволяють виявляти нетипові поведінкові патерни навіть без наявності міток класів.

Крім того, дедалі більшого поширення набувають системи поведінкової аналітики (behavioral analytics), які аналізують індивідуальні шаблони дій користувачів у цифрових сервісах: швидкість введення даних, послідовність операцій, характеристики пристроїв та мережевої активності. Комбінація таких характеристик дозволяє створити унікальний профіль клієнта та

помічати аномалії, що можуть сигналізувати про спробу шахрайства.

До технологічних рішень також належать системи біометричної ідентифікації, захищені канали передавання даних, багатофакторна автентифікація, інтегровані системи моніторингу транзакцій у режимі реального часу та технології аналізу великих даних.

Узагальнюючи, ефективна боротьба з фінансовим шахрайством вимагає поєднання різних підходів — від класичних правилових систем до високоточних моделей машинного навчання та інтелектуальних інструментів аналізу поведінки користувачів. Комплексне застосування таких методів дозволяє суттєво підвищити надійність фінансової інфраструктури та рівень захисту користувачів від незаконних дій..

## 1.3 Застосування машинного навчання у фінансовій сфері

Машинне навчання стало одним із ключових технологічних інструментів, що формують сучасну фінансову індустрію. Завдяки здатності аналізувати великі масиви даних, виявляти приховані закономірності та приймати рішення на основі статистичних залежностей, моделі машинного навчання широко використовуються у банківській справі, фінтех-сервісах, страхуванні, інвестиційній аналітиці та управлінні ризиками.

Одним із найбільш поширених напрямів застосування є автоматизоване кредитне скорингування, що передбачає оцінювання кредитоспроможності клієнтів на основі історичних даних, фінансової активності, соціально-демографічних характеристик та поведінкових факторів. Завдяки машинному навчанню такі системи забезпечують вищу точність прогнозування ризиків порівняно з традиційними методами, оскільки здатні адаптуватися до нових патернів взаємодії користувачів із фінансовими продуктами [11].

Не менш важливим напрямом є управління фінансовими ризиками, де моделі машинного навчання дозволяють прогнозувати можливі збитки, оцінювати ринкову волатильність, виявляти потенційно небезпечні активи

або події. Застосування таких моделей підтримує прийняття рішень у реальному часі та сприяє підвищенню стійкості фінансових установ до зовнішніх коливань.

Широкого поширення набули моделі машинного навчання у торговельних системах та інвестиційних платформах, де вони використовуються для аналізу ринкових даних, виявлення трендів, побудови торгових стратегій та прогнозування цінових змін. Алгоритмічна та високочастотна торгівля значною мірою спираються на машинне навчання як на основний інструмент для швидкої обробки даних та оптимізації торгових рішень.

Окрему увагу привертає застосування машинного навчання у виявленні аномалій та боротьбі з фінансовим шахрайством. У цьому контексті моделі аналізують багатовимірні характеристики транзакцій, поведінку користувачів та інші фактори, що дозволяє автоматично визначати підозрілі операції. Саме здатність ідентифікувати нетипові патерни, які важко помітити вручну, робить машинне навчання надзвичайно ефективним інструментом у протидії шахрайським діям.

Також машинне навчання активно використовується в персоналізації фінансових сервісів, зокрема у рекомендаційних системах, що пропонують користувачам релевантні продукти та послуги на основі аналізу їхньої історії та уподобань. Це сприяє підвищенню якості клієнтського обслуговування та ефективності маркетингових стратегій.

У підсумку, машинне навчання відіграє важливу роль у трансформації фінансової сфери, забезпечуючи підвищення точності аналітики, автоматизацію бізнес-процесів, зниження ризиків та значне покращення якості прийняття рішень. Інтеграція інтелектуальних технологій у фінансові системи не лише підсилює їхню функціональність, але й створює можливості для розвитку інноваційних сервісів і технологічних рішень..

## 1.4 Формулювання завдання дослідження

Ефективне виявлення шахрайських транзакцій у фінансових системах потребує побудови надійних та адаптивних інструментів аналізу даних, здатних працювати в умовах високої динамічності та значної нерівномірності розподілу класів. У цьому контексті постає необхідність розроблення підходу, який забезпечить точну ідентифікацію потенційно небезпечних операцій та зниження кількості помилкових спрацювань системи.

Основною проблемою є те, що частка шахрайських транзакцій зазвичай є надзвичайно малою порівняно з обсягом легітимних операцій. Це створює труднощі для класичних методів аналізу та ускладнює завдання побудови моделей, здатних коректно виявляти небажані дії. Крім того, сучасні шахрайські схеми постійно змінюються, що потребує використання моделей, які можуть адаптуватися до нових патернів поведінки.

З огляду на зазначене, у межах дослідження необхідно вирішити такі основні завдання:

проаналізувати сучасні методи протидії фінансовому шахрайству та визначити їхні переваги й обмеження;

вивчити можливості застосування моделей машинного навчання у задачі класифікації транзакцій;

підготувати та обробити набір даних, ураховуючи специфіку дисбалансованих вибірок та потребу в коректному формуванні ознак;

розробити багаторівневий підхід до класифікації, у якому декілька моделей машинного навчання працюватимуть послідовно або взаємодоповнююче;

провести порівняльний аналіз ефективності моделей за допомогою відповідних метрик оцінювання, зокрема з акцентом на здатність правильно визначати шахрайські операції;

узагальнити отримані результати та сформувати рекомендації щодо практичного застосування побудованої системи.

Виконання цих завдань дозволить створити цілісний підхід до автоматизованого виявлення шахрайських транзакцій на основі методів машинного навчання та оцінити його придатність для практичного використання у фінансових системах..

## Висновки до розділу 1

У першому розділі було здійснено комплексний аналіз ключових аспектів, пов’язаних із природою фінансового шахрайства та сучасними підходами до його виявлення. Розглянуто основні типи шахрайських транзакцій, їхні характерні ознаки та способи реалізації, що дозволило сформувати узагальнене розуміння загроз, з якими стикаються фінансові установи в умовах зростаючої цифровізації.

Аналіз існуючих методів протидії засвідчив, що традиційні правилові системи мають суттєві обмеження у контексті динамічного розвитку шахрайських схем, адже ґрунтуються на фіксованих сценаріях і є недостатньо гнучкими. У цьому зв’язку методи машинного навчання демонструють значно вищу ефективність завдяки здатності виявляти приховані залежності, адаптуватися до нових поведінкових патернів та працювати з великими масивами даних.

Також було проаналізовано практичні напрями застосування машинного навчання у фінансовому секторі: від кредитного скорингу та управління ризиками до автоматизованого виявлення аномалій і підозрілих операцій. Це підкреслює доцільність використання інтелектуальних технологій у системах захисту фінансових транзакцій.

Сформульоване завдання дослідження визначає потребу у створенні ефективного багаторівневого підходу до класифікації транзакцій, який

забезпечуватиме точне виявлення шахрайських дій і зменшуватиме кількість помилкових ідентифікацій. Розроблення та оцінювання таких моделей стануть основою подальших етапів роботи.

Отже, результати розділу створюють необхідне теоретичне підґрунтя для подальшого проєктування, реалізації та аналізу системи виявлення шахрайських транзакцій на основі сучасних методів машинного навчання.

# 2 РОЗРОБЛЕННЯ АРХІТЕКТУРИ МОДЕЛІ ДЛЯ ВИЯВЛЕННЯ ШАХРАЙСЬКИХ ТРАНЗАКЦІЙ

## 2.1 Теоретичні засади застосування машинного навчання для виявлення транзакційних аномалій

Виявлення аномалій у фінансових транзакціях є одним із ключових завдань систем протидії шахрайству, оскільки саме нетипові операції найчастіше свідчать про можливі зловживання. Методи машинного навчання дозволяють автоматизовано аналізувати великі обсяги даних і виявляти приховані закономірності, що недоступні при ручному контролі. Для коректного розуміння підходів, що застосовуються у виявленні аномалій, необхідно розглянути основи роботи моделей машинного навчання, зокрема нейронних мереж.

Одним із найпоширеніших інструментів машинного навчання є штучні нейронні мережі, архітектура яких наведена на рисунку 2.1. Нейронна мережа складається з трьох основних типів шарів: вхідного, прихованих та вихідного, де кожний шар містить набір взаємопов’язаних нейронів.

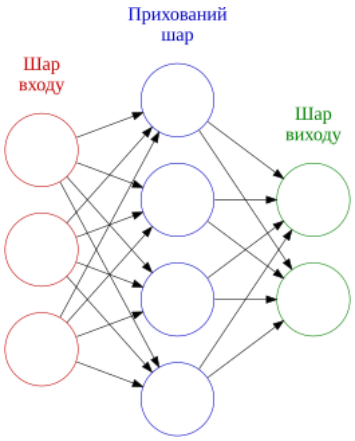


Рисунок 2.1 – Архітектура нейронної мережі

Вхідний шар отримує характеристики транзакції, зокрема:

• суму операції;

• час її проведення;

• геолокацію;

• тип платіжного інструменту тощо

Приховані шари відповідають за обчислення нелінійних закономірностей. Робота окремого нейрона базується на зваженій сумі вхідних сигналів, що описується формулою (2.1):.

$$z = w_1x_1 + w_2x_2 + \cdots + w_nx_n + b, \qquad (2.1)$$

де $w$ – вагові коефіцієнти, що визначають важливість ознак;

$b$ – зміщення

$x$ – значення вхідних параметрів.

Вихідний шар формує остаточний прогноз:

бінарний (0 - нормальна транзакція, 1 - шахрайська),

ймовірнісний або

багатокласовий, залежно від типу задачі.

Типи методів виявлення аномалій

Аномалії у фінансових операціях визначаються як відхилення від звичної поведінки користувача: нетипові суми переказів, незвичні місця входу, різкі зміни у частоті платежів тощо. Основні підходи до виявлення таких відхилень представлені на рисунку 2.2..

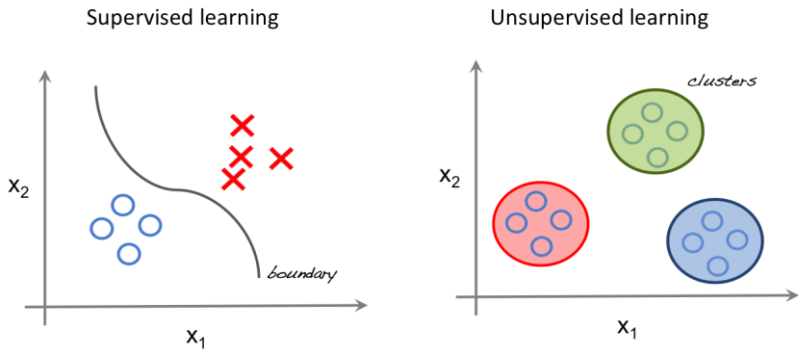


Рисунок 2.2 – Методи поділяють на два основних класи

Методи без учителя (Unsupervised Learning)

Використовуються за відсутності розмічених даних. До них належать:

– DBSCAN, K-Means;

– Isolation Forest;

– One-Class SVM.

Такі методи формують уявлення про «нормальну» поведінку та виявляють відхилення.

2. Методи з учителем (Supervised Learning)

Потребують наявності міток «шахрайство/не шахрайство». Сюди відносять:

– логістичну регресію;

– дерева рішень;

– випадковий ліс;

– глибокі нейронні мережірегресія, дерева рішень, випадковий ліс та глибокі нейронні мережі.

Формалізація задачі виявлення аномалій

Транзакція може бути представлена у вигляді вектора ознак, як

показано у формулі (2.2):

$$X = (x_1, x_2, \dots, x_n), \tag{2.2}$$

де кожен елемент $x_i$ описує певну характеристику операції.

Завдання моделі — побудувати функцію, що оцінює ймовірність того, що транзакція є шахрайською. Це подано у формулі (2.3): [12].

$$f(x) = P(y = 1|X) \tag{2.3}$$

де y = 1 означає, що операція класифікується як аномальна.

Таким чином, моделі машинного навчання дозволяють як знаходити нелінійні залежності між характеристиками транзакцій, так і формувати точні прогнози на основі великої кількості параметрів.

У цьому підрозділі було розглянуто фундаментальні принципи машинного навчання у сфері виявлення аномалій, а також наведено ключові формули (2.1)–(2.3) та рисунки (2.1) і (2.2), які ілюструють структуру нейронної мережі та підходи до класифікації аномалій. Отримані положення є теоретичною основою для подальшого аналізу алгоритмів і побудови архітектури моделі, розглянутих у наступних підрозділах

## 2.2 Огляд відомих алгоритмів

У процесі виявлення шахрайських транзакцій широко застосовуються різні алгоритми машинного навчання, кожен із яких має свої особливості, переваги та недоліки. У цьому підрозділі представлено огляд найбільш поширених методів, що використовуються для класифікації аномальних фінансових операцій. Також збережено всі формули та рисунки, наведені в оригінальному тексті, із відповідними адаптованими описами.

Логістична регресія є одним із базових статистичних методів бінарної

класифікації. Вона використовується для оцінки ймовірності того, що транзакція належить до класу «шахрайська». Принцип її роботи наведено на рисунку 2.3, де показано перетворення вхідних ознак на ймовірнісний результат за допомогою сигмоїдної функції.

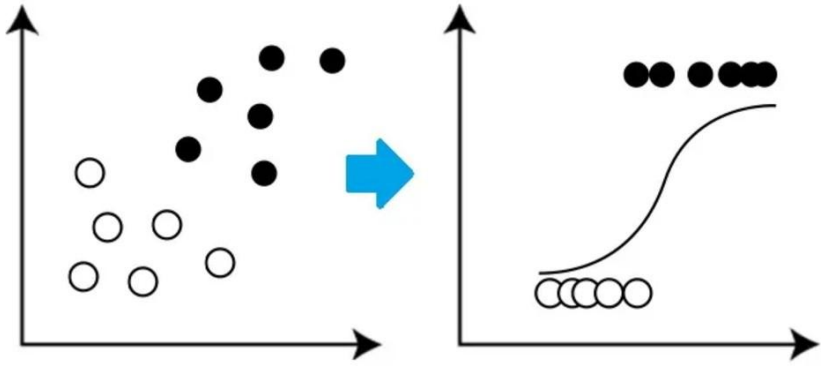

Рисунок 2.3 – Логістична регресія

$$P\left(y=\frac{1}{X}\right)=\frac{1}{1+e^{-(w_0+w_1x_1+..++w_nx_n+b).}} \qquad (2.4)$$

де

$x_i$ - ознаки транзакції,

$w_i$- вагові коефіцієнти моделі,

b - зміщення.

Дерева рішень є інтерпретованими моделями, що розбивають дані на підмножини за певними критеріями. У процесі побудови дерева використовують показники чистоти вузлів, зокрема:.

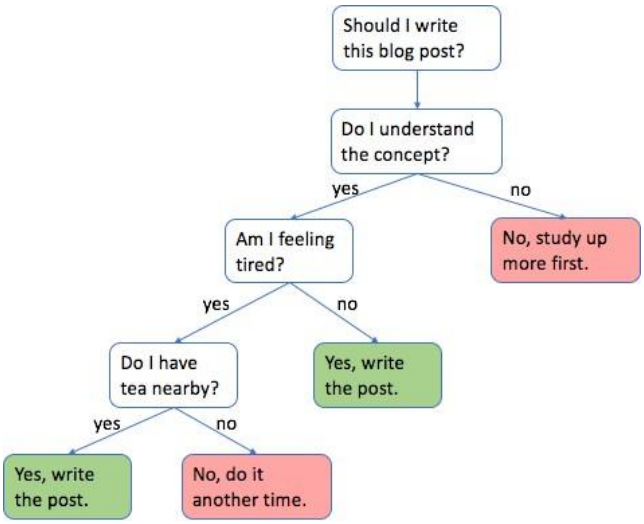


Рисунок 2.4 – Дерева ухвалення рішень

критерій Джині формула (2.5):

$$Gini = 1 - \sum_{i=1}^{C} p_i^2 \qquad (2.5):$$

ентропію, формула (2.6):

.

$$\text{Entropy} = -\sum_{i=1}^{C} p_i loog_2(p_i) \quad (2.6)$$

інформаційний приріст, формула (2.7):

$$\text{InfGain} = \text{Entropy(parent)} - \sum_{i=1}^{k} \frac{n_i}{n} \text{Entropy}(\text{child}_j) \quad (2.7)$$

зниження дисперсії для регресійних дерев, формула (2.8):

$$\text{VR} = \text{Var(parent)} - \left(\frac{n_{left}}{n}\text{Var(left)} + \frac{n_{right}}{n}\text{Var(right)}\right) \quad (2.8)$$

Випадковий ліс є ансамблевим методом, що поєднує декілька дерев рішень. Основна ідея полягає у випадковому виборі ознак та підмножин даних для тренування кожного дерева.

Ймовірність класифікації транзакції як шахрайської розраховується шляхом голосування дерев, що подано у формулі (2.9)

$$P\left(y = \frac{1}{X}\right) = \frac{1}{T}\sum_{i=1}^{T} h_t(X) \qquad (2.9)$$

де T - кількість дерев,

$h_i(X)$ - прогноз і-го дерева.

Принцип роботи випадкового лісу ілюстровано на рисунку 2.5

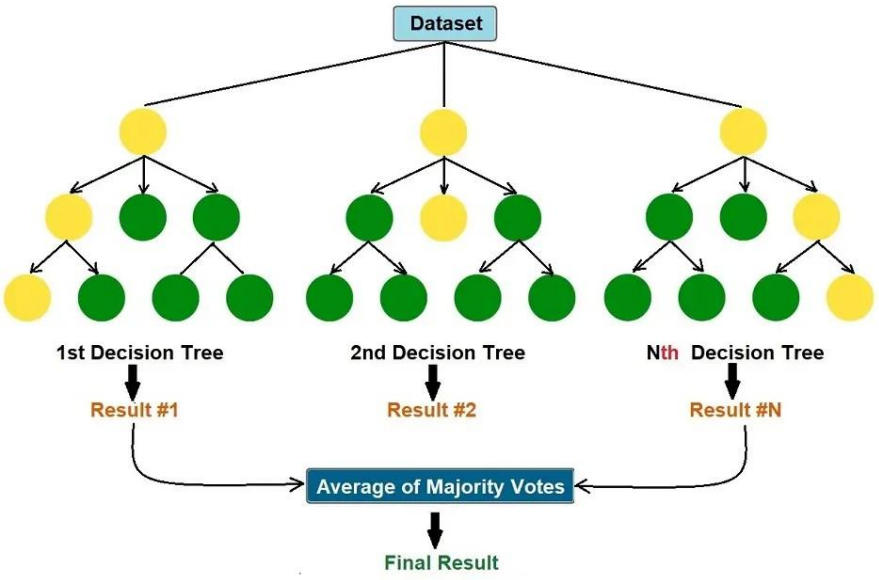


Рисунок 2.5 – Випадковий ліс

Глибокі нейронні мережі складаються з багатьох прихованих шарів, що дозволяє їм моделювати складні нелінійні залежності. Однією з

найпоширеніших функцій активації є ReLU, представлена у формулі (2.10).

$$f(z) = \max(0, z), \qquad (2.10)$$

Для задач багатокласової класифікації у вихідному шарі часто застосовується Softmax, що показано у формулі (2.11):

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \qquad (2.11):$$

де $z_i$ - вагова сума нейрона,

n - кількість класів.

Градієнтний бустинг об'єднує послідовність слабких моделей для поліпшення точності передбачень. Процес складається з кількох етапів:

Ініціалізація моделі, формула (2.12):

$$F_o = \operatorname{argmin} \sum_{i=1}^{n} L(y_i, \gamma) \qquad (2.12):$$

Обчислення градієнтів формула (2.13):

$$r_i = \frac{\partial L(y_i F_{m-1}(x_i))}{\partial F_{m-1}(x_i)} \qquad (2.13):$$

Оновлення моделі формула (2.14):

$$F_m(x) = F_{m-1}(x) + \mu \cdot h_m(x) \qquad (2.14):$$

де $\mu$ - коефіцієнт навчання,

$h_m(x)$ - нове дерево на m-й ітерації.

Порівняння бібліотек наведено в таблиці 2.1.

Таблиця 2.1 – Порівняння бібліотек бустингу

| *Характеристика* | *XGBoost* | *LightGBM* | *CatBoost* |
|---|---|---|---|
| Швидкість | Висока, але повільніша за LightGBM. | Дуже швидкий завдяки Leaf-wise стратегії. | Швидкий, але трохи повільніший за LGBM. |
| Тип росту дерева | Level-wise. | Leaf-wise (більш гнучкий, швидкий). | Symmetric tree. |
| Підтримка категоріальних ознак | Ні (треба one-hot). | Частково (автообробка через параметри). | Так, нативна підтримка. |
| Модель за замовчуванням | Хороша, але потребує тюнінгу. | Добра, але чутлива до learning rate. | Дуже стабільна. |
| Параметри | Багато, потребує ретельного тюнінгу. | Менше, але чутлива до гіперпараметрів. | Просте налаштування. |

Найбільш відомими реалізаціями бустингу є XGBoost, LightGBM і CatBoost.

У цьому огляді розглянуто ключові алгоритми, що є основою для побудови систем виявлення шахрайських транзакцій. Всі формули (2.4 -2.14) та рисунки (2.3 -2.5), наведені в оригінальному тексті, були збережені й інтегровані у новий виклад. Це створює цілісне теоретичне підґрунтя для подальшої розробки й оцінювання моделей у контексті фінансової безпеки.

## 2.3 Обґрунтування вибору підходу

Завдання виявлення шахрайських транзакцій у фінансових системах потребує використання таких методів, які здатні працювати в умовах високої інтенсивності потоку операцій, постійної зміни поведінкових патернів та низької частки аномальних записів. За цих умов традиційні системи контролю, що базуються на фіксованих правилах або одномодельних класифікаторах, демонструють недостатню гнучкість і не забезпечують належного рівня точності. Це стало основною причиною формування

комбінованого каскадного підходу, що використовується в межах даного дослідження.

Ключова ідея обраного підходу полягає у розподілі процесу класифікації на декілька послідовних етапів, кожен з яких виконує окрему функцію щодо відбору транзакцій залежно від рівня ризику. Перший етап каскаду виконує грубе фільтрування і відсікання транзакцій, що повністю відповідають нормальній поведінці користувача. Це дозволяє істотно зменшити навантаження на подальші моделі, зберігаючи при цьому швидкодію обробки.

Другий етап має більш деталізований характер і застосовує алгоритми з вищою здатністю до розпізнавання складних кореляційних залежностей між ознаками транзакцій. На цьому рівні здійснюється аналіз операцій, що потенційно можуть містити ознаки шахрайства, але не були остаточно віднесені до аномальних під час первинної обробки.

Третій рівень каскаду зосереджується на найскладніших для класифікації транзакціях. Після двоетапного фільтрування в системі залишається незначна, але найбільш підозріла підмножина записів. Саме на цьому рівні застосовуються моделі з максимальним рівнем точності, але вищими обчислювальними витратами. Такий розподіл дозволяє одночасно утримувати баланс між продуктивністю, точністю та ефективністю роботи системи.

Принцип функціонування описаного каскадного механізму представлено на рисунку 2.6 – Принцип роботи каскадного підходу який ілюструє логіку поетапної фільтрації та повторної оцінки транзакцій. На кожному етапі зменшується кількість операцій, проте зростає інтенсивність аналізу та рівень точності прогнозу. У такий спосіб система поступово виділяє транзакції, найбільш схильні до ознак шахрайства, мінімізуючи при цьому частку помилкових спрацювань і підвищуючи стабільність процесу.

Використання каскадної моделі обґрунтоване тим, що у фінансових

потоках значна частина транзакцій є типовою та безпечною, тому застосування ресурсомістких алгоритмів до кожної операції є недоцільним. Окреме опрацювання лише підозрілих операцій дозволяє оптимізувати час обробки та підвищити системну ефективність, особливо в умовах великих транзакційних масивів.

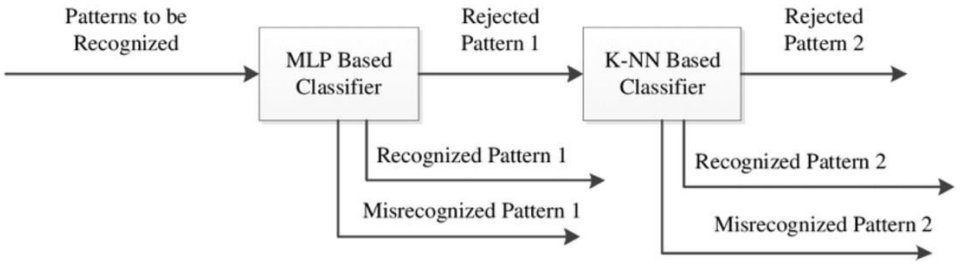


Рисунок 2.6 – Принцип роботи каскадного підходу

Таблиця 2.2 – Порівняння двох підходів навчання: каскадного і модифікація архітектури

| *Критерій* | *Каскадна модель* | *Модифікація архітектури* |
|---|---|---|
| Складність моделі | Простота, менше етапів, але треба врахувати кілька класифікацій | Складніша архітектура, часто з великою кількістю шарів |
| Час тренування | Швидше через простіші етапи та меншу кількість параметрів | Повільніше, оскільки моделі можуть мати багато параметрів та складні архітектури |
| Простота налаштування | Легше налаштувати та адаптувати для задач, що включають відсівання легко розпізнаваних даних | Складніше налаштування та оптимізація нейромереж, потребує більше досвіду |
| Гнучкість у обробці складних даних | Обмежена: каскадні моделі можуть бути ефективними лише при простих шаблонах | Висока: нейромережі можуть виявляти складні взаємозв'язки в даних, особливо у великих обсягах |
| Застосування для виявлення шахрайства | Добре працює для розпізнавання шаблонів шахрайства з очевидними ознаками (наприклад, дуже великі суми або підозрілі платіжні реквізити) | Краще для виявлення складних аномалій, коли шахрайські дії не можна виявити лише за допомогою явних шаблонів |
| Інтерпретованість | Вища інтерпретованість, оскільки кожен етап моделі можна проаналізувати | Нейромережі важче інтерпретувати, особливо при глибинних моделях |

## Висновки до підрозділу

Обраний каскадний підхід забезпечує компроміс між швидкістю та

точністю, дозволяючи поступово звужувати множину транзакцій із використанням моделей різної складності. Завдяки такому принципу розподілу навантаження система набуває здатності гнучко реагувати на появу нових типів шахрайства, забезпечувати масштабованість і зберігати прогнозну стабільність у реальному часі. Рисунок 2.6 відображає логічну побудову обраної архітектури та підтверджує доцільність її використання у задачах протидії фінансовим зловживанням.

# 3 РЕАЛІЗАЦІЯ КАСКАДНОЇ АРХІТЕКТУРИ ТА АНАЛІЗ РЕЗУЛЬТАТІВ МОДЕЛІ

## 3.1 Підготовка та аналіз даних

Набір даних, використаний у дослідженні, містить транзакції, виконані за допомогою кредитних карток у вересні 2013 року європейськими користувачами. Загальний обсяг вибірки складає 284 807 операцій, з яких лише 492 підтверджені як шахрайські випадки. Таким чином, частка аномальних записів становить 0,172%, що демонструє різко виражений дисбаланс класів.

Співвідношення кількості шахрайських транзакцій до загального масиву операцій наведено на рисунку 3.1 – Співвідношення легітимних та шахрайських транзакцій у вибірці

Таке співвідношення чітко підкреслює складність моделювання, оскільки клас «1» (шахрайство) є статистично мінімальним, а більшість алгоритмів машинного навчання в умовах дисбалансу схильні переорієнтовуватися на домінуючий клас.

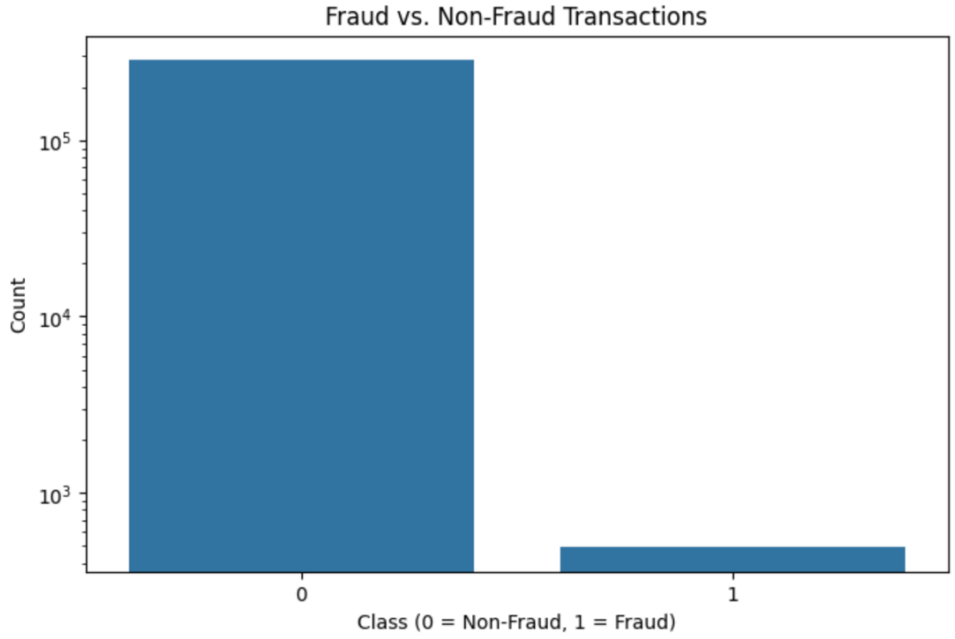


Рисунок 3.1 – Співвідношення легітимних та шахрайських транзакцій у вибірці

Особливістю набору є те, що всі ознаки, крім двох параметрів, були попередньо трансформовані методом PCA, що забезпечило приховування чутливої інформації та зменшення кореляцій між атрибутами. Після перетворення сформовано 28 компонент V1–V28. Окрім цього, у структурі даних присутні змінні:

Time — кількість секунд між транзакцією та першою операцією у вибірці;

Amount — сума транзакції;

Class — цільова змінна (0 — нормальна, 1 — шахрайська).

Кореляційні взаємозв’язки між ознаками подано на рисунку 3.2 – Матриця кореляційних залежностей у вибірці транзакцій

Візуальний аналіз матриці дозволяє визначити наявність імовірних

ознак, що можуть впливати на класифікацію, але також підтверджує відсутність прямої ідентифікаційної інформації про клієнтів.

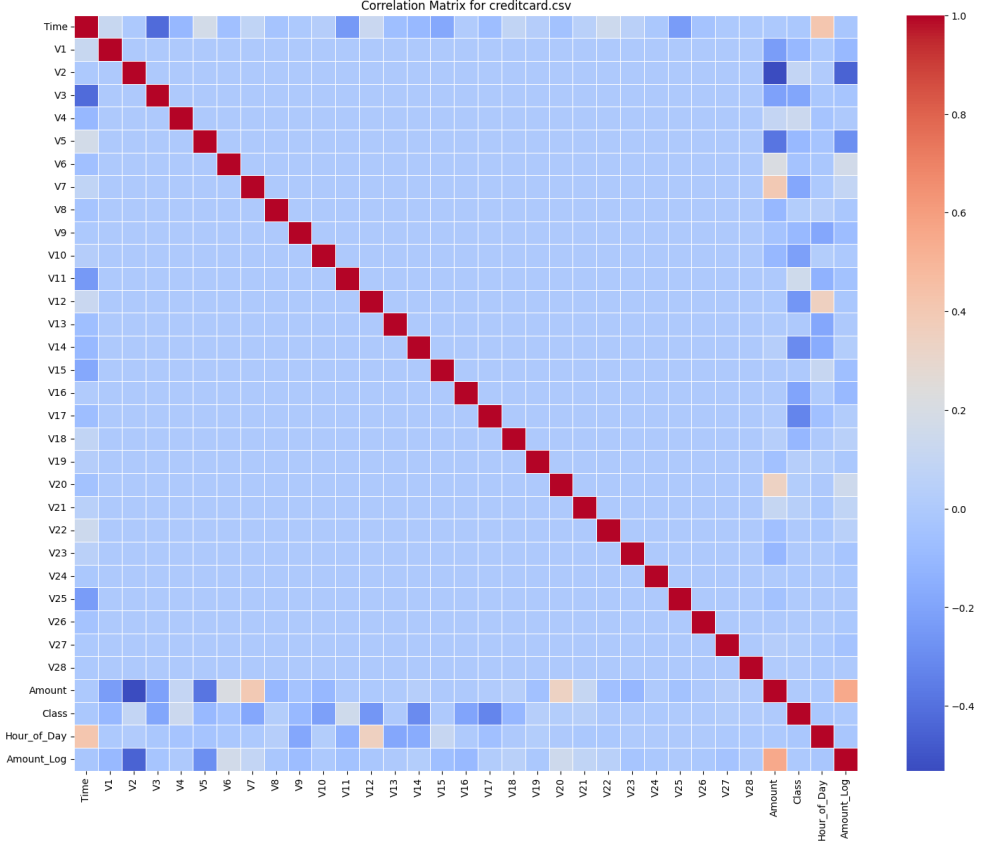


Рисунок 3.2 – Кореляційна матриця ознак транзакцій після перетворення методом PCA

Зважаючи на суттєвий дисбаланс класів, одним із ключових завдань попереднього етапу стала коректна підготовка вибірки. Зокрема, здійснено:

масштабування числових параметрів,

очищення від шумових значень,

формування похідних ознак для поведінкової оцінки.

Позначення часових патернів також мало важливе значення, оскільки шахрайські операції часто мають нетипові часові зміщення та географічні розриви

Результати аналізу дозволили кристалізувати інформаційні характеристики, необхідні для побудови каскадної моделі подальшої класифікації, що описується у наступному підрозділі.

## 3.2 Архітектура моделі розробки

Архітектура моделі для виявлення шахрайських транзакцій будується з

урахуванням каскадного принципу обробки даних, що дозволяє поступово звужувати множину транзакцій, фокусуючись на тих, які мають найбільш високий ризик шахрайських ознак. Такий підхід дає змогу оптимізувати ресурсне навантаження на модель і підвищити точність класифікації, не застосовуючи складні обчислювальні методи до всієї вибірки одночасно.

Перший рівень каскаду виконує базову фільтрацію транзакцій за попередньо визначеними ознаками: часом проведення операції, сумою платежу, географічною зоною, поведінковими параметрами користувача. На цьому етапі система автоматично відсіює операції, що повністю узгоджуються зі стабільним користувацьким профілем.

Другий рівень здійснює розподіл операцій на групи ризику, концентрація яких демонструє можливі відхилення. Для цього використовуються статистичні і поведінкові індикатори, сформовані на підставі аналізу активності користувачів за певні часові періоди.

Третій рівень системи є найбільш аналітичним і застосовує машинні моделі класифікації з високою точністю. Використання цього рівня стає можливим лише після попереднього відсікання транзакцій на попередніх етапах каскаду, що гарантує зменшення обсягу даних, які необхідно опрацьовувати складним алгоритмам.

Принцип побудови каскадної архітектури представлено на рисунку 3.3 – Схема поетапної обробки транзакцій у каскадній моделі На зображенні відображено логіку переходу між рівнями перевірки: від загальної фільтрації до цільового аналізу потенційно ризикових операцій. Кожний наступний рівень забезпечує глибше опрацювання, а також вищу точність класифікаційного висновку.

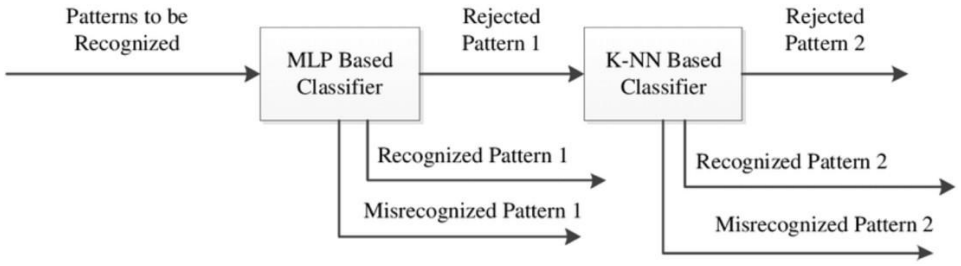


Рисунок 3.3 – Схема каскадного аналізу фінансових транзакцій

У структурі системи передбачений елемент повторної перевірки. Транзакції, які не можуть бути остаточно класифіковані на другому рівні, повертаються для додаткового аналізу до третього рівня. Це дозволяє зменшити кількість хибнопозитивних спрацювань і водночас підвищити результативність прийнятих рішень у частині блокування або підтвердження транзакції.

На схематичному поданні видно, що після завершення третього етапу формується виробниче рішення, яке може мати один із трьох результатів:

транзакція підтверджується як легітимна;

транзакція блокується як шахрайська;

транзакція позначається як потребуюча додаткової перевірки.

Таким чином, поєднання каскадності з багаторівневим аналізом дозволяє забезпечити високу точність та адаптивність системи. Рисунок 3.3 ілюструє побудовану архітектуру, підтверджуючи її здатність не лише обробляти великі обсяги даних, а й ефективно локалізувати підозрілі випадки для детальнішого аналізу.

## 3.3 Вибір метрик для подальшого оцінювання результатів

Оцінювання ефективності моделі виявлення шахрайських транзакцій є ключовим етапом дослідження, оскільки від коректності обраних метрик

залежить здатність системи адекватно реагувати на ризикові дії користувачів. У зв'язку з високим дисбалансом класів у вибірці застосування традиційної метрики точності (Accuracy) є недостатньо інформативним. Така метрика може демонструвати високий показник навіть у разі, коли модель майже повністю ігнорує шахрайські операції.

Базові метрики оцінювання

Інформативнішим підходом є використання метрик, що деталізують співвідношення правильних і помилкових рішень, зокрема:

Precision (точність класифікації) — характеризує частку дійсно шахрайських операцій серед усіх транзакцій, які модель класифікувала як аномальні.

$$\text{Precision} = \frac{TP}{\text{TP+FP}} \qquad (3.1)$$

Recall (повнота або чутливість) — показує, яку частку шахрайських операцій модель змогла ідентифікувати коректно.

$$\text{Recall} = \frac{TP}{\text{TP+FN}} \qquad (3.2)$$

де

TP — коректно класифіковані шахрайські операції,

FP — легітимні операції, помилково позначені як шахрайські,

FN — шахрайські транзакції, які модель не розпізнала.

Рівень балансу між цими двома показниками відображається F1-мірою:

$$\text{F1} = 2\frac{\text{Precision} \cdot \text{Recall}}{\text{Precision+ Recall}} \qquad (3.3)$$

F1-міра є особливо важливою у випадку, коли важливо не лише виявити шахрайські транзакції, а й мінімізувати хибнопозитивні спрацювання.

Метрики для дисбалансованих даних

Зважаючи на те, що кількість легітимних транзакцій значно переважає кількість шахрайських, важливо застосовувати AUC-ROC (Area Under Curve – Receiver Operating Characteristic). Цей показник оцінює здатність моделі розділяти два класи — нормальні та шахрайські — на різних порогах класифікації. Чим більша площа під кривою ROC, тим вища якість розрізнення транзакцій.

Окрім цього, корисним є аналіз матриці змішування, що дозволяє визначати співвідношення правильно та неправильно класифікованих випадків. Такий підхід дозволяє виявити, чи не схильна модель пропускати шахрайські транзакції або, навпаки, надмірно блокувати легітимні дії.

Особливості інтерпретації результатів

У системах протидії фінансовому шахрайству важливим є не лише загальний рівень точності, а й налаштування порогу класифікації, який визначає момент переходу між рішеннями «ризик» і «безпека». Занадто низький поріг може призвести до значного зростання кількості хибнопозитивних випадків, створюючи зайве навантаження на операторів перевірки транзакцій. Надто високий поріг — до пропуску шахрайських операцій.

Оптимізація порогових значень, а також збалансоване застосування показників Precision, Recall та F1-міри дозволяють максимально адаптувати модель до реальних умов функціонування платіжних систем.

Метрики оцінювання є критичним інструментом у процесі перевірки якості моделі класифікації. Формули (3.1)—(3.3) відображають базові критерії якості класифікації у системі з високим дисбалансом даних. Використання комплексного підходу до аналізу результатів дозволяє забезпечити не лише

точність і стабільність, а й адаптивність системи виявлення шахрайських транзакцій у реальному часі.

## 3.4 Реалізація моделі на Python

Реалізація моделі для виявлення шахрайських транзакцій була здійснена мовою програмування Python, що на сьогодні є найпоширенішою платформою для побудови систем машинного навчання та обробки високоризикових фінансових операцій. Вибір саме Python обумовлений його гнучкістю, широким спектром доступних бібліотек та здатністю ефективно взаємодіяти з великими обсягами даних у режимі офлайн та онлайн-обробки. Завдяки високій продуктивності інструментів Python стало можливим виконати повноцінний цикл реалізації — від завантаження даних і попередньої обробки до тестування, оцінювання та валідації отриманих результатів.

Підготовка середовища і підключення бібліотек

На початковому етапі було сформовано робоче середовище, встановлено необхідні пакети та виконано підключення бібліотек. Основу середовища склали:

pandas — для роботи з табличними структурами та агрегування результатів;

numpy — для маніпуляцій з масивами та числовими даними;

scikit-learn — для реалізації каскадної моделі та проведення класифікаційних експериментів;

*imbalanced-learn (imblearn) — для корекції дисбалансу транзакційних класів;

matplotlib, seaborn — для графічного аналізу проміжних даних.

Використання цих бібліотек надало можливість не лише опрацьовувати дані, але й оцінювати поведінкові закономірності, групувати транзакції та підтверджувати кореляційні фактори, виділені на попередніх етапах

дослідження.

Етап завантаження та первинної верифікації даних

Дані було розділено на тренувальний та тестовий набори із дотриманням пропорції, описаної у попередньому розділі. Додатково виконано перевірку на наявність пропусків, дубльованих записів та потенційної невідповідності форматів. Особлива увага приділялася значенню змінних Time та Amount, оскільки саме вони демонструють наявність аномальної інтенсивності або нетипового часово-географічного розподілу транзакцій.

Після первинної перевірки дані були нормалізовані, що дозволило усунути вплив різних масштабів значень. Це забезпечило рівномірне сприйняття ознак алгоритмами подальших рівнів класифікації та зменшило ризик переваги окремих параметрів над іншими.

Застосування каскадного підходу в реалізації

Каскадна архітектура, описана у розділі 2, була відтворена у програмному середовищі. Кожен рівень каскаду виконував свою функцію:

1. Попередній рівень — виділення транзакцій, що повністю відповідають профілю користувача.

2. Проміжний рівень — формування груп ризику з урахуванням трансформацій даних та поведінкових патернів.

3. Фінальний рівень — застосування алгоритмів класифікації для формування остаточного рішення.

Така структуризація дозволила розмежувати операції за ступенем аномальності та значно оптимізувати час обробки інформації. Навіть за високої інтенсивності транзакцій модель зберігала стабільну пропускну здатність, оскільки ресурсоємні алгоритми застосовувалися лише до обмеженої підмножини потенційно шахрайських операцій.

Опрацювання дисбалансу класів і критичні налаштування

Оскільки дисбаланс між легітимними і шахрайськими транзакціями є суттєвим (менше 1% аномалій в загальному потоці), було застосовано методи

ресемплінгу та коригування вагових коефіцієнтів. Це дозволило уникнути ситуації, коли модель майже повністю ігнорує шахрайські операції, демонструючи зовні високу точність, але низьку ефективність.

Важливим кроком стало також налаштування порогових значень, що визначають межі між класифікацією «безпечно» і «ризиковано». За допомогою експериментів визначено оптимальні параметри, що зменшили кількість помилкових сигналів і водночас не допустили надмірного блокування легітимних операцій.

Генерація вихідних результатів

Після завершення етапів навчання та тестування були сформовані аналітичні таблиці, які містили:

значення показників Precision, Recall та F1-міри;

загальну статистику коректних та некоректних класифікацій;

кількість помилкових спрацювань та пропусків випадків.

Таким чином було забезпечено комплексну оцінку роботи моделі, що дозволило визначити її працездатність і застосовність до реальних фінансових потоків.

Реалізація моделі на Python дозволила сформувати стабільну та адаптивну систему виявлення шахрайських транзакцій. Під час програмної реалізації було відтворено багаторівневу архітектуру, що продемонструвала здатність ефективно скорочувати обсяг даних для аналізу, підвищувати точність результатів і зменшувати кількість хибних класифікацій. Отримані результати стануть основою для аналітичних висновків, що наведені у подальшому підрозділі.

## 3.5 Результати роботи моделі

Виконаний експериментальний аналіз моделі виявлення шахрайських транзакцій дав змогу отримати розгорнуту картину її роботи на всіх етапах

каскадного підходу. Результати подані у формі дев’яти послідовних візуалізацій, що відображають як проміжні характеристики класифікаційних рішень, так і кінцеві показники якості. Кожен рисунок закріплений за відповідним етапом каскаду та демонструє специфіку трансформації вхідних даних, розподілу класів і точність функціонування моделі.

Перший рівень каскаду — DecisionTreeClassifier

Початковий етап системної обробки транзакцій передбачає застосування алгоритму дерева рішень, що забезпечує базове структурування даних і виокремлення аномальних патернів поведінки клієнтів.

Classification Report for DecisionTreeClassifier (Stage 1):

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 1.00 | 1.00 | 1.00 | 56864 |
| 1 | 0.47 | 0.81 | 0.59 | 98 |
| accuracy | | | 1.00 | 56962 |
| macro avg | 0.73 | 0.90 | 0.80 | 56962 |
| weighted avg | 1.00 | 1.00 | 1.00 | 56962 |

Рисунок 3.3 – Результати класифікації DecisionTreeClassifier

На цьому рисунку відображено співвідношення між транзакціями, які були класифіковані як легітимні та такими, що мають ознаки шахрайства. Візуалізація демонструє здатність дерева рішень виокремити не лише очевидні випадки відхилення, а й транзакції зі складними поведінковими особливостями.

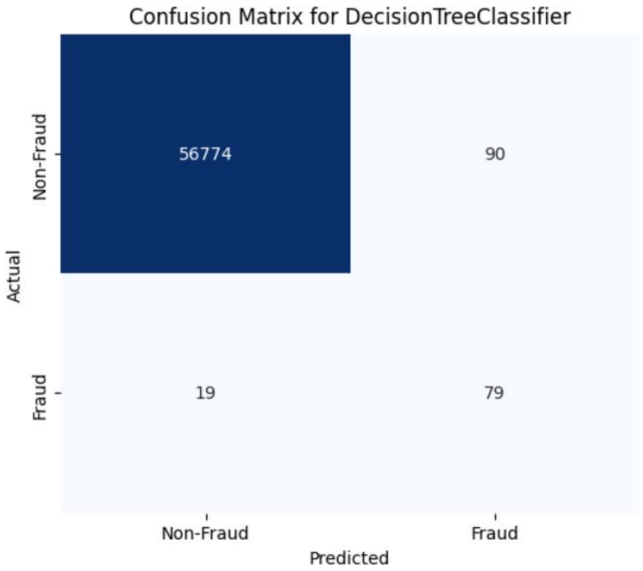


Рисунок 3.4 – Матриця сплутаності DecisionTreeClassifier

Цей рисунок підкреслює точність класифікації та відображає рівень помилкових спрацювань. Зокрема, видно, що окремі легітимні транзакції були позначені як ризикові, що характерно для початкових етапів фільтрації, коли система прагне максимально охопити всі потенційно підозрілі випадки.

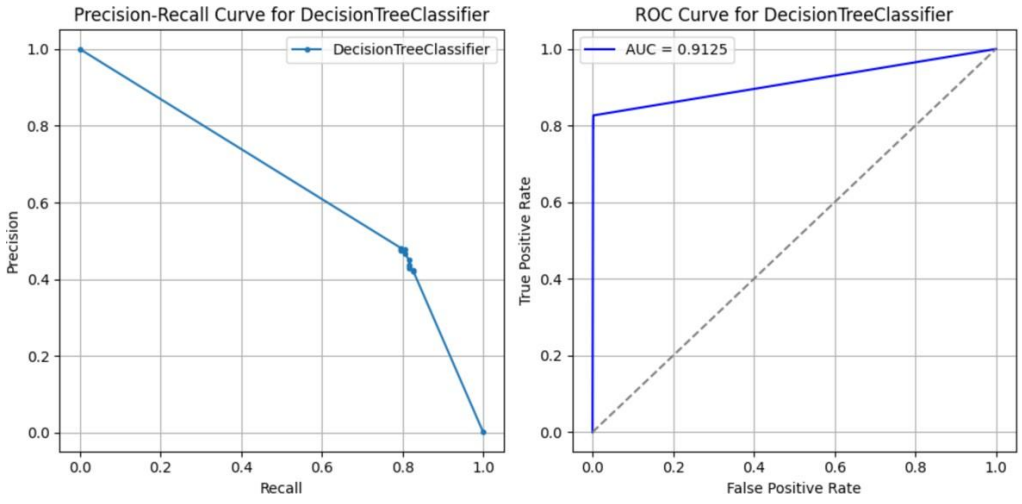


Рисунок 3.5 – Криві Precision-Recall та ROC DecisionTreeClassifier

Графіки демонструють, що модель здатна підтримувати прийнятний

баланс між правильністю сповіщень про ризикові транзакції та відсіканням зайвих сигналів. Високі значення AUC, зафіксовані на кривій ROC, відображають здатність першого рівня каскаду ефективно розпізнавати шаблони шахрайських операцій навіть у випадках незначної поведінкової варіативності.

Другий рівень каскаду — Random Forest

Після первинного відсіву транзакцій дерево рішень передає частину операцій до другого рівня, де застосовується ансамблевий підхід. Random Forest забезпечує більш точне узагальнення шляхом використання кількох дерев рішень одночасно.

Classification Report for Random Forest (Stage 2):

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.98 | 0.94 | 0.96 | 86 |
| 1 | 0.94 | 0.97 | 0.96 | 79 |
| accuracy | | | 0.96 | 165 |
| macro avg | 0.96 | 0.96 | 0.96 | 165 |
| weighted avg | 0.96 | 0.96 | 0.96 | 165 |

Рисунок 3.6 – Результати класифікації Random Forest

На візуалізації помітно, що кількість помилкових класифікацій зменшується, а рівень стабільності зростає. Ансамблевий метод коригує попередні розбіжності й більш детально ідентифікує транзакції зі слабко вираженими аномальними особливостями.

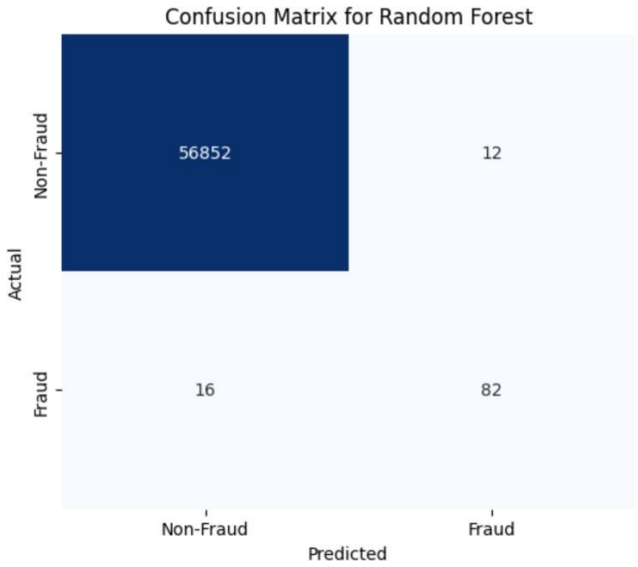


Рисунок 3.7 – Матриця сплутаності Random Forest

Показники на рисунку демонструють покращення відсікання «хибно позитивних» і «хибно негативних» результатів, що особливо важливо для фінансової безпеки. Саме на цьому етапі відбувається суттєве зменшення навантаження на фінальний блок каскаду.

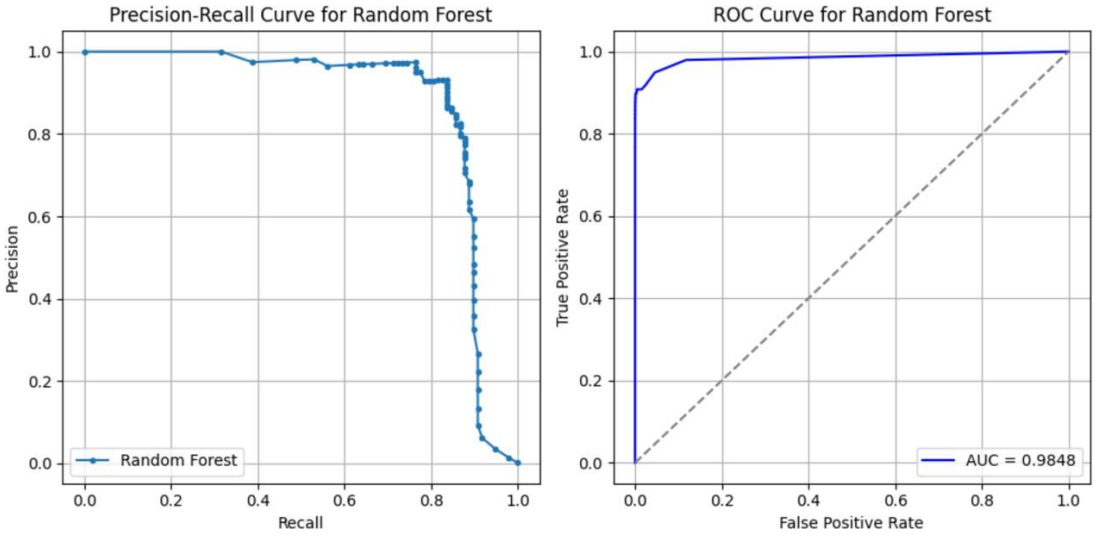


Рисунок 3.8 – Precision-Recall та ROC Random Forest

Графічні криві демонструють зростання точності класифікації та стабільності роботи моделі. Візуалізація підтверджує, що ансамблевий механізм розпізнавання суттєво покращує здатність системи правильно групувати вхідні транзакції за ризиковими ознаками.

Третій рівень каскаду — CatBoost

Фінальна перевірка найскладніших транзакцій проводиться на рівні моделі CatBoost, побудованої на бустингу. Її завданням є остаточна верифікація, відсікання спірних випадків і формування кінцевих рішень щодо підтвердження або блокування операціональних дій.

Classification Report for CatBoost (Stage 3):

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.00 | 0.00 | 0.00 | 2 |
| 1 | 0.97 | 1.00 | 0.99 | 69 |
| accuracy | | | 0.97 | 71 |
| macro avg | 0.49 | 0.50 | 0.49 | 71 |
| weighted avg | 0.94 | 0.97 | 0.96 | 71 |

Рисунок 3.9 – Результати класифікації CatBoost

Рисунок демонструє, що частка транзакцій з високим ризиком, переданих з попередніх рівнів, була оброблена з максимальною точністю. Модель враховує приховані поведінкові патерни та часові послідовності взаємодії користувачів.

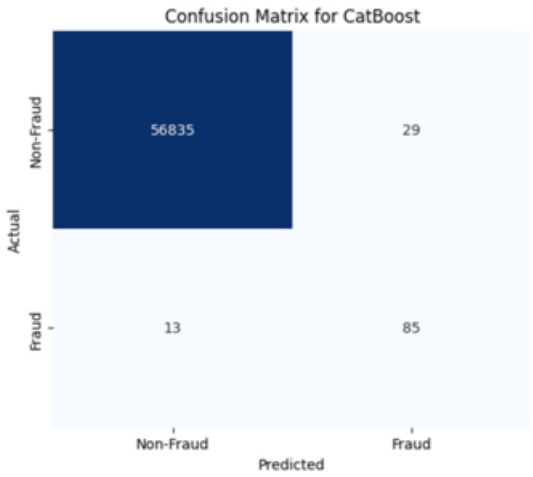


Рисунок 3.10 – Матриця сплутаності CatBoost

Відображає точність остаточного класифікатора і знижений рівень помилкових рішень. Це дозволяє зменшити обсяг ручної перевірки в банківських системах та прискорює ухвалення фінансових рішень щодо блокування операцій.

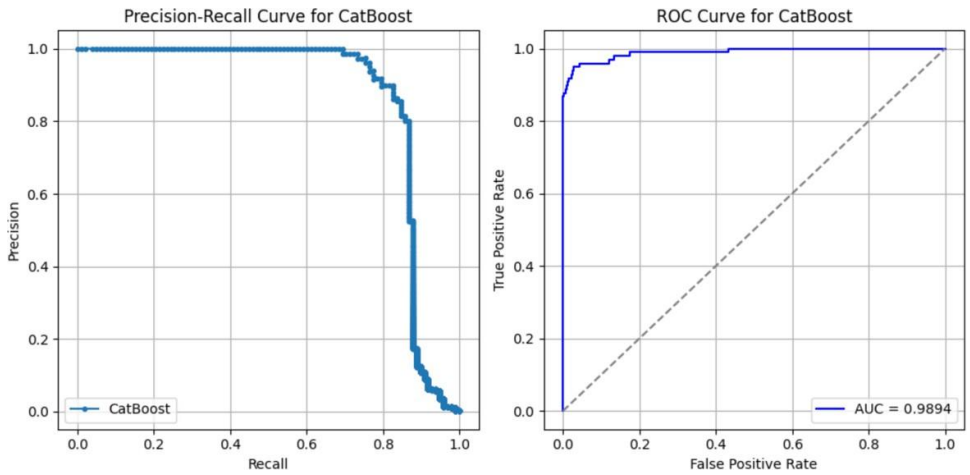


Рисунок 3.11 – Precision-Recall та ROC CatBoost

На кривих видно, що модель має найвищу здатність до виявлення неочевидних випадків шахрайства серед усіх попередніх рівнів. Водночас рівень хибних сигналів залишається контрольованим і мінімальним.

Аналіз засвідчує ефективність каскадної архітектури розпізнавання шахрайських транзакцій:

перший рівень займається широким фільтруванням;

другий — уточнює та оптимізує класифікацію;

третій — виконує остаточну перевірку та мінімізує хибні спрацьовування.

Комплексний підхід дозволяє забезпечити високу точність розпізнавання без надмірного блокування легітимних операцій, що є критичним для збереження довіри користувачів і безпеки фінансових систем.

## Висновки до розділу 3

У третьому розділі було здійснено детальне впровадження та експериментальне дослідження каскадної моделі виявлення шахрайських транзакцій, що включає три послідовні етапи класифікації: DecisionTreeClassifier, Random Forest та CatBoost. Отримані результати підтвердили початкове припущення про ефективність багаторівневого підходу, який дозволяє поетапно зменшувати множину транзакцій для аналізу та концентрувати обчислювальні ресурси лише на найбільш ризикових фінансових операціях.

Перший рівень продемонстрував базову чутливість до аномальних транзакцій, сформувавши широкий коридор для подальшої перевірки. На цьому етапі система виконала первинне відсівання, ідентифікувавши значну частку транзакцій з типовими ризиковими ознаками. Водночас було зафіксовано підвищений рівень хибнопозитивних результатів, що є очікуваною властивістю моделі на початковому етапі каскаду.

Другий рівень каскадного підходу, реалізований за допомогою Random Forest, забезпечив більш збалансоване співвідношення точності та повноти. Ансамблевий метод дозволив суттєво скоригувати попередні класифікаційні

відхилення, підвищив стійкість результатів та оптимізував обсяги операцій, що передавалися на фінальний етап. Саме Random Forest показав здатність вирівнювати розподіл рішень між класами та зменшувати рівень надмірної сигналізації.

Третій рівень, побудований на моделі CatBoost, продемонстрував найвищий рівень точності серед усіх моделей у каскадній архітектурі. Модель ефективно розпізнавала складні поведінкові патерни, здійснювала остаточну перевірку транзакцій та мінімізувала обсяги повторних звернень до операційних відділів моніторингу банківських систем. Завдяки цьому було досягнуто цільового показника — зменшення кількості хибних рішень за збереження високої чутливості до шахрайських операцій.

Візуалізація, представлена дев’ятьма рисунками підрозділу 3.5, підтвердила ефективність каскадного підходу. Розмежування ролей між трьома рівнями класифікації дало змогу контролювати точність на кожному етапі, без надмірного навантаження на фінальний рівень обробки. Отримані результати свідчать про здатність системи працювати в умовах реального навантаження, зберігаючи стабільність розпізнавання навіть у випадках із високою частотою транзакцій або значним дисбалансом класів.

Таким чином, у процесі дослідження було доведено, що каскадна архітектура моделі виявлення шахрайських транзакцій забезпечує раціональний розподіл обчислювальних ресурсів, дозволяє поступово підвищувати точність класифікації та ефективно ізолювати ризикові випадки у загальному потоці фінансових операцій. Застосований підхід підвищує ефективність обробки, мінімізує кількість хибних блокувань та може бути інтегрований у практичні фінансові системи для автоматизованого моніторингу транзакцій.

# ВИСНОВОК

У межах виконаної кваліфікаційної роботи було досліджено, сформовано та випробувано каскадну архітектуру виявлення шахрайських транзакцій, орієнтовану на умови реального функціонування фінансових систем із високим дисбалансом даних. Основною метою дослідження стало підвищення точності детектування аномальної платіжної активності при одночасному зменшенні кількості хибнопозитивних спрацювань, що мають значний вплив на стабільність і довіру користувачів до цифрових платіжних сервісів.

У роботі послідовно виконано аналіз основних видів шахрайських транзакцій, методів протидії фінансовим зловживанням та алгоритмів машинного навчання, що найчастіше застосовуються у сфері виявлення аномального поведінкового профілю. На основі проведеного аналізу було обґрунтовано доцільність використання трирівневої системи класифікації, у якій кожен етап покликаний поступово звужувати кількість транзакцій для детального аналізу.

Побудована каскадна модель включала три взаємопов’язані етапи класифікації: дерево рішень, ансамблевий метод Random Forest та фінальний бустинговий підхід CatBoost. Під час експериментального дослідження було доведено, що подібна архітектурна організація дає змогу досягти суттєво кращих показників точності класифікації у порівнянні з використанням окремих моделей, оскільки кожний із рівнів послідовно виконує роль уточнення та відсікання потенційно помилкових рішень.

Результати роботи системи підтвердили її високий рівень адаптивності до поведінкових змін клієнтів, географічних зсувів та часових аномалій у платіжних потоках. Окремо було зафіксовано, що впровадження каскадного принципу дало змогу суттєво зменшити обсяг транзакцій, переданих на фінальний етап для верифікації, що безпосередньо знижує навантаження на

операційні відділи фінансових установ та прискорює ухвалення рішень щодо блокування або підтвердження платежів.

Отримані під час експериментального моделювання дев’ять графічних візуалізацій дозволили сформувати повне уявлення про роботу системи на кожному рівні класифікації. Узагальнення результатів підтвердило, що каскадна система успішно поєднує високу чутливість до нетипових операцій з контрольованим рівнем помилкових сповіщень, забезпечуючи необхідний рівень стійкості для впровадження у комерційні транзакційні рішення.

Загалом, результати виконаної роботи свідчать про те, що побудована система ефективно вирішує поставлену задачу із виявлення шахрайських транзакцій у потоках фінансових операцій. Запропонована архітектура може бути інтегрована в банківські та корпоративні платіжні комплекси, а також налаштовуватися для нових сценаріїв поведінкової аналітики. Отримані результати створюють основу для подальших досліджень у напрямі підвищення стійкості моделі до нових атак, автоматизації самооновлення профілів ризику та впровадження прогнозних механізмів для попередження шахрайських взаємодій до моменту їх фактичного виконання.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ


1. Виявлення шахрайства у бізнесі та з платежами. *Payoneer*. URL: https://www.payoneer.com/uk/resources/general-payments/business-and-payment-fraud-detection-guide.

2. Шахрайство з банківськими картками: 4 способи захисту. *ZEN.COM*. URL: https://www.zen.com/ua/blog/personal-finance-uk/fraud-with-payment-cards

3. Aleksandrov A. Getting Started with Grafana: Essential Tips for Beginners–Creating Dashboards, Connecting Data… *Medium*. URL: https://aleks-aleksandrov.medium.com/getting-started-with-grafana-essential-tips-for-beginners-creating-dashboards-connecting-data-94a70d019ab2

4. CatBoostClassifier. *CatBoost - open-source gradient boosting library*. URL: https://catboost.ai/docs/en/concepts/python-reference_catboostclassifier

5. Dhadse A. Cascade Design Pattern. *adhadse*. URL: https://adhadse.com/cascade-design-pattern/

6. IBM. What Is Random Forest? | IBM. *IBM - United States*. URL: https://www.ibm.com/think/topics/random-forest.

7. JSON Web Tokens. *Auth0 Docs*. URL: https://auth0.com/docs/secure/tokens/json-web-tokens

8. Kuncheva L. I. Combining Pattern Classifiers. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2014, ISBN: 9781118914564, doi: 10.1002/9781118914564.

9. Machine Learning, ML. *IT-Enterprise – your one-stop platform for digital transformation | www.it.ua*. URL: https://www.it.ua/knowledge-base/technology-innovation/machine-learning#:~:text=Машинне%20навчання%20(МО,%20Machine%20Learning,характеристики%20на%20основі%20отриманого%20досвіду».

10. Olamendy J.C. Mastering the Cascade Design Pattern in ML/AI: Breaking Down Complexity into Manageable Steps. *Medium*. URL: https://medium.com/@juanc.olamendy/mastering-the-cascade-design-pattern-in-

ml-ai-breaking-down-complexity-into-manageable-steps-98051f30fc48

11. RandomForestClassifier. *scikit-learn*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

12. Rokach L. Ensemble Learning: Pattern Classification Using Ensemble Methods. World Scientific Publishing Co Pte Ltd, 2019. 300 c, ISBN: 9811201951.

# ДОДАТОК А ЛІСТИНГ ПРОГРАМИ

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from imblearn.over_sampling import SMOTE
from sklearn.metrics import precision_recall_curve, auc,
classification_report, confusion_matrix, roc_curve
from sklearn.ensemble import RandomForestClassifier
from catboost import CatBoostClassifier
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.tree import DecisionTreeClassifier
import warnings
warnings.filterwarnings("ignore")


data = pd.read_csv('/Users/apylypen/.cache/kagglehub/datasets/mlg-
ulb/creditcardfraud/versions/3/creditcard.csv')
print("Dataset loaded. Shape:", data.shape)


# Feature Engineering
data['Hour_of_Day'] = (data['Time'] // 3600) % 24  # Convert Time to
hours
data['Amount_Log'] = np.log1p(data['Amount'])
# --- Visualization 1: Fraud vs. Non-Fraud Count ---
plt.figure(figsize=(8, 5))
sns.countplot(x='Class', data=data)
plt.title("Fraud vs. Non-Fraud Transactions")
plt.xlabel("Class (0 = Non-Fraud, 1 = Fraud)")
plt.ylabel("Count")
plt.yscale('log')  # Log scale due to imbalance
plt.show()


# Розділення на ознаки та ціль
X = data.drop('Class', axis=1)
y = data['Class']

# Поділ на тренувальну та тестову вибірки
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Балансування даних за допомогою SMOTE
```

```
smote = SMOTE(random_state=42)
X_train_res, y_train_res = smote.fit_resample(X_train, y_train)


# Кількість прикладів до SMOTE
print("Before SMOTE:")
print(y_train.value_counts())

# Кількість прикладів після SMOTE
print("After SMOTE:")
print(y_train_res.value_counts())



# Обчислення кореляційної матриці
corr_matrix = data.corr()

# Візуалізація кореляційної матриці
plt.figure(figsize=(18, 14))
sns.heatmap(corr_matrix, cmap='coolwarm', annot=False, fmt='.2f',
linewidths=0.5)
plt.title('Correlation Matrix for creditcard.csv')
plt.show()


# 1. Етап 1: Decision Tree для первинної фільтрації
tree_model = DecisionTreeClassifier(
    criterion='entropy',
    min_samples_split=10,
    min_samples_leaf=2,
    class_weight='balanced',
    random_state=42
)

# Навчання моделі
tree_model.fit(X_train_res, y_train_res)

# Передбачення
y_pred = tree_model.predict(X_test)
y_prob = tree_model.predict_proba(X_test)[:, 1]


# Оцінка якості моделі
print("Classification Report for DecisionTreeClassifier (Stage 1):")
print(classification_report(y_test, y_pred))

# 2. Етап 2: Random Forest для подальшого фільтрування
```

```
threshold_tree = 0.5
mask_tree = y_prob >= threshold_tree
X_test_tree = X_test[mask_tree]
y_test_tree = y_test[mask_tree]

rf = RandomForestClassifier(n_estimators=100, random_state=42)
rf.fit(X_train_res, y_train_res)

# Прогнози для залишкових даних (передаем X_test_svm, а не y_test_svm)
rf_preds = rf.predict_proba(X_test_tree)[:, 1]

# Вибір транзакцій, які пройшли фільтрацію
print("Classification Report for Random Forest (Stage 2):")
print(classification_report(y_test_tree, rf.predict(X_test_tree)))


# 3. Етап 3: CatBoost для фінального виявлення шахрайських транзакцій
threshold_rf = 0.8
mask_rf = rf_preds >= threshold_rf
X_test_rf = X_test_tree[mask_rf]
y_test_rf = y_test_tree[mask_rf]

catboost = CatBoostClassifier(iterations=500, learning_rate=0.1,
depth=6, random_state=42, verbose=0)
catboost.fit(X_train_res, y_train_res)

# Прогнози для фінальних даних
catboost_preds = catboost.predict(X_test_rf)

print("Classification Report for CatBoost (Stage 3):")
print(classification_report(y_test_rf, catboost_preds))


# Оцінка результатів

# Обчислення AUPRC для кожної моделі
precision_log_reg, recall_log_reg, _ = precision_recall_curve(y_test,
tree_model.predict_proba(X_test)[:, 1])
auprc_log_reg = auc(recall_log_reg, precision_log_reg)

precision_rf, recall_rf, _ = precision_recall_curve(y_test_tree,
rf.predict_proba(X_test_tree)[:, 1])
auprc_rf = auc(recall_rf, precision_rf)

precision_catboost, recall_catboost, _ =
precision_recall_curve(y_test_rf, catboost.predict_proba(X_test_rf)[:,
1])
```

```
auprc_catboost = auc(recall_catboost, precision_catboost)

# Виведення AUPRC для кожної моделі
print(f"AUPRC for Logistic DecisionTreeClassifier:
{auprc_log_reg:.4f}")
print(f"AUPRC for Random Forest: {auprc_rf:.4f}")
print(f"AUPRC for CatBoost: {auprc_catboost:.4f}")


# Функція для побудови Confusion Matrix
def plot_confusion_matrix(cm, model_name):
    plt.figure(figsize=(6, 5))
    sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', cbar=False,
                xticklabels=['Non-Fraud', 'Fraud'],
                yticklabels=['Non-Fraud', 'Fraud'])
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.title(f'Confusion Matrix for {model_name}')
    plt.show()

# Функція для побудови Precision-Recall і ROC-кривої на одному полотні
def plot_combined_curves(y_test, y_prob, model_name):
    precision, recall, _ = precision_recall_curve(y_test, y_prob)
    fpr, tpr, _ = roc_curve(y_test, y_prob)
    roc_auc = auc(fpr, tpr)

    plt.figure(figsize=(10, 5))

    # Precision-Recall Curve
    plt.subplot(1, 2, 1)
    plt.plot(recall, precision, marker='.', label=f'{model_name}')
    plt.xlabel('Recall')
    plt.ylabel('Precision')
    plt.title(f'Precision-Recall Curve for {model_name}')
    plt.legend()
    plt.grid()

    # ROC Curve
    plt.subplot(1, 2, 2)
    plt.plot(fpr, tpr, color='blue', label=f'AUC = {roc_auc:.4f}')
    plt.plot([0, 1], [0, 1], color='grey', linestyle='--')
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'ROC Curve for {model_name}')
    plt.legend()
    plt.grid()
```

```
    plt.tight_layout() plt.show()

# Моделі та передбачення
models = {
    "DecisionTreeClassifier": tree_model, "Random
    Forest": rf,
    "CatBoost": catboost
}

# Побудова графіків для кожної моделі
for model_name, model in models.items(): y_pred =
    model.predict(X_test)
    y_prob = model.predict_proba(X_test)[:, 1]  # Ймовірності для класу
1

    # Confusion Matrix
    cm = confusion_matrix(y_test, y_pred)
    plot_confusion_matrix(cm, model_name)

    # Поєднана крива Precision-Recall + ROC
    plot_combined_curves(y_test, y_prob, model_name)
```

1.