МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Острозька академія»
Навчально-науковий інститут інформаційних технологій та бізнесу
Кафедра інформаційних технологій та аналітики даних

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття освітнього ступеня магістра

на тему: «Управління проєктом з впровадження моделі виявлення шахрайських операцій у банкінгу»

Виконала: студентка 2 курсу, групи МУП-21
другого (магістерського) рівня вищої освіти
спеціальності 122 Комп'ютерні науки
освітньо-професійної програми «Управління проєктами»
Черній Анастасія Олексіївна

Керівник: ***старший викладач,***
Клебан Юрій Вікторович

Рецензент: ***кандидат технічних наук, доцент,***
доцент кафедри прикладної математики
Донецького національного університету
імені Василя Стуса
Загоруйко Любов Василівна

РОБОТА ДОПУЩЕНА ДО ЗАХИСТУ
Завідувач кафедри інформаційних технологій та аналітики даних
_____ (проф., д.е.н. Кривицька О.Р.)
Протокол № 5 від 04 грудня 2025

Острог, 2025

АНОТАЦІЯ
кваліфікаційної роботи
на здобуття освітнього ступеня магістра

Тема: Управління проєктом з впровадження моделі виявлення шахрайських операцій у банкінгу
Автор: Анастасія ЧЕРНІЙ
Науковий керівник: Юрій КЛЕБАН
Захищена «……»…… 2025 року.

Пояснювальна записка до кваліфікаційної роботи: 151 (кількість сторінок роботи) с., 11 (кількість рисунків) рис., 12 (кількість таблиць) табл., 18 (кількість додатків) додатків, 204 (кількість джерел) джерел.

Ключові слова: УПРАВЛІННЯ ПРОЄКТАМИ, БАНКІВСЬКЕ ШАХРАЙСТВО, ВПРОВАДЖЕННЯ ІТ-СИСТЕМ, МАШИННЕ НАВЧАННЯ, ПРОЄКТНИЙ МЕНЕДЖМЕНТ, ЦИФРОВА ТРАНСФОРМАЦІЯ, РИЗИК-МЕНЕДЖМЕНТ, БАНКІВСЬКІ ТЕХНОЛОГІЇ

Короткий зміст праці:

Кваліфікаційна робота присвячена дослідженню та розробці методології управління проєктом впровадження моделі виявлення шахрайських операцій у банківській установі. Проаналізовано сутність та класифікацію шахрайських операцій, досліджено сучасні підходи до управління ІТ-проєктами у фінансовому секторі.

Розглянуто світовий досвід впровадження систем виявлення банківського шахрайства та популярні проєктні методології. Досліджено особливості управління проєктами з машинного навчання у банківській сфері, включаючи ризики, ресурси та стейкхолдерів.

Розроблено комплексний план управління проєктом впровадження моделі виявлення шахрайства, створено структуру декомпозиції робіт, визначено критичний шлях проєкту, розроблено систему управління ризиками та контролю якості.

Запропоновано методи оцінки ефективності впровадження системи та стратегії управління змінами в організації. Розроблено рекомендації щодо вдосконалення процесів управління подібними проєктами в банківських установах.

Практичне значення роботи полягає у створенні структурованого підходу до управління проєктами впровадження систем штучного інтелекту у банкінгу, що дозволить підвищити успішність таких ініціатив та знизити проєктні ризики.

ABSTRACT
of the qualification work
for obtaining a master's degree

***Topic*:** Project management for the implementation of a fraud detection model in banking

Author: Anastasiia Chernii

Scientific supervisor: Yuriy Kleban

Protected «……»…… 20____ .

Explanatory note to the qualification work: 151 (number of pages) p., 11 (number of figures) figures, 12 (number of tables) tables, 18 (number of appendices) appendices, 204 (number of sources) sources.

***Keywords*:** PROJECT MANAGEMENT, BANKING FRAUD, IT SYSTEM IMPLEMENTATION, MACHINE LEARNING, PROJECT MANAGEMENT, DIGITAL TRANSFORMATION, RISK MANAGEMENT, BANKING TECHNOLOGIES

Summary of the work:

The qualification work is devoted to research and development of project management methodology for implementing fraud detection model in a banking institution. The essence and classification of fraudulent operations are analyzed, modern approaches to IT project management in the financial sector are studied.

The world experience of implementing banking fraud detection systems and popular project methodologies are considered. The peculiarities of managing machine learning projects in the banking sector are studied, including risks, resources and stakeholders.

A comprehensive project management plan for implementing fraud detection model has been developed, work breakdown structure has been created, critical path of the project has been determined, risk management system and quality control have been developed.

Methods for evaluating system implementation effectiveness and organizational change management strategies are proposed. Recommendations for improving management processes of similar projects in banking institutions have been developed.

The practical significance of the work lies in creating a structured approach to managing artificial intelligence system implementation projects in banking, which will increase the success rate of such initiatives and reduce project risks.

ЗМІСТ

ВСТУП

Мета дослідження полягає у дослідженні підходів, розробці та впровадженні проєкту з імплементації моделі виявлення шахрайських операцій у для систем цифрового банкінгу.

Об'єктом дослідження є процес управління проєктом впровадження системи виявлення шахрайських операцій у цифровому банкінгу.

Предметом дослідження є теоретико-методологічні засади та практичні аспекти розробки й впровадження моделей виявлення шахрайських операцій в системах цифрового банкінгу.

Завдання роботи:

1. Дослідити теоретичні основи та провести класифікацію шахрайських операцій у системах цифрового банкінгу
2. Проаналізувати існуючі методологічні підходи до моделювання систем виявлення шахрайства в реальному часі
3. Розробити математичну модель та алгоритми виявлення шахрайських операцій у реальному часі
4. Здійснити оптимізацію параметрів розробленої моделі та оцінити її ефективність
5. Розробити алгоритм впровадження системи виявлення шахрайських операцій у банківську інфраструктуру
6. Визначити перспективи розвитку моделей виявлення шахрайства у банкінгу

В процесі дослідження використано наступні методи:

1. Дослідницький аналіз даних (EDA)
2. Методи математичного моделювання
3. Методи машинного навчання та обробки даних у реальному часі
4. Методи проєктного менеджменту
5. Методи статистичного аналізу
6. Порівняльний аналіз

У сучасному світі банківська справа зазнала кардинальних перетворень, перетворившись з консервативної системи на динамічну, доступну та надзвичайно зручну для клієнтів екосистему. Те, що раніше вимагало тривалого очікування в чергах та особистої присутності у філіях, сьогодні вміщується в один дотик на екрані смартфона.

Розвиток економіки кожної держави значною мірою визначається тим, наскільки активно суспільство інтегрується в цифровий простір. У наш час ключовим вектором інноваційних процесів у підприємництві стало перенесення бізнес-операцій у віртуальне середовище. Статистика свідчить, що щороку від третини до двох третин підприємств у різних країнах світу (незалежно від їхнього економічного розвитку) починають функціонувати в онлайн-форматі.[1] Це означає, що все більше компаній впроваджують електронні комерційні платформи для здійснення своєї господарської діяльності.

Цифровий банкінг - це не просто технологічне нововведення, а справжня революція у фінансовій взаємодії. Він народився з прагнення зробити банківські послуги максимально простими, швидкими та зручними. Від класичних онлайн-транзакцій до миттєвих міжнародних переказів - сучасні технології повністю змінили уявлення про фінансові послуги. Еволюція розпочалася ще в 1960-х роках з перших комп'ютерних систем у банках. Однак справжній прорив стався в 1967 році, коли британський Barclays встановив перший у світі банкомат у Лондоні. Цей момент можна вважати народженням цифрового банкінгу.

Сьогодні цифровий банкінг включає широкий спектр послуг: від класичного онлайн-банкінгу та мобільних додатків до електронних гаманців та миттєвих платежів. Клієнти можуть керувати рахунками, здійснювати перекази, сплачувати рахунки та навіть оформляти кредити, не виходячи з дому.

У статті «E-banking Overview: Concepts, Challenges and Solutions» [2] автори наголошують на багатогранності електронного банкінгу, яка включає інтернет-платформи, мобільні додатки, термінали самообслуговування та інші цифрові канали комунікації. Особливістю сучасних електронних банківських систем, за їхнім твердженням, є не лише забезпечення традиційних фінансових операцій,

але й створення персоналізованого клієнтського досвіду. Критичними викликами впровадження електронного банкінгу є питання кібербезпеки, захисту персональних даних та мінімізації ризиків шахрайських атак.

Згідно із Національним банком України, за 2024 рік в Україні суттєво погіршилась ситуація з платіжним шахрайством. Як свідчать дані за 2023 рік, кількість незаконних операцій з картками зросла майже на 25% і досягла 272 тисяч випадків. Фінансові втрати виявились ще відчутнішими - загальна сума збитків склала близько 833 мільйонів гривень, що на 73% більше порівняно з 2022 роком. Середній "чек" шахраїв також збільшився до 3065 гривень (+39%). Характерно, що більшість шахрайств (83%) відбувається в інтернеті, і лише кожне шосте - через банкомати чи термінали. Найбільшу небезпеку становить соціальна інженерія - на неї припадає 80% усіх збитків, що значно більше ніж торік (53%). Шахраї активно використовують фейкові повідомлення про державні виплати, виготовляють дублікати SIM-карт, телефонують під виглядом "служби безпеки банку" та розповсюджують обманні схеми через месенджери. Цікаво, що з 2023 року до статистики включили не лише банки, а й небанківські установи та Укрпошту, хоча їхня частка в загальному обсязі шахрайств виявилась незначною.[3]

Дослідники по-різному підходили до класифікації кібератак у банківській сфері. Частина дослідників використали модель дерева атак, щоб показати взаємозв'язки між різними видами загроз - від фішингу та соціальної інженерії до шкідливого ПЗ для контролю над пристроями та створення фальшивих веб-сторінок для крадіжки облікових даних. Існує також більш розширена класифікація, яка включає сканери портів, атаки соціальної інженерії, фішинг, трояни, фармінг, відмову в обслуговуванні, злом PIN-коду, експлойти для суперкористувачів і помилки сервера. Цікавий погляд представлено у розрізі поділу атак на локальні, віддалені та гібридні, де локальні атаки можуть виглядати як легітимні банківські сторінки зі справжнім URL та SSL-сертифікатом, але насправді використовувати підроблені форми для крадіжки паролів.[4]

Одним із найпопулярніших видів шахрайства в сфері електронного банкінгу є фішинг, при якому використовується імітація для отримання інформації від цілі. Фішинг є формою соціальної інженерії, коли шахрай намагається обманним шляхом отримати конфіденційні дані легітимних користувачів, імітуючи електронні комунікації або телефонні дзвінки від надійної чи публічної організації в автоматизований спосіб.

Це може бути як і телефонний фішинг, тобто телефонний дзвінок від імені співробітника банку або служби безпеки, під час якого просять надати CVV-код картки або інші дані. Або ж електронний фішинг, що має на увазі лист на електронну пошту клієнта, підписаний його обслуговуючим банком, з пропозицією перейти за посиланням на аналог особистого кабінету та ввести логін і пароль.[5]

Згідно із статтею «Electronic Banking (e-Banking) Fraud with Phishing Attack Methods» [1] аналіз кейсів фішингових атак пов'язаний з певними труднощами. Оскільки фішингова атака може використовувати різні схеми відповідно до обраної вигадки, підхід до виявлення та аналізу може відрізнятися в залежності від випадку, хоча він і схожий на інші приклади атак. З цієї причини криміналістичні дослідження випадків роблять великий внесок у роботу, що проводиться в цій галузі, і в боротьбу з фішинговими атаками.

Банки активно борються із цими явищами, застосовуючи найрізноманітніші методи. Яскравим прикладом є сервіс Fraud Payments Tracker, який дозволяє банкам оперативно обмінюватися інформацією про підозрілі транзакції. Цей інструмент особливо ефективний завдяки можливості як автоматизованої роботи через API-інтерфейси, так і ручного управління через веб-інтерфейс. Важливим кроком у розвитку такої співпраці стало приєднання Монобанку до системи у грудні 2021 року, що розширило мережу банків, здатних миттєво реагувати на несанкціоновані платежі. Система працює за чітким алгоритмом: банк жертви надсилає інформацію про підозрілі перекази, система визначає банк-отримувач та передає дані, а той, у свою чергу, перевіряє інформацію та вживає необхідних

заходів. Такий підхід дозволяє ефективно протидіяти шахрайським операціям у банківській системі України, Молдови та Казахстану.[6]

Моделювання, особливо з використанням машинних методів, дозволяє виявляти приховані закономірності у великих масивах транзакцій та адаптуватися до нових схем шахрайства. Це забезпечує швидке автоматизоване виявлення підозрілих операцій у режимі реального часу, мінімізуючи фінансові втрати. Важливою перевагою є здатність моделей навчатися на історичних даних та постійно вдосконалювати точність прогнозів, при цьому зменшуючи кількість помилкових спрацювань та підвищуючи довіру користувачів до системи безпеки.

Згідно зі статтею "Виявлення шахрайства з кредитними картками методами машинного навчання"[7], проблема виявлення шахрайства має специфічні виклики через незбалансованість даних (менше 1% шахрайських транзакцій), динамічну природу шахрайства та необхідність швидкого виявлення. Логістична регресія показала найкращі результати серед досліджених класифікаторів (порівняно з K-найближчих сусідів, машиною опорних векторів та деревом рішень). Нейронні мережі показали різні результати для різних підходів: на даних з надмірною вибіркою гірше виявляли шахрайські транзакції, а на даних з недостатньою вибіркою часто помилково класифікували звичайні транзакції як шахрайські.

РОЗДІЛ 1

ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ ЗАСАДИ ДОСЛІДЖЕННЯ ШАХРАЙСЬКИХ ОПЕРАЦІЙ В БАНКІНГУ

1.1. Сутність та класифікація шахрайських операцій у банкінгу

Банкінг є невід'ємною частиною сучасної економіки, забезпечуючи широкий спектр фінансових послуг, які сприяють економічному зростанню та стабільності. Основна мета банківської діяльності полягає у забезпеченні ефективного управління фінансовими ресурсами, наданні кредитів, зберіганні депозитів та здійсненні платіжних операцій. Банки виступають посередниками між вкладниками, які надають свої кошти у вигляді депозитів, та позичальниками, які отримують кредити для фінансування своїх потреб. Це сприяє ефективному розподілу фінансових ресурсів у економіці. Вони здійснюють оцінку та управління ризиками, пов'язаними з кредитуванням та інвестиціями. Це включає аналіз кредитоспроможності позичальників, диверсифікацію портфеля активів та використання фінансових інструментів для хеджування ризиків. Банки надають широкий спектр платіжних послуг, включаючи перекази коштів, оплату рахунків, обробку чеків та електронні платежі. Це сприяє зручності та безпеці фінансових операцій.

Таким чином, банкінг відіграє ключову роль у забезпеченні стабільності та розвитку економіки, надаючи необхідні фінансові послуги та сприяючи ефективному управлінню фінансовими ресурсами. Шахрайські операції у банкінгу становлять складну багатовимірну проблематику, яка вимагає комплексного підходу до аналізу та розуміння. В контексті сучасної банківської діяльності, ці операції являють собою протиправні дії, спрямовані на несанкціоноване отримання фінансових ресурсів шляхом використання обману, маніпуляцій та експлуатації технологічних вразливостей банківських систем і психологічних особливостей клієнтів.

Теоретичну основу для розуміння шахрайських операцій закладає концепція операційного ризику, визначена Базельським комітетом з банківського нагляду.

Згідно з Basel III [8], операційний ризик трактується як "ризик збитків внаслідок недоліків або помилок у внутрішніх процесах, діяльності співробітників, роботі систем або зовнішніх подій". Ця дефініція охоплює широкий спектр загроз, включаючи внутрішнє та зовнішнє шахрайство, які становлять значний компонент загального операційного ризику банківських установ.

Еволюція шахрайських схем тісно пов'язана з технологічним прогресом у банківській сфері. Якщо традиційні форми банківського шахрайства зосереджувалися переважно на фізичному доступі до коштів або документів, то сучасні цифрові шахрайські операції характеризуються використанням складних технологічних рішень, психологічних маніпуляцій та соціальної інженерії. Особливістю таких операцій є їхня масштабованість - зловмисники можуть одночасно атакувати тисячі потенційних жертв, використовуючи автоматизовані системи та штучний інтелект для персоналізації атак.

Сучасні шахрайські схеми також характеризуються міждисциплінарністю, поєднуючи елементи кібербезпеки, психології, соціології та економіки. Зловмисники активно використовують досягнення у сфері поведінкових наук для розробки більш ефективних методів впливу на потенційних жертв, створюючи складні багатоетапні сценарії, що імітують легітимні банківські процедури.

Фішинг залишається найбільш поширеним та ефективним методом шахрайських атак у банківській сфері. За даними Anti-Phishing Working Group, протягом першого кварталу 2024 року було зафіксовано понад мільйон фішингових атак, що робить його найбільш розповсюдженим типом кіберзагроз[9]. Ефективність фішингових кампаній обумовлена декількома ключовими факторами, що робить їх особливо привабливими для зловмисників.

По-перше, сучасні фішингові атаки характеризуються високим рівнем персоналізації. Зловмисники використовують попередньо зібрану інформацію про потенційних жертв з відкритих джерел, соціальних мереж (Instagram, Twitter, LinkedIn, Facebook, т.п.) та витоків даних для створення високоцільових повідомлень, які важко відрізнити від легітимних комунікацій банків. Ця

практика, відома як spear phishing, демонструє значно вищі показники успішності порівняно з масовими нецільовими розсилками.

Технологічна складність сучасних фішингових кампаній включає використання підроблених SSL-сертифікатів (засіб захисту особистої інформації користувачів в інтернеті), що створює ілюзію безпечного з'єднання, маскування доменних імен через використання схожих символів різних алфавітів (punycode attacks), та створення повнофункціональних копій банківських веб-інтерфейсів. Особливу увагу заслуговують фішингові атаки, що використовують QR-коди - цей тренд набув широкого розповсюдження після пандемії COVID-19, коли QR-коди стали звичним способом взаємодії з цифровими сервісами.

Психологічні аспекти фішингових атак базуються на принципах соціальної інженерії та експлуатації когнітивних упереджень. Зловмисники створюють штучне відчуття терміновості, використовують авторитет банківських брендів та експлуатують страхи клієнтів щодо можливої втрати коштів. Ефективність таких підходів підтверджується дослідженнями у сфері поведінкової економіки, які демонструють, що люди часто приймають нераціональні рішення під впливом емоційного стресу.

Соціальна інженерія представляє собою найбільш складну форму шахрайських атак, оскільки її успішність залежить не стільки від технологічних засобів, скільки від глибокого розуміння психологічних особливостей людської поведінки. Кевін Мітнік, один з найвідоміших експертів у сфері соціальної інженерії, у своїх дослідженнях підкреслює, що найслабшою ланкою в будь-якій системі безпеки є людина [10].

Теоретичною основою для розуміння механізмів соціальної інженерії служать принципи впливу, описані Робертом Чалдіні у його фундаментальній праці "Psychology of Persuasion". Ці принципи включають взаємність, зобов'язання та послідовність, соціальне підтвердження, симпатію, авторитет та дефіцит [11]. Зловмисники систематично використовують ці психологічні тригери для створення ситуацій, де жертви добровільно надають конфіденційну інформацію або виконують дії, що суперечать їхнім інтересам безпеки.

Сучасні атаки соціальної інженерії характеризуються багатоетапністю та тривалістю. Зловмисники можуть протягом тижнів або навіть місяців будувати довірчі відносини з потенційними жертвами, поступово збираючи інформацію та створюючи передумови для успішної атаки. Такі операції часто включають детальне вивчення корпоративної структури банків, особистих даних співробітників та клієнтів, що дозволяє створювати надзвичайно переконливі сценарії.

Особливою формою соціальної інженерії є вішинг (голосовий фішинг) - телефонні шахрайські дзвінки, під час яких зловмисники видають себе за представників банків або інших фінансових установ. Ефективність таких атак обумовлена тим, що голосова комунікація створює більше відчуття автентичності порівняно з текстовими повідомленнями. Зловмисники використовують професійні call-центри, підробляють номери телефонів банків через ID абонента та володіють детальною інформацією про банківські процедури, що робить їх надзвичайно переконливими.

Шкідливе програмне забезпечення, орієнтоване на банківську сферу, пройшло значну еволюцію від простих кейлогерів (програмне забезпечення або апаратний пристрій, що реєструє різноманітні дії користувача на клавіатурі комп'ютера) до складних багатофункціональних платформ, здатних виконувати широкий спектр зловмисних операцій. Сучасні банківські троянські програми, такі як Zeus, Emotet, Trickbot та їхні численні модифікації, представляють собою складні програмні комплекси з модульною архітектурою, що дозволяє зловмисникам адаптувати їх функціональність до специфічних потреб атаки.

Технологічна складність сучасного банківського малвару (зловмисного програмного забезпечення) включає здатність до модифікації веб-трафіку в режимі реального часу через використання техніки «Людина в браузері». Ці атаки дозволяють зловмисникам змінювати вміст веб-сторінок банків безпосередньо у браузері користувача, створюючи підроблені форми для введення додаткової інформації або модифікуючи деталі транзакцій без відома користувача.

Особливу увагу заслуговують можливості сучасного малвару щодо обходження двофакторної аутентифікації. Зловмисники використовують різноманітні техніки, включаючи перехоплення SMS-повідомлень, створення підроблених мобільних додатків та навіть фізичні пристрої для перехоплення комунікацій між мобільними операторами та банками. Деякі варіанти малвару здатні автоматично виконувати фінансові транзакції, використовуючи зібрані аутентифікаційні дані та обходячи системи моніторингу через імітацію нормальної поведінки користувачів.

Розповсюдження банківського малвару відбувається через різноманітні канали, включаючи зловмисні електронні листи, компрометовані веб-сайти, соціальні мережі та навіть легітимні програмні продукти, інфіковані через атаки на ланцюг постачання (supply chain attacks). Особливо небезпечними є атаки типу водопою (watering hole), коли зловмисники компрометують популярні веб-сайти, які регулярно відвідують представники цільових груп.[12]

Фармінг представляє собою найбільш технологічно складну форму шахрайських атак, що поєднує елементи соціальної інженерії з глибокими знаннями мережевих технологій та інфраструктури інтернету. На відміну від фішингу, де користувачів намагаються переконати перейти за шкідливими посиланнями, фармінг працює на більш фундаментальному рівні, перенаправляючи користувачів на шкідливі сайти навіть при правильному введенні адреси банку.

Технічна реалізація фармінгових атак може включати кілька різних підходів. DNS-спуофінг передбачає компрометацію DNS-серверів або локальних DNS-кешів для перенаправлення запитів на контрольовані зловмисниками сервери. Ця техніка особливо ефективна, оскільки користувачі не помічають жодних підозрілих ознак у своїй поведінці - вони вводять правильну адресу банку, але потрапляють на підроблений сайт.

Інший підхід до фармінгу передбачає модифікацію файлу hosts на комп'ютерах користувачів через шкідливе програмне забезпечення. Цей метод

дозволяє зловмисникам контролювати DNS-резолюцію на рівні окремих пристроїв, що робить атаку менш помітною для мережевих систем безпеки.

Особливо складними є комбіновані фармінгові атаки, що використовують компрометовані маршрутизатори або точки доступу Wi-Fi. Зловмисники можуть налаштувати публічні Wi-Fi точки з підробленими DNS-серверами або компрометувати домашні маршрутизатори користувачів для перенаправлення всього банківського трафіку на контрольовані ними сервери.

Шахрайські операції також класифікують за технологічними характеристиками на локальні, віддалені та гібридні.

Локальні атаки характеризуються безпосереднім впливом на пристрої кінцевих користувачів та експлуатацією вразливостей в операційних системах, веб-браузерах або встановленому програмному забезпеченні. Ця категорія атак має кілька ключових особливостей, що роблять її особливо небезпечною з точки зору банківської безпеки.

Першою особливістю локальних атак є їхня здатність функціонувати незалежно від мережевого з'єднання після початкового інфікування. Шкідливе програмне забезпечення може зберігати зібрану інформацію локально та передавати її зловмисникам при наступному підключенні до інтернету, що ускладнює виявлення таких атак системами мережевого моніторингу.

Друга характеристика пов'язана з можливістю глибокої інтеграції шкідливого коду в операційну систему. Сучасні локальні атаки можуть використовувати техніки руткіт (rootkit) для приховування своєї присутності від антивірусних програм та систем моніторингу. Це дозволяє зловмисникам тривалий час збирати банківську інформацію, не викликаючи підозр користувачів.

Третій аспект локальних атак стосується їхньої здатності до адаптації під специфічні банківські системи. Шкідливе програмне забезпечення може аналізувати встановлені додатки, закладки браузера та історію відвідувань для ідентифікації банків, якими користується жертва, та налаштовуватися відповідно до специфіки цих установ.

Особливу категорію локальних атак становлять атаки на мобільні пристрої. Зловмисні мобільні додатки можуть отримувати широкі дозволи на доступ до SMS-повідомлень, контактів, та інших критично важливих функцій пристрою. Це дозволяє зловмисникам не лише збирати банківську інформацію, але й перехоплювати коди двофакторної аутентифікації, що значно підвищує ефективність атак.

Віддалені атаки базуються на експлуатації мережевих протоколів, веб-технологій та централізованої інфраструктури для проведення шахрайських операцій. Ця категорія атак характеризується можливістю одночасного впливу на велику кількість потенційних жертв та високим рівнем автоматизації злочинних процесів.

Основною перевагою віддалених атак є їхня масштабованість. Зловмисники можуть створювати централізовані системи для управління тисячами підроблених веб-сайтів, автоматизованими системами розсилки фішингових електронних листів та обробки зібраних даних. Така централізація дозволяє ефективно координувати масштабні злочинні кампанії та швидко адаптуватися до нових захисних заходів банків.

Технологічна складність віддалених атак включає використання розподілених мереж серверів у різних юрисдикціях, що ускладнює правоохоронну діяльність та блокування шкідливих ресурсів. Зловмисники активно використовують технології CDN (Content Delivery Network) та проксі-сервери для маскування справжнього розташування своїх серверів та забезпечення високої доступності шкідливих ресурсів.

Особливою формою віддалених атак є використання ботнетів - мереж заражених комп'ютерів, контрольованих зловмисниками. Ботнети можуть використовуватися для розсилки фішингових електронних листів, DDoS-атак на легітимні банківські сервіси та навіть для проведення автоматизованих фінансових транзакцій з використанням зібраних облікових даних.

Віддалені атаки також характеризуються високим рівнем адаптивності. Зловмисники можуть у реальному часі аналізувати ефективність різних тактик,

A/B тестувати різні варіанти фішингових сторінок та швидко реагувати на зміни в системах безпеки банків.

Гібридні атаки представляють найбільш складну та загрозливу категорію шахрайських операцій, оскільки поєднують переваги як локальних, так і віддалених атак, створюючи синергетичний ефект, що значно підвищує їхню ефективність. Ці атаки характеризуються багатоетапністю, адаптивністю та здатністю до тривалої прихованої діяльності.

Типовий сценарій гібридної атаки може починатися з фішингового електронного листа або компрометованого веб-сайту, що служить вектором для інфікування пристрою користувача. Після успішного інфікування шкідливе програмне забезпечення встановлює постійний канал зв'язку з командними серверами зловмисників, що дозволяє отримувати оновлення, завантажувати додаткові модулі та координувати дії з іншими інфікованими пристроями.

Складність гібридних атак полягає в їхній здатності до адаптації під специфічні умови кожної жертви. Шкідливе програмне забезпечення може аналізувати банківські звички користувачів, ідентифікувати найбільш підходящі моменти для атак та навіть імітувати нормальну поведінку для уникнення виявлення системами безпеки.

Особливою характеристикою гібридних атак є їхня здатність до горизонтального поширення в корпоративних мережах. Після компрометації одного пристрою в організації, зловмисники можуть використовувати його як плацдарм для атак на інші системи, поступово розширюючи свій контроль та збираючи критично важливу інформацію про банківські операції організації.

Аналіз глобальних статистичних даних щодо шахрайських операцій демонструє тривожну тенденцію до зростання як кількості, так і складності таких атак. За даними Anti-Phishing Working Group, 2023 рік став рекордним за кількістю фішингових атак - було зафіксовано майже 5 мільйонів таких інцидентів, що представляє значне зростання порівняно з попередніми роками. Лише у четвертому кварталі 2023 року було зареєстровано 1,077,501 фішингову атаку, що свідчить про стабільно високий рівень активності зловмисників.[13]

Особливо тривожною є тенденція до зростання складності атак. За даними дослідження CrowdStrike, інтерактивні атаки, де за клавіатурою знаходиться людина, зросли на 60% у 2023 році. Показово, що 75% атак, які отримали доступ до систем, не використовували шкідливе програмне забезпечення, що свідчить про зміщення фокусу зловмисників у бік більш витончених методів, таких як фішинг облікових даних, атаки на паролі та соціальна інженерія. [14]

Фінансові втрати від кіберзлочинності також демонструють стрімке зростання. У 2024 році жертви повідомили про збитки від кіберзлочинності на суму $16.6 мільярдів доларів, що становить зростання на 33% порівняно з 2023 роком. Фішинг продовжує домінувати як найбільш поширена тактика шахрайства, що підкреслює його ефективність та доступність для зловмисників.[15]

Важливим трендом є зростання цільових атак на соціальні медіа платформи. За даними APWG, атаки на соціальні мережі зросли на 126% у четвертому кварталі 2023 року порівняно з третім кварталом, що відображає розширення векторів атак та використання зловмисниками нових каналів для досягнення потенційних жертв.

Ситуація з шахрайськими операціями в Україні характеризується особливими викликами, пов'язаними як з воєнним станом, так і з активною цифровізацією банківського сектору. За офіційними даними Національного банку України, загальна сума збитків від шахрайства з банківськими картами у 2023 році склала 833 мільйони гривень, що відображає масштаб проблеми в національному контексті.[3]

Значний вплив на ситуацію з банківською безпекою має воєнний стан та пов'язані з ним зміни у поведінці населення. Зловмисники активно експлуатують гуманітарну кризу, створюючи фішингові кампанії під виглядом благодійних організацій, військових структур або соціальних служб. НБУ виявив більше 4.5 тисю фішингових ресурсів у 2022 році [16], а з початку 2023 року було зафіксовано більше 8 тис. шахрайських доменів, що свідчить про інтенсифікацію діяльності зловмисників. [17]

Українська специфіка також включає високий рівень готовності населення до використання цифрових банківських послуг, що, з одного боку, сприяє розвитку фінансових технологій, але з іншого - створює більше можливостей для зловмисників. Широке використання мобільних банківських додатків та безконтактних платежів вимагає адаптації як захисних механізмів, так і освітніх програм для населення.

Аналіз розподілу шахрайських атак за секторами демонструє, що фінансовий сектор залишається основною ціллю зловмисників. За даними APWG, атаки проти онлайн-платіжних систем та банківського сектору разом становили 30.9% всіх атак у першому кварталі 2024 року, що підкреслює стабільну привабливість фінансових установ для кіберзлочинців.[9]

Особливої уваги заслуговує зростання атак типу Business Email Compromise (BEC), які спрямовані на корпоративних клієнтів банків. Кількість спостережуваних BEC-атак з банківськими переказами зросла на 33% у першому кварталі 2024 року порівняно з попереднім кварталом, що свідчить про зміщення фокусу зловмисників у бік більш прибуткових корпоративних цілей.

Використання машинного навчання для виявлення шахрайських операцій представляє собою один з найбільш перспективних напрямків розвитку банківської безпеки. Сучасні системи виявлення шахрайства базуються на аналізі великих масивів транзакційних даних та виявленні аномальних патернів поведінки, що можуть свідчити про шахрайську активність.

Основою для ефективних систем машинного навчання у банківській сфері є комплексний підхід до аналізу даних, що включає як структуровані транзакційні дані, так і неструктуровану інформацію про поведінку користувачів. Алгоритми Random Forest, Gradient Boosting та нейронні мережі використовуються для створення ансамблевих моделей, які демонструють високу точність виявлення шахрайських операцій при мінімізації неправильних-позитивних результатів.

Особливого значення набувають методи глибокого навчання для аналізу послідовностей транзакцій. Рекурентні нейронні мережі (RNN) та довготривала короткочасна пам'ять (LSTM) дозволяють аналізувати часові ряди транзакцій та

виявляти складні патерни шахрайської поведінки, які важко ідентифікувати традиційними статистичними методами.

1.2. Аналіз сучасних методів виявлення шахрайських операцій

Еволюція методів виявлення шахрайства відображає загальний розвиток технологій та підходів до аналізу даних. Якщо ще декілька десятиліть тому банки покладалися переважно на базові правила та людський досвід, то сьогодні більшість фінансових організацій впровадили технології машинного навчання або штучного інтелекту для боротьби з шахрайством. Ця трансформація обумовлена не лише зростанням кількості транзакцій, але й значним ускладненням шахрайських схем, які все частіше використовують передові технології для обходу традиційних систем захисту.

Розуміння різноманітних методів виявлення шахрайства та їх практичного застосування є критично важливим для банківських аналітиків, ризик-менеджерів та IT-спеціалістів. Кожен метод має свої унікальні переваги та обмеження, а їх ефективність значною мірою залежить від специфіки конкретного банку, типів транзакцій та характеристик клієнтської бази.

1.2.1. Традиційні методи виявлення шахрайства

До традиційних методи виявлення шахрайства відносять системи на основі правил (rule-based systems), що довгий час залишалися основою банківських систем виявлення шахрайства. Ці системи функціонують за принципом встановлення конкретних критеріїв та порогових значень, при перевищенні яких транзакція автоматично маркується як підозріла або блокується.

Фундаментальна логіка rule-based систем базується на експертних знаннях та історичному досвіді банківських спеціалістів. Типовими правилами можуть бути максимальні суми транзакцій для різних типів операцій, географічні обмеження, часові рамки для проведення операцій, або комбінації цих факторів. Наприклад, система може автоматично блокувати операцію, якщо клієнт намагається зняти

понад 5000 доларів США за межами своєї країни протягом 24 годин після останнього використання картки в домашньому регіоні.

Основною перевагою таких систем є їх прозорість та зрозумілість. Банківські працівники можуть легко пояснити клієнтам, чому конкретна транзакція була заблокована, а регулятори мають повне розуміння логіки прийняття рішень. Крім того, такі системи відносно прості в імплементації та не вимагають складних математичних обчислень або великих обчислювальних ресурсів.

Однак з часом стали очевидними серйозні недоліки цього підходу. Системи на основі правил характеризуються високим рівнем хибно-позитивних результатів, оскільки вони не здатні враховувати індивідуальні особливості поведінки клієнтів. Наприклад, людина, яка регулярно подорожує по всьому світу, може постійно стикатися з блокуванням своїх карток через географічні правила, що призводить до погіршення клієнтського досвіду.

Найбільш критичним обмеженням цього підходу є статичність. Шахраї швидко адаптуються до встановлених правил та знаходять способи їх подолання. Як тільки злочинці розуміють логіку системи, вони можуть структурувати свої операції таким чином, щоб залишатися нижче встановлених порогів або використовувати лазівки в правилах. Це створює постійну гонку між розробниками систем безпеки та шахраями, де перші завжди на крок позаду.

Також варто зазначити складність управління великою кількістю правил у великих банках. З часом кількість правил може досягати сотень або тисяч, що робить систему важко керованою та призводить до конфліктів між різними правилами. Оптимізація такої системи вимагає постійного моніторингу та ручного налаштування, що є трудомістким та дорогим процесом.

Поряд з системами на основі правил, традиційні підходи до виявлення шахрайства включають використання базових статистичних методів. Ці підходи ґрунтуються на принципі, що шахрайські транзакції є статистичними викидами (outliers) порівняно з нормальною поведінкою клієнтів.

Одним з найпоширеніших статистичних методів є аналіз Z-score, який дозволяє ідентифікувати транзакції, що значно відхиляються від середніх

показників клієнта. Для кожного клієнта розраховуються середні значення та стандартні відхилення для різних параметрів транзакцій: суми, частоти, часу проведення, географічного розташування. Транзакції, які виходять за межі встановлених стандартних відхилень (зазвичай 2-3 сигми), маркуються як підозрілі.

Метод квартилів та міжквартильного розмаху (IQR) також широко використовується для виявлення аномальних транзакцій. Цей підхід менш чутливий до екстремальних значень порівняно з методами, що базуються на середніх значеннях, що робить його більш стійким для аналізу фінансових даних, які часто містять значні викиди.

Кореляційний аналіз дозволяє виявляти незвичайні комбінації параметрів транзакцій. Наприклад, якщо зазвичай великі суми транзакцій корелюють з певними типами транзакцій або географічними регіонами, то транзакція з великою сумою в нетиповому місці може бути маркована як підозріла.

Часові ряди та їх аналіз також відіграють важливу роль в традиційних статистичних методах. Аналіз сезонних коливань, трендів та циклічних патернів у поведінці клієнтів дозволяє виявляти аномальні відхилення від очікуваних патернів. Особливо ефективним є використання ковзних середніх та експоненційного згладжування для виявлення різких змін у поведінці клієнтів.

Переваги статистичних методів включають їх математичну обґрунтованість та можливість кількісної оцінки ймовірності шахрайства. На відміну від простих систем на основі правил, статистичні підходи можуть адаптуватися до індивідуальних особливостей поведінки клієнтів та забезпечувати більш персоналізований підхід до виявлення аномалій. Однак статистичні методи також мають суттєві обмеження. Вони припускають, що дані мають певний розподіл (часто нормальний), що не завжди відповідає реальності фінансових транзакцій. Крім того, ці методи можуть бути неефективними для виявлення складних шахрайських схем, які використовують множинні взаємопов'язані фактори.

1.2.2. Машинне навчання та штучний інтелект у виявленні шахрайства

Революція в області машинного навчання та штучного інтелекту кардинально змінила підходи до виявлення шахрайства в банківській сфері. На відміну від вищевказаних підходів, алгоритми машинного навчання здатні автоматично виявляти складні патерни в даних, адаптуватися до нових типів шахрайства та покращувати свою точність з часом. Основна перевага машинного навчання полягає в здатності обробляти величезні обсяги даних та виявляти неочевидні зв'язки між різними параметрами транзакцій. Сучасні банки генерують мільйони транзакцій щодня, і традиційні методи просто не здатні ефективно аналізувати такі обсяги інформації в режимі реального часу.

Машинне навчання дозволяє створювати більш персоналізовані моделі поведінки для кожного клієнта. Замість використання загальних правил для всіх клієнтів, алгоритми можуть враховувати індивідуальні особливості: звички витрат, географічні переміщення, часові патерни активності, типи транзакцій тощо. Це значно знижує кількість хибно-позитивних спрацьовувань та покращує клієнтський досвід. Адаптивність є ще однією ключовою перевагою машинного навчання. Шахрайські схеми постійно еволюціонують, і статичні правила швидко втрачають свою ефективність. Алгоритми машинного навчання можуть автоматично перенавчатися на нових даних, виявляти нові типи шахрайства та адаптувати свої моделі до змінюваних умов.

Реалізація машинного навчання в банківських системах зазвичай включає декілька етапів. Спочатку проводиться збір та підготовка даних, що включає очищення, нормалізацію та створення релевантних ознак. Потім здійснюється вибір та навчання відповідних алгоритмів на історичних даних з відомими випадками шахрайства. Нарешті, навчені моделі інтегруються в реальні банківські системи для обробки транзакцій в режимі реального часу.

Однак впровадження машинного навчання також створює нові виклики. Ці системи часто функціонують як "чорні скриньки", що ускладнює пояснення рішень регуляторам та клієнтам. Крім того, алгоритми можуть бути схильними

до упереджень, особливо якщо навчальні дані не є репрезентативними або містять історичні упередження.

Серед методів, які застосовуються у машинному навчанні варто виділити алгоритми класифікації, які формують основу більшості сучасних систем виявлення шахрайства. Ці методи дозволяють автоматично категоризувати транзакції як легітимні або шахрайські на основі аналізу множини характеристик.

Дерева рішень є одними з найпопулярніших алгоритмів класифікації в банківській сфері завдяки своїй інтерпретованості та ефективності. Дерево рішень створює ієрархічну структуру правил, яка послідовно розділяє дані на основі найбільш інформативних ознак. Наприклад, перший рівень дерева може розділяти транзакції за сумою, другий - за географічним розташуванням, третій - за часом проведення тощо. Переваги дерев рішень включають легкість інтерпретації результатів, здатність працювати як з числовими, так і з категоріальними даними, та стійкість до викидів. Банківські аналітики можуть легко зрозуміти логіку прийняття рішень та пояснити її клієнтам або регуляторам. Крім того, дерева рішень автоматично виконують відбір найбільш важливих ознак, що спрощує процес розробки моделі.

Однак дерева рішень схильні до перенавчання, особливо на складних наборах даних. Вони можуть створювати занадто специфічні правила, які добре працюють на навчальних даних, але погано узагальнюють на нові випадки. Для вирішення цієї проблеми часто використовуються ансамблеві методи, такі як Random Forest або градієнтний бустинг (Gradient Boosting).

Random Forest поєднує множину дерев рішень, кожне з яких навчається на випадковій підвибірці даних та ознак. Остаточне рішення приймається шляхом голосування всіх дерев, що значно підвищує стабільність та точність моделі. Цей метод особливо ефективний для роботи з великими наборами даних та зашумленими даними, що характерно для банківських транзакцій.

Градієнтний бустинг також поєднує множину дерев, шляхом послідовного навчання. Кожна наступна модель (дерево рішень) фокусується на виправленні

помилок, допущених попередніми, що дозволяє поступово зменшувати загальну похибку системи.

Support Vector Machines (SVM) представляють інший підхід до класифікації, що базується на пошуку оптимальної межі розділення між класами. SVM ефективні для динамічних патернів шахрайства, але вимагають якісних навчальних даних. SVM особливо корисні в ситуаціях з високою розмірністю ознак та відносно невеликими datasets.

Основна ідея SVM полягає в знаходженні гіперплощини, яка максимально розділяє два класи (легітимні та шахрайські транзакції) з максимальним відступом. Це досягається за рахунок використання так званих опорних векторів - точок даних, які знаходяться найближче до межі розділення. SVM можуть ефективно працювати з нелінійними залежностями за рахунок використання kernel-функцій, які дозволяють перетворювати дані в простір вищої розмірності, де класи стають лінійно розділеними. Популярні kernel-функції включають поліноміальні, радіальні базисні функції (RBF) та сигмоїдні функції.

Моделі класифікації в контексті виявлення шахрайства представляють формалізований математичний підхід до категоризації транзакцій. Ці моделі базуються на теорії ймовірностей та статистичному результаті, дозволяючи кількісно оцінювати ризик шахрайства для кожної транзакції.

Базова математична формулізація задачі класифікації полягає у знаходженні функції (1)

$$Y = F(X) \qquad (1)$$

яка відображає простір ознак X у простір класів Y = {0, 1}, де 0 відповідає легітимній транзакції, а 1 - шахрайській.

Дерева рішень використовують рекурсивне розбиття простору ознак на основі критеріїв інформативності, таких як ентропія або коефіцієнт Джіні. Це метрика нечистоти вузла, яка показує, наскільки змішаний набір транзакцій у вузлі - тобто, наскільки вузол містить різні класи. Якщо всі транзакції належать до одного класу, вузол "чистий", і Джіні = 0. Критерій Джіні для вузла визначається як:

$$Gini = 1 - \sum_i p_i^2 \quad (2)$$

де p_i - частка транзакцій класу i у вузлі.

Ансамблеві методи комбінують прогнози декількох базових класифікаторів. Random Forest використовує bagging та випадковий відбір ознак:

$$Y = majority\_vote(T_1(X), T_2(X), \ldots, T_n(X)) \quad (3)$$

де $T_i(X)$ це прогноз, який дає i-те дерево рішення для прикладу X, $T_1(X)$, $T_2(X)$, ..., $T_n(X)$ - прогнози всіх дерев у ансамблі (усього n дерев), majority_vote(...) - механізм голосування більшості (вибір класу, за який проголосувала більшість дерев).

Gradient Boosting послідовно будує моделі, кожна з яких корегує помилки попередніх:

$$F\_m(x) = F\_\{m-1\}(x) + \gamma\_m \, h\_m(x) \quad (4)$$

де $F_m(x)$ - поточна модель на кроці m, яка об'єднує всі попередні передбачення, $F_{m-1}(x)$ - модель на попередньому кроці (m − 1) - тобто той прогноз, який вже маємо, $h_m(x)$ - «Слабкий учень» (новий простий класифікатор або регресор, який навчається на залишках).

1.2.3. Кластеризація та виявлення аномалій

Кластеризація в контексті виявлення шахрайства дозволяє групувати схожі транзакції та виявляти аномальні патерни без попереднього маркування даних. Ці методи особливо корисні для виявлення нових типів шахрайства, які ще не були задокументовані.

K-means є найпоширенішим алгоритмом кластеризації. Він мінімізує суму квадратів відстаней від точок до центрів кластерів:

$$J = \sum_{i=1}^{n} \sum_{j=1}^{k} w_{ij} \, ||x_i - \mu_j||^2 \quad (5)$$

де $w_{ij} = 1$, якщо точка x_i належить кластеру j, та $w_{ij} = 0$ у протилежному випадку; μ_j - центр j-го кластеру.

Алгоритм працює ітеративно: спочатку випадково ініціалізуються центри кластерів, потім точки призначаються найближчим центрам, після чого центри перераховуються як середні арифметичні точок у кластерах. Процес

повторюється до збіжності. У контексті виявлення шахрайства K-means може виявляти групи схожих транзакцій. Аномалії можуть бути ідентифіковані як точки, які знаходяться далеко від усіх центрів кластерів, або як невеликі кластери з незвичайними характеристиками.

Ієрархічний кластеринг будує ієрархію кластерів, що може бути представлена у вигляді дендрограми. Аґломеративний кластеринг починає з кожної точки як окремого кластеру і послідовно об'єднує найближчі кластери. Різні метрики відстані (single linkage, complete linkage, average linkage) визначають критерії об'єднання.

Для проведення кластеризації визначається міра близькості (схожості), яку використовують при багатомірному аналізі даних. Метрика визначається переважно вимірюванням відстані між об’єктами, тобто кожен об’єкт описується набором характеристик, представлених у вигляді багатомірного простору. Розрахувавши метричну відстань між ними, можна зробити висновок про їх відстань. До найпоширеніших методів її визначення належать:

1. Евклідова відстань. Цей метод дає можливість не враховувати знакові відмінності, пропорційно збільшуючи відстань у випадку різних абсолютних значень метрик. Розраховується за формулою:

$$d_{ijE} = d_E(x_i x_j) = \sqrt{(x_i - x_j)(x_i - x_j)^T} = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2} \qquad (6)$$

де n - число ознак об’єкта,

x_{ik} , x_{jk} - значення k-ї змінної i-го та j-го об’єктів,

d_{ijE} - відстань між цими об’єктами.

2. Квадрат евклідової відстані. Застосовуються, щоб надати більшої ваги віддаленішим спостереженням. Розраховується за формулою:

$$d_{ijE} = d_{E^2}(x_i x_j) = \sum_{k=1}^{n} (x_{ik} - x_{jk})^2 \qquad (7)$$

3. Зважена евклідова відстань. Розраховується за формулою:

$$d_{ij} = d(x_i x_j) = \sqrt{\sum_{k=1}^{n} \gamma_k (x_{ik} - x_{jk})^2} \qquad (8)$$

де γ_k - вага k-ї ознаки, пропорційна її важливості, γ_k є [0; 1]

4. Лінійна відстань Хеммінга. Використовується за умови, що усі ознаки є дихотомічними. Вплив окремих великих відстаней зменшено. Як цей метод, так і Манхеттенська відстань (Відстань міських кварталів) розраховується за формулою:

$$d_{ijE} = d_{E^2}(x_i x_j) = \sum_{k=1}^{n} \left| x_{ik} - x_{jk} \right| \quad (9)$$

1.2.4. Логістична регресія

Логістична регресія залишається одним з найпоширеніших методів класифікації завдяки своїй простоті та можливості отримання ймовірнісних оцінок. На відміну від дерев рішень, які дають чітку класифікацію, логістична регресія надає ймовірність того, що транзакція є шахрайською. Це дозволяє банкам встановлювати різні пороги для різних типів операцій та клієнтів.

Логістична регресія є однією з найфундаментальніших моделей класифікації. Вона моделює ймовірність того, що транзакція є шахрайською, використовуючи логістичну функцію (2):

$$P(Y = 1|X) = 1 / (1 + \exp(-\beta_0 - \beta_1 x_1 - \beta_2 x_2 - \ldots - \beta_p x_p)) \quad (10)$$

де β_0, β_1, ..., β_p - параметри моделі, які оцінюються методом максимальної правдоподібності. Логістична регресія забезпечує інтерпретовані коефіцієнти, що показують, як кожна ознака впливає на ймовірність шахрайства.

1.2.5. Нейронні мережі та глибоке навчання

Нейронні мережі та технології глибокого навчання представляють найсучасніший підхід до виявлення шахрайства в банківській сфері. Ці методи здатні виявляти складні нелінійні залежності в даних, які недоступні традиційним алгоритмам машинного навчання.

Штучні нейронні мережі (ANN) імітують структуру та функціонування біологічних нейронних мереж. Вони складаються з множини взаємопов'язаних вузлів (нейронів), кожен з яких обробляє вхідну інформацію та передає результат далі по мережі. Штучні нейронні мережі використовуються для прогнозування виникнення шахрайства разом з іншими алгоритмами машинного навчання.

Основна сила нейронних мереж полягає в їх здатності автоматично виявляти та вивчати складні патерни в даних без явного програмування правил. У контексті виявлення шахрайства це означає, що мережа може самостійно знаходити комбінації параметрів транзакцій, які вказують на підозрілу активність, навіть якщо ці комбінації не очевидні для людського аналізу.

Багатошарові перцептрони (Multi-Layer Perceptrons, MLP) є найпростішим типом нейронних мереж, що використовуються для виявлення шахрайства. Вони складаються з вхідного шару, одного або декількох прихованих шарів та вихідного шару. Кожен нейрон у прихованому шарі обчислює зважену суму своїх входів, застосовує активаційну функцію і передає результат далі.

Процес навчання нейронної мережі здійснюється за допомогою алгоритму зворотного поширення помилки (backpropagation). Мережа обробляє навчальні приклади, порівнює свої прогнози з правильними відповідями та корегує ваги зв'язків для мінімізації помилки. Цей процес повторюється ітеративно до досягнення задовільної точності.

Глибокі нейронні мережі (Deep Neural Networks, DNN) містять значно більше прихованих шарів, що дозволяє їм вивчати більш складні та абстрактні представлення даних. У контексті банківських даних це означає можливість виявлення шахрайських патернів, які проявляються лише при аналізі множинних рівнів взаємодії між різними параметрами транзакцій.

Конволюційні нейронні мережі (CNN) знаходять застосування у виявленні шахрайства, особливо при аналізі зображень документів, підписів або інших візуальних елементів. CNN обробляють зображення (наприклад, скановані чеки, ID картки) для виявлення аномалій. Ці мережі використовують спеціальні шари згортки та об'єднання для ефективного аналізу просторових патернів у зображеннях.

Рекурентні нейронні мережі (RNN) та їх удосконалені варіанти, такі як LSTM (Long Short-Term Memory) та GRU (Gated Recurrent Unit), особливо корисні для аналізу послідовностей транзакцій. Вони здатні "запам'ятовувати" інформацію з

попередніх транзакцій клієнта та використовувати її для оцінки поточної операції.

LSTM мережі вирішують проблему зникаючого градієнта, яка притаманна звичайним RNN, дозволяючи ефективно аналізувати довгі послідовності транзакцій. Це особливо важливо для виявлення шахрайських схем, які розгортаються протягом тривалого періоду часу.

Автокодери (Autoencoders) представляють особливий тип нейронних мереж, призначений для виявлення аномалій. Автокодери використовуються для виявлення шахрайства з кредитними картками з балансом точності та повноти. Вони навчаються стискати вхідні дані до компактного представлення, а потім відновлювати оригінальні дані з цього представлення.

Ідея полягає в тому, що автокодер, навчений на легітимних транзакціях, буде добре відновлювати нормальні операції, але матиме високу помилку відновлення для аномальних (потенційно шахрайських) транзакцій. Variational Autoencoders (VAE) та Adversarial Autoencoders представляють більш складні варіанти цього підходу.

1.2.6. Графові моделі

Графові моделі представляють потужний інструмент для виявлення шахрайства, особливо коли мова йде про організовані злочинні групи та складні схеми відмивання грошей. Техніки машинного навчання, такі як графовий аналіз, можуть використовуватися для виявлення мереж потенційних шахраїв шляхом аналізу зв'язків між сутностями. Основна ідея графового підходу полягає в моделюванні фінансової екосистеми як мережі, де вузлами є клієнти, рахунки, торгові точки, банкомати тощо, а ребрами - транзакції або інші типи взаємодій між цими сутностями. Такий підхід дозволяє виявляти підозрілі патерни, які неможливо виявити при аналізі окремих транзакцій.

Графові нейронні мережі (Graph Neural Networks, GNN) поєднують переваги графового моделювання з потужністю глибокого навчання. Графові нейронні мережі використовуються для виявлення шахрайства з «lambda» архітектурою.

GNN здатні навчатися представленням вузлів графа, враховуючи як їх власні характеристики, так і структуру околиці.

Процес роботи GNN включає декілька етапів агрегації та оновлення інформації. На кожній ітерації вузли отримують інформацію від своїх сусідів, агрегують її з власною інформацією та оновлюють своє представлення. Після декількох ітерацій кожен вузол містить інформацію про свою розширену околицю, що дозволяє виявляти складні патерни взаємодій.

Graph Convolutional Networks (GCN) є одним з найпопулярніших типів GNN. Вони застосовують операцію згортки до графових структур, дозволяючи ефективно агрегувати інформацію від сусідніх вузлів. GraphSAGE (Graph Sample and Aggregate) використовує стратегії вибіркового агрегування для масштабування до великих графів.

Практичне застосування графових моделей у виявленні шахрайства включає виявлення кілець (ring detection), де група шахраїв циркулярно переміщує кошти між своїми рахунками для запутування сліду. Алгоритми пошуку циклів у графі можуть ефективно виявляти такі схеми.

Аналіз центральності дозволяє ідентифікувати ключових гравців у шахрайських мережах. Різні міри центральності (betweenness, closeness, eigenvector centrality) виявляють вузли, які відіграють важливу роль у структурі мережі та можуть бути центрами шахрайської активності.

Кластерний аналіз графів допомагає виявляти щільно пов'язані групи сутностей, які можуть представляти організовані злочинні групи. Алгоритми спільнот (community detection) можуть автоматично ідентифікувати такі групи на основі структури зв'язків. Часова еволюція графів також несе важливу інформацію. Dynamic Graph Analysis дозволяє відстежувати зміни в структурі мережі з часом та виявляти аномальні патерни активності, такі як раптове утворення нових зв'язків або зміна інтенсивності взаємодій

1.2.7. Виявлення аномалій

Виявлення аномалій (anomaly detection) є фундаментальним підходом до ідентифікації шахрайства, оскільки шахрайські транзакції за визначенням є відхиленнями від нормальної поведінки. Метою виявлення аномалій є виявлення шахрайської активності в системах електронного банкінгу з підтриманням кількості помилкових значень на прийнятному рівні.

Статистичні методи виявлення аномалій базуються на припущенні, що нормальні дані слідують певному статистичному розподілу, а аномалії є викидами з цього розподілу. «Z-score», модифікований «Z-score» та «Tukey's fences» є класичними статистичними методами, які можуть ефективно виявляти прості аномалії в одновимірних даних.

Багатовимірні статистичні методи, такі як відстань Махаланобіса, враховують кореляції між різними ознаками та можуть виявляти аномалії, які не очевидні при аналізі окремих параметрів. Цей метод особливо корисний для фінансових даних, де різні параметри транзакцій часто корелюють між собою.

Машинне навчання для виявлення використовує алгоритми для вивчення нормальної поведінки та виявлення аномалій в режимі реального часу, вони пропонують більш гнучкі та потужні підходи до виявлення аномалій, особливо в багатовимірних просторах ознак.

Однокласова SVM (One-Class SVM) є популярним методом для виявлення аномалій, який навчається визначати межі нормальної поведінки, використовуючи лише дані про легітимні транзакції. Це особливо корисно в ситуаціях, коли шахрайські приклади рідкісні або недоступні для навчання.

Ізольований ліс (Isolation Forest) - це алгоритм, що базується на принципі, що аномалії легше ізолювати (відокремити від решти даних), ніж нормальні точки. Алгоритм створює випадкові дерева розділення та вимірює кількість розділень, необхідних для ізоляції кожної точки. Аномалії зазвичай вимагають менше розділень.

Local Outlier Factor (LOF) визначає аномальність точки на основі локальної щільності її околиці. Точки, які знаходяться в регіонах з значно нижчою щільністю порівняно з їх сусідами, вважаються аномаліями. Цей метод особливо

ефективний для виявлення локальних аномалій у даних з неоднорідною щільністю.

Кластер-базовані методи виявлення аномалій припускають, що нормальні дані утворюють щільні кластери, а аномалії знаходяться далеко від цих кластерів або в малих кластерах. DBSCAN може ідентифікувати точки як шум, якщо вони не належать до жодного щільного кластера, що робить їх потенційними аномаліями.

Глибоке навчання також знайшло широке застосування у виявленні аномалій. Автокодери, які згадувалися раніше, є одним з найпопулярніших підходів. Вони навчаються стискати нормальні дані до низькорозмірного представлення і потім відновлювати оригінальні дані. Високі помилки відновлення вказують на потенційні аномалії.

Variational Autoencoders (VAE) розширюють концепцію автокодерів, додаючи ймовірнісний компонент. Вони можуть не лише виявляти аномалії, але й оцінювати ймовірність того, що конкретна транзакція є аномальною.

Generative Adversarial Networks (GAN) використовуються для генерації синтетичних нормальних транзакцій. Дискримінатор GAN може служити детектором аномалій, ідентифікуючи транзакції, які не схожі на згенеровані "нормальні" приклади.

Gaussian Mixture Models (GMM) припускають, що дані генеруються сумішшю гаусівських розподілів. Кожен кластер моделюється окремим гаусівським розподілом з власними параметрами (середнє μ та коваріаційна матриця Σ).

GMM навчаються за допомогою EM-алгоритму, який ітеративно оптимізує параметри моделі. У контексті виявлення шахрайства GMM можуть моделювати розподіл нормальних транзакцій, а точки з низькою ймовірністю під цією моделлю можуть вважатися аномаліями.

1.2.8. Байєсівські моделі

Байєсівські моделі забезпечують принципову основу для врахування невизначеності у виявленні шахрайства. Вони дозволяють поєднувати попередні знання експертів з даними для отримання більш надійних висновків.

Теорема Байєса є фундаментальною основою:

$$P(\theta|D) = P(D|\theta) \times P(\theta) / P(D) \quad (12)$$

де P(θ|D) - апостеріорна ймовірність параметрів θ за умови спостережених даних D; P(D|θ) - правдоподібність; P(θ) - априорна ймовірність; P(D) - маргінальна ймовірність даних.

Байєсівська логістична регресія розширює класичну логістичну регресію, розглядаючи параметри моделі як випадкові величини з певними розподілами. Замість точкових оцінок параметрів отримуються їх розподіли, що дозволяє кількісно оцінювати невизначеність прогнозів.

Наївний байєсівський класифікатор, незважаючи на припущення про незалежність ознак, часто показує хороші результати у виявленні шахрайства завдяки своїй стійкості до шуму та здатності працювати з невеликими навчальними вибірками.

Для категоріальних ознак використовується мультиноміальний розподіл:

$$P(x_i|Y = c_j) = (\prod_{k=1}^{m} \theta_{jk}\hat{}(x_{ik})) / (\prod_{k=1}^{m} x_{ik}!) \quad (13)$$

де θ_{jk} - параметр мультиноміального розподілу для класу j та ознаки k.

Для безперервних ознак зазвичай припускається гаусівський розподіл:

$$P(x_i|Y = c_j) = 1/\sqrt{(2\pi\sigma_j^2)}\, exp(-(x_i - \mu_j)^2/(2\sigma_j^2)) \quad (14)$$

Байєсівські мережі представляють складніші моделі, які можуть враховувати залежності між ознаками. Вони використовують направлені ациклічні графи для представлення умовних залежностей між змінними.

Спільний розподіл ймовірностей у байєсівській мережі факторизується як:

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|Parents(X_i)) \quad (15)$$

Навчання байєсівських мереж може здійснюватися як за допомогою методів максимальної правдоподібності, так і з використанням повністю байєсівських підходів з MCMC методами для апостеріорного виводу.

1.2.9. Регресійні моделі

Регресійні моделі в контексті виявлення шахрайства використовуються для прогнозування безперервних величин, таких як ймовірність шахрайства, очікувані збитки або рівень ризику. Ці моделі особливо корисні для ранжування транзакцій за рівнем підозрілості.

Лінійна регресія моделює залежність між ознаками X та цільовою змінною Y як лінійну функцію:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p + \varepsilon \quad (16)$$

де ε - випадкова похибка з нормальним розподілом.

Метод найменших квадратів мінімізує суму квадратів залишків:

$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \quad (17)$$

Поліноміальна регресія розширює лінійну модель, включаючи поліноміальні терми:

$$Y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \ldots + \beta_n x^n + \varepsilon \quad (18)$$

Це дозволяє моделювати нелінійні залежності між ознаками та цільовою змінною.

Generalized Additive Models (GAM) дозволяють моделювати нелінійні ефекти кожної ознаки окремо:

$$g(E[Y]) = \alpha + f_1(x_1) + f_2(x_2) + \ldots + f_p(x_p) \quad 9)$$

де g - функція зв'язку, f_i - гладкі функції ознак.

1.2.10. Інструменти для реалізації систем виявлення шахрайства

Сучасна екосистема інструментів для реалізації систем виявлення шахрайства включає широкий спектр програмного забезпечення, бібліотек та платформ. Вибір конкретних інструментів залежить від масштабу проєкту, технічних вимог, бюджету та експертизи команди розробників.

Python займає провідне місце серед мов програмування для розробки систем виявлення шахрайства завдяки своїй простоті, потужним бібліотекам та активній спільноті розробників. Екосистема Python включає декілька ключових бібліотек, які формують основу більшості проєктів.

Scikit-learn є найпопулярнішою бібліотекою загального призначення для машинного навчання. Вона надає уніфікований інтерфейс для широкого спектра алгоритмів: від простих лінійних моделей до складних ансамблевих методів. Для виявлення шахрайства особливо корисними є модулі для класифікації (LogisticRegression, RandomForestClassifier, SVC), кластеризації (KMeans, DBSCAN) та виявлення аномалій (IsolationForest, LocalOutlierFactor).

NumPy та Pandas формують фундамент для роботи з даними. NumPy забезпечує ефективні операції з багатовимірними масивами, а Pandas надає потужні інструменти для маніпуляції структурованими даними. Для банківських даних особливо корисними є функції групування, агрегації та часових рядів.

Лістинг 1.1. Приклад використання scikit-learn для базової класифікації:

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
# Підготовка даних
X_train, X_test, y_train, y_test = train_test_split(features, labels, test_size=0.2)
# Навчання моделі
rf_model = RandomForestClassifier(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)

# Прогнозування та оцінка
predictions = rf_model.predict(X_test)
print(classification_report(y_test, predictions))
```

TensorFlow та PyTorch є провідними фреймворками для глибокого навчання. TensorFlow, розроблений Google, надає комплексну платформу для розробки та деплою моделей машинного навчання. Keras, високорівневий API для TensorFlow, значно спрощує створення нейронних мереж.

PyTorch, розроблений Facebook, пропонує більш гнучкий підхід до створення нейронних мереж завдяки динамічним обчислювальним графам. Це робить його особливо привабливим для дослідницьких проєктів та експериментів з новими архітектурами. XGBoost та LightGBM є спеціалізованими бібліотеками для gradient boosting, які часто показують найкращі результати в конкурсах з

машинного навчання та практичних застосуваннях. Вони оптимізовані для швидкості та ефективності роботи з великими datasets.

Лістинг 1.2. Приклад створення простої нейронної мережі з Keras:

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout

model = Sequential([
    Dense(128, activation='relu', input_shape=(n_features,)),
    Dropout(0.3),
    Dense(64, activation='relu'),
    Dropout(0.3),
    Dense(1, activation='sigmoid')
])

model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=['accuracy'])
model.fit(X_train, y_train, epochs=100, batch_size=32, validation_split=0.2)
```

Мова програмування «R» також залишається важливим інструментом, особливо серед статистиків та аналітиків з традиційною статистичною підготовкою. «R» пропонує унікальні переваги для статистичного моделювання та візуалізації даних.

Лістинг 1.3. Приклад використання R для логістичної регресії:

```
library(caret)
library(pROC)

# Підготовка даних
trainIndex <- createDataPartition(data$fraud, p = 0.8, list = FALSE)
trainData <- data[trainIndex, ]
testData <- data[-trainIndex, ]

# Навчання моделі
model <- glm(fraud ~ ., data = trainData, family = binomial())

# Прогнозування
predictions <- predict(model, testData, type = "response")
predicted_classes <- ifelse(predictions > 0.5, 1, 0)

# Оцінка
confusionMatrix(factor(predicted_classes), factor(testData$fraud))
```

Ключовими пакетами «R» для виявлення шахрайства є caret (Classification And REgression Training), який надає уніфікований інтерфейс для сотень алгоритмів машинного навчання, randomForest для ансамблевих методів. Спеціалізовані бібліотеки та фреймворки також відіграють важливу роль. PyOD (Python Outlier Detection) надає комплексний набір алгоритмів для виявлення аномалій. NetworkX дозволяє працювати з графовими структурами, що критично важливо для аналізу мереж транзакцій.

Для роботи з великими даними Apache Spark забезпечує розподілені обчислення. MLlib, бібліотека машинного навчання для Spark, дозволяє масштабувати алгоритми на кластери серверів. PySpark надає Python API для роботи з Spark.

Хмарні платформи значно спрощують розгортання та масштабування систем виявлення шахрайства. Amazon Web Services (AWS) пропонує SageMaker для розробки та розгортання моделей машинного навчання, Amazon Fraud Detector - готове рішення для виявлення шахрайства. Google Cloud Platform надає AutoML та AI Platform, Microsoft Azure - Machine Learning Service.

1.3. Світовий досвід протидії банківському шахрайству в цифровому середовищі

Сучасна банківська індустрія переживає справжню революцію у сфері безпеки завдяки впровадженню передових технологій машинного навчання. Фінансові установи у всьому світі визнали, що традиційні методи виявлення шахрайства більше не можуть ефективно протистояти складним та постійно еволюціонуючим схемам фінансових злочинів. Аналіз досвіду провідних банків США, Великобританії та Сінгапуру демонструє, що використання алгоритмів машинного навчання для виявлення шахрайства в реальному часі стало не просто конкурентною перевагою, а необхідністю для виживання в сучасному цифровому фінансовому середовищі.

1.3.1. PayPal: найсучасніша система виявлення шахрайства

PayPal, один з чемпіонів у сфері цифрових платежів, розробив одну з найсучасніших систем виявлення шахрайства в індустрії. Система PayPal обробляє понад 20 мільярдів транзакцій на рік і демонструє, як можна ефективно поєднувати традиційні підходи з передовими технологіями машинного навчання.

Архітектура системи PayPal базується на багатошаровому підході. Перший шар включає базові правила та фільтри, які блокують очевидно шахрайські транзакції на основі чорних списків, географічних обмежень та інших простих критеріїв. Цей шар обробляється дуже швидко і дозволяє зменшити навантаження на більш складні компоненти системи.

Другий шар використовує машинне навчання для детального аналізу транзакцій. PayPal впровадила ансамблеві методи, що поєднують результати декількох різних алгоритмів: логістичну регресію, дерева рішень, нейронні мережі та граєнтний бустинг. Кожен алгоритм спеціалізується на виявленні певних типів шахрайства, а фінальне рішення приймається на основі зваженого голосування.

Особливо інноваційним є використання PayPal графового аналізу для виявлення складних шахрайських схем. Система створює граф взаємодій між користувачами, пристроями, IP-адресами та іншими сутностями. Алгоритми графового аналізу виявляють підозрілі кластери активності, кільця переказів коштів та інші індикатори організованого шахрайства.[18]

Результати впровадження вражають: PayPal зменшила рівень шахрайства з 0.32% в 2013 році до 0.18% в 2019 році, одночасно скоротивши кількість хибно-позитивних спрацьовувань на 60%. Це призвело до покращення клієнтського досвіду та економії мільйонів доларів щорічно.

1.3.2. JPMorgan Chase: застосуванні штучного інтелекту

JPMorgan Chase, найбільший банк США за активами, демонструє найбільш амбіційний підхід до впровадження технологій машинного навчання у боротьбі з фінансовими злочинами. Банк інвестував понад 15 мільярдів доларів у

технології протягом останніх років, значна частина яких була спрямована на розробку систем виявлення шахрайства на основі штучного інтелекту.

Одним з найбільш інноваційних напрямків роботи банку є використання передових технологій штучного інтелекту для виявлення компрометації ділової електронної пошти. Ця ініціатива особливо важлива, оскільки атаки через ділову електронну пошту стали однією з найбільш поширених форм фінансового шахрайства у корпоративному секторі. JPMorgan Chase розробив складну систему аналізу текстових повідомлень, яка використовує природну обробку мови та глибоке навчання для ідентифікації підозрілих комунікацій. [19]

Включення рішень на основі штучного інтелекту у процеси комплаєнсу для боротьби з відмиванням грошей допомогло виявляти шахрайські документи шляхом аналізу тенденцій та нерегулярностей, що посилило здатність виявляти шахрайські дії. Компанія досягла підвищеної точності та досягла вражаючого 95% зниження хибних спрацьовувань. Цей результат є особливо значущим, оскільки хибні спрацьовування завжди були однією з найбільших проблем традиційних систем виявлення шахрайства, призводячи до блокування законних транзакцій та погіршення клієнтського досвіду. [20]

Технологічна платформа JPMorgan Chase включає використання синтетичних даних для тренування моделей виявлення шахрайства. Цей підхід дозволяє банку створювати реалістичні набори даних без розкриття конфіденційної інформації клієнтів, що особливо важливо в умовах посилених вимог до захисту персональних даних. Синтетичні дані генеруються з використанням генеративно-змагальних мереж (GAN), які здатні відтворювати статистичні характеристики реальних транзакційних даних, зберігаючи при цьому конфіденційність. [21]

Банк також активно використовує технології обробки природної мови для аналізу неструктурованих даних, включаючи електронні листи, повідомлення у месенджерах та документи. Ця система дозволяє виявляти підозрілі комунікації, які можуть свідчити про планування або здійснення шахрайських операцій.

Алгоритми аналізують не лише зміст повідомлень, але й метадані, включаючи час відправлення, частоту комунікацій та поведінкові патерни користувачів.

1.3.3. Wells Fargo: фокус на поведінковій аналітиці

Wells Fargo, четвертий за величиною банк США, розробив унікальну систему поведінкової аналітики, яка аналізує патерни поведінки клієнтів для виявлення аномальних дій. Банк використовує передові алгоритми машинного навчання для створення індивідуальних профілів поведінки кожного клієнта, що дозволяє виявляти відхилення від звичайних патернів використання банківських послуг.

Система Wells Fargo аналізує понад 200 різних параметрів для кожної транзакції, включаючи час проведення операції, географічне розташування, тип пристрою, з якого здійснюється доступ, та поведінкові характеристики користувача під час введення даних. Алгоритми машинного навчання постійно навчаються на нових даних, адаптуючись до змін у поведінці клієнтів та нових типів шахрайських схем. [22]

Особливу увагу Wells Fargo приділяє виявленню шахрайства у сфері мобільного банкінгу. Банк розробив спеціалізовані алгоритми, які аналізують особливості використання мобільних додатків, включаючи швидкість набору тексту, тиск на екран, траєкторію рухів пальців та інші біометричні характеристики. Ці дані використовуються для створення унікального цифрового відбитка кожного користувача, що дозволяє виявляти несанкціонований доступ до облікових записів.

Банк також впровадив систему аналізу соціальних мереж, яка допомагає виявляти потенційні загрози безпеки шляхом моніторингу публічної інформації про клієнтів. Ця система використовує алгоритми обробки природної мови для аналізу постів у соціальних мережах, виявляючи ознаки фінансових проблем, які можуть призвести до шахрайських дій.

1.3.4. HSBC: глобальна мережа захисту

HSBC, один з найбільших банків Великобританії з глобальною присутністю, створив унікальну систему виявлення шахрайства, яка об'єднує дані з різних

юрисдикцій для забезпечення комплексного захисту [23, 24]. Використовуючи штучний інтелект, HSBC перевіряє понад мільярд транзакцій на ознаки фінансових злочинів щомісяця, зменшуючи хибні спрацьовування та ручні перевірки.

Система HSBC використовує передові алгоритми кластеризації для групування подібних транзакцій та ідентифікації аномальних патернів [25, 26]. Банк розробив власні алгоритми на основі ансамблевих методів, які поєднують результати різних моделей машинного навчання для підвищення точності виявлення шахрайства. Ця система здатна аналізувати транзакції в режимі реального часу, приймаючи рішення про блокування або дозвіл операції протягом мілісекунд.

Особливістю підходу HSBC є використання графових нейронних мереж для аналізу складних мереж транзакцій. Ці алгоритми дозволяють виявляти приховані зв'язки між різними учасниками фінансових операцій, що особливо ефективно для боротьби з організованими схемами відмивання грошей та фінансування тероризму.

Банк також впровадив систему федеративного навчання, яка дозволяє різним підрозділам HSBC у різних країнах навчати моделі машинного навчання на місцевих даних, не передаючи конфіденційну інформацію через кордони. Це особливо важливо в умовах різних національних вимог до захисту даних та банківської таємниці. [27, 28]

1.3.5. Barclays: інновації у сфері цифрового банкінгу

Barclays активно впроваджує технології штучного інтелекту для захисту своїх клієнтів від різноманітних форм фінансового шахрайства [29, 30]. Банк розробив комплексну систему, яка поєднує традиційні методи виявлення шахрайства з передовими технологіями машинного навчання [31].

Система Barclays використовує техніки глибокого навчання для аналізу зображень та документів, що дозволяє виявляти підроблені документи посвідчення особи та фінансові документи Алгоритми комп'ютерного зору

аналізують мікроскопічні деталі документів, виявляючи ознаки підробки, які неможливо помітити неозброєним оком [32].

Банк також впровадив систему аналізу голосових біометричних даних для захисту телефонного банкінгу [33, 34]. Ця система здатна ідентифікувати клієнтів за унікальними характеристиками їхнього голосу, виявляючи спроби несанкціонованого доступу до облікових записів [35]. Алгоритми аналізують не лише тембр голосу, але й особливості вимови, інтонації та ритму мовлення.

Особливу увагу Barclays приділяє захисту від шахрайства у сфері електронної комерції. Банк розробив спеціалізовані алгоритми для аналізу онлайн-транзакцій, які враховують специфіку різних типів транзакцій та особливості поведінки споживачів у цифровому середовищі [36].

1.3.6. DBS Bank: технологічний лідер Сінгапуру

DBS Bank, найбільший банк Сінгапуру, зарекомендував себе як один з найбільш технологічно прогресивних банків у світі [34, 37, 38]. Банк активно використовує штучний інтелект та машинне навчання не лише для виявлення шахрайства, але й для покращення загального клієнтського досвіду [39, 31, 40].

Система DBS Bank використовує передові алгоритми часових рядів для аналізу трендів у транзакційній поведінціюЦі алгоритми здатні виявляти тонкі зміни у патернах витрат, які можуть свідчити про компрометацію облікового запису або картки [41, 42]. Банк також впровадив систему предиктивної аналітики, яка дозволяє прогнозувати ймовірність шахрайських атак на основі аналізу історичних даних та поточних трендів у кіберзлочинності [40].

DBS Bank особливу увагу приділяє захисту мобільного банкінгу, розробивши унікальну систему багатофакторної аутентифікації, яка поєднує традиційні методи (паролі, SMS-коди) з передовими біометричними технологіями [43, 44, 45, 46, 47, 48]. Система аналізує біометричні дані клієнтів, включаючи відбитки пальців, геометрію обличчя та характеристики голосу, створюючи комплексний профіль безпеки [49, 46].

Банк також впровадив систему аналізу геолокаційних даних, яка відстежує місцезнаходження клієнтів під час здійснення транзакцій. Алгоритми машинного навчання аналізують патерни переміщення клієнтів, виявляючи аномальні локації або неможливі переміщення, які можуть свідчити про використання викрадених платіжних даних [50, 51].

1.3.7. OCBC: персоналізований підхід до безпеки

Oversea-Chinese Banking Corporation (OCBC), один з провідних банків Сінгапуру, розробив унікальну систему персоналізованого виявлення шахрайства, яка адаптується до індивідуальних особливостей кожного клієнта [47, 52]. Виявлення шахрайства з кредитними картками за допомогою машинного навчання стало одним з пріоритетних напрямків розвитку банку [52].

Система OCBC використовує передові алгоритми кластеризації для сегментації клієнтів на основі їхньої поведінки та демографічних характеристик [53]. Для кожного сегмента розробляються спеціалізовані моделі виявлення шахрайства, що дозволяє досягти вищої точності та зменшити кількість хибних спрацьовувань [54, 140].

Банк також впровадив систему континуального навчання, яка дозволяє моделям машинного навчання постійно адаптуватися до нових типів шахрайських схем [56]. Ця система використовує техніки онлайн-навчання, що дозволяє оновлювати моделі в режимі реального часу без необхідності повного.

OCBC особливу увагу приділяє захисту від шахрайства у сфері міжнародних переказів [57, 58]. Банк розробив спеціалізовані алгоритми для аналізу транскордонних транзакцій, які враховують специфіку різних юрисдикцій та валютних ризиків. Ця система здатна виявляти підозрілі схеми переказу коштів, які можуть використовуватися для відмивання грошей або фінансування незаконної діяльності.

1.3.8. Загальні тенденції та виклики

Аналіз досвіду провідних банків світу демонструє кілька ключових тенденцій у застосуванні машинного навчання для виявлення шахрайства. По-перше, всі

банки рухаються в напрямку створення інтегрованих екосистем безпеки, які поєднують різні типи даних та аналітичних підходів. По-друге, спостерігається тенденція до персоналізації систем безпеки, коли алгоритми адаптуються до індивідуальних особливостей кожного клієнта.

Водночас, банки стикаються з серйозними викликами у впровадженні технологій машинного навчання. Головними з них є необхідність забезпечення інтерпретованості моделей для регуляторних органів, захист від атак на самі системи машинного навчання та забезпечення справедливості алгоритмів для уникнення дискримінації окремих груп клієнтів.

Розвиток технологій також створює нові можливості для шахраїв. Використання технологій діпфейку, штучного інтелекту для генерації підроблених документів та інших передових технологій вимагає від банків постійного вдосконалення своїх систем захисту.

1.4. Популярні системи виявлення фінансового шахрайства

Сучасний ринок систем виявлення фінансового шахрайства характеризується наявністю декількох домінуючих платформ, кожна з яких використовує унікальні технологічні підходи та методології. SAS Fraud Management, FICO Falcon, Actimize та Feedzai представляють найбільш впливові та технологічно передові рішення, які використовуються провідними банками світу. Аналіз цих систем дозволяє зрозуміти поточний стан та майбутні напрямки розвитку технологій боротьби з фінансовим шахрайством. [59]

1.4.1. SAS Fraud Management

SAS Fraud Management представляє собою комплексну платформу, яка інтегрує передові аналітичні технології з глибоким розумінням специфіки банківського бізнесу. Ця система розроблялася протягом десятиліть та акумулювала досвід роботи з найбільшими фінансовими установами світу, що дозволило створити рішення, яке відповідає найвищим стандартам надійності та ефективності.

SAS Fraud Management побудована на модульній архітектурі, яка дозволяє банкам гнучко конфігурувати систему відповідно до своїх специфічних потреб. Центральним компонентом є аналітичний двигун SAS, який використовує власні алгоритми машинного навчання та статистичного аналізу.

Система підтримує обробку даних у реальному часі завдяки використанню технології SAS Event Stream Processing, яка дозволяє аналізувати мільйони транзакцій за секунду з мінімальною затримкою. Ця технологія особливо важлива для виявлення шахрайства з платіжними картками, де рішення повинні прийматися протягом мілісекунд.[60]

Платформа використовує гібридний підхід до зберігання даних, поєднуючи традиційні реляційні бази даних з сучасними big data технологіями, такими як Hadoop та Apache Spark. Це дозволяє ефективно обробляти як структуровані транзакційні дані, так і неструктуровані дані з різних джерел.

SAS Fraud Management використовує широкий спектр алгоритмів машинного навчання, включаючи як традиційні статистичні методи, так і сучасні техніки глибокого навчання. Система включає понад 100 готових алгоритмів виявлення шахрайства, які можуть бути адаптовані до специфічних потреб банку. Нейронні мережі у SAS Fraud Management використовуються для виявлення складних нелінійних залежностей у даних. Система підтримує різні архітектури нейронних мереж, включаючи багатошарові перцептрони, рекурентні нейронні мережі для аналізу послідовностей транзакцій та автоенкодери для виявлення аномалій. Поведінковий аналіз є однією з ключових переваг SAS Fraud Management. Система створює детальні профілі поведінки для кожного клієнта, аналізуючи не лише транзакційну активність, але й особливості взаємодії з банківськими каналами. Алгоритми відстежують зміни у поведінці клієнтів та автоматично адаптують порги виявлення шахрайства. [60]

Система також використовує передові техніки аналізу часових рядів для виявлення сезонних та трендових змін у поведінці клієнтів. Це дозволяє відрізняти природні зміни у витратах (наприклад, під час свят чи відпусток) від потенційно шахрайської активності.

SAS Fraud Management розроблена з урахуванням складності банківської IT-інфраструктури та підтримує інтеграцію з широким спектром банківських систем. Платформа надає готові інтеграції для найпопулярніших банківських платформ та систем управління ризиками. Система підтримує як хмарні, так і локальні розгортання, що дозволяє банкам вибирати оптимальну модель розгортання відповідно до їхніх регуляторних вимог та IT-стратегії. SAS також пропонує гібридні рішення, які поєднують переваги хмарних та локальних розгортань.

1.4.2. FICO Falcon Platform

FICO Falcon Platform є однією з найстаріших та найбільш поширених систем виявлення шахрайства у світі, захищаючи понад 2.6 мільярда платіжних карток по всьому світу [61]. Система розроблялася протягом понад 30 років та постійно еволюціонувала, інкорпоруючи найновіші досягнення у сфері машинного навчання та аналітики [62]. FICO Falcon побудована на власній технологічній платформі FICO Decision Management Suite, яка забезпечує високопродуктивну обробку даних у реальному часі [63]. Система здатна обробляти тисячі транзакцій за секунду, що робить її однією з найбільш масштабованих платформ на ринку.

ЇЇ архітектура базується на принципах розподіленої обробки та використовує передові техніки кешування та паралелізації для забезпечення низької затримки [64]. FICO Falcon може розгортатися як у хмарі, так і локально, підтримуючи різні моделі розгортання відповідно до потреб клієнтів [65]. Система використовує власний формат зберігання даних, оптимізований для швидкого доступу до історичних даних та обчислення агрегаційних ознак у реальному часі. Це дозволяє системі швидко обчислювати складні поведінкові характеристики для кожної транзакції [66, 67].

FICO Falcon використовує власні нейронні мережі, розроблені спеціально для виявлення шахрайства [68, 69]. Ці мережі характеризуються високою швидкістю навчання та здатністю адаптуватися до нових типів шахрайства без повного

перенавчання [70]. Також використовується техніка "профілювання", яка створює індивідуальні профілі поведінки для кожного власника картки [71]. Ці профілі включають інформацію про географічні патерни використання, часові характеристики, типи та суми транзакцій. Нейронні мережі аналізують відхилення від цих профілів для виявлення потенційно шахрайських транзакцій [67, 70].

Адаптивні алгоритми FICO Falcon автоматично налаштовують пороги виявлення шахрайства на основі зворотного зв'язку про підтверджені випадки шахрайства. Це дозволяє системі постійно покращувати свою точність без втручання аналітиків [65, 157].

Однією з унікальних особливостей FICO Falcon є використання консорціумної моделі, яка дозволяє банкам ділитися анонімізованою інформацією про шахрайські атаки [73, 74, 75, 76]. Ця модель значно підвищує ефективність виявлення нових типів шахрайства, оскільки досвід одного банку може бути використаний для захисту клієнтів інших банків [76]. Консорціумні дані дозволяють системі виявляти глобальні тренди у шахрайстві та швидко адаптуватися до нових загроз [75]. FICO підтримує кілька консорціумів для різних регіонів та типів карток, що забезпечує релевантність даних для конкретних.

1.4.3. Actimize

Nice Actimize (раніше відома як Actimize) є провідним постачальником рішень для боротьби з фінансовими злочинами, що охоплюють не лише виявлення шахрайства, але й боротьбу з відмиванням грошей, санкційний скринінг та інші аспекти фінансової безпеки [77]. Actimize пропонує інтегровану платформу, яка об'єднує різні аспекти фінансової безпеки в єдиній системі. Це дозволяє банкам мати цілісний погляд на ризики та підвищує ефективність виявлення складних схем фінансових злочинів, які можуть охоплювати кілька типів незаконної діяльності [78].

Платформа використовує концепцію "360-градусного погляду на клієнта", інтегруючи дані з різних джерел для створення комплексного профілю ризику [79, 78, 80]. Це включає не лише транзакційні дані, але й інформацію про взаємодію клієнта з банком через різні канали, дані соціальних мереж та зовнішні джерела ризикової інформації.

Actimize активно використовує передові технології штучного інтелекту, включаючи обробку природної мови для аналізу неструктурованих даних, комп'ютерний зір для аналізу документів та графові алгоритми для виявлення складних мереж зв'язків [81, 82, 83]. Система використовує ансамблеві методи машинного навчання, які поєднують результати кількох різних алгоритмів для підвищення точності виявлення [82]. Це включає комбінування традиційних правил з алгоритмами машинного навчання та використання технік мета-навчання для оптимізації цієї комбінації. Actimize розробила власні алгоритми для роботи з графовими даними, які дозволяють виявляти складні схеми шахрайства та відмивання грошей [81, 80]. Ці алгоритми аналізують мережі транзакцій для виявлення підозрілих патернів, таких як циркулярні схеми переказів або структурування транзакцій [80].

Платформа Actimize підтримує як аналіз у реальному часі для миттєвого виявлення шахрайства, так і глибокий історичний аналіз для виявлення складних схем, які розгортаються протягом тривалого періоду часу [84]. Система використовує технології потокової обробки даних для аналізу транзакцій у реальному часі, забезпечуючи мінімальну затримку у прийнятті. Водночас, паралельно працюють алгоритми пакетної обробки, які проводять глибокий аналіз історичних даних для виявлення довгострокових патернів.

1.4.4. Feedzai

Feedzai представляє нове покоління систем виявлення шахрайства, побудованих з нуля для роботи у хмарному середовищі та використання найсучасніших технологій машинного навчання та штучного інтелекту [85, 86, 87].

Feedzai розроблена як хмарно-орієнтована платформа (cloud-native), що дозволяє їй ефективно використовувати переваги хмарних технологій, включаючи еластичність, масштабованість та економічну ефективність. Система може автоматично масштабуватися відповідно до навантаження, забезпечуючи оптимальну продуктивність при мінімальних витратах].

Архітектура базується на мікросервісах, що дозволяє незалежно розробляти, розгортати та масштабувати різні компоненти системи. Це забезпечує високу гнучкість та можливість швидкого впровадження нових функцій [87]. Feedzai активно використовує технології автоматизованого машинного навчання (AutoML), які дозволяють банкам створювати та оптимізувати моделі виявлення шахрайства без глибоких знань у сфері data science [88, 89, 90]. Система автоматично підбирає оптимальні алгоритми, налаштовує їхні параметри та створює ансамблі моделей [89].

Платформа включає візуальний інтерфейс для створення моделей, який дозволяє бізнес-аналітикам працювати з машинним навчанням без необхідності програмування. Це значно прискорює процес розробки та впровадження нових моделей [87]. Feedzai оптимізована для роботи з надзвичайно низькими затримками, забезпечуючи прийняття рішень протягом мілісекунд [91]. Система використовує передові техніки кешування та попереднього обчислення для мінімізації часу обробки кожної транзакції.

Платформа підтримує різні режими обробки, від синхронного аналізу для платіжних карток до асинхронного аналізу для банківських переказів, оптимізуючи продуктивність для кожного типу транзакцій [92].

1.4.5. Порівняльний аналіз.

Порівнюючи розглянуті системи, можна виділити кілька ключових тенденцій у розвитку технологій виявлення шахрайства. По-перше, всі системи рухаються в напрямку більшої автоматизації та зменшення залежності від ручної настройки правил. По-друге, зростає важливість аналізу у реальному часі та здатності швидко адаптуватися до нових загроз.

Спостерігається тенденція до інтеграції різних типів аналізу в єдиних платформах, що дозволяє банкам мати цілісний погляд на ризики. Також зростає важливість хмарних технологій та мікросервісних архітектур, які забезпечують гнучкість та масштабованість.

Всі розглянуті системи активно інвестують у розвиток технологій штучного інтелекту, особливо у сфері обробки природної мови та комп'ютерного зору, що відкриває нові можливості для аналізу неструктурованих даних.

1.5. Методологічні підходи до розробки та інтеграції систем виявлення шахрайства.

Успішне впровадження систем виявлення шахрайства на основі машинного навчання вимагає дотримання структурованих методологічних підходів, які забезпечують якість, надійність та ефективність розроблених рішень. Банківська галузь використовує три основні методології: CRISP-DM (Cross-Industry Standard Process for Data Mining), KDD (Knowledge Discovery in Databases) та SEMMA (Sample, Explore, Modify, Model, Assess). Кожна з цих методологій має свої особливості та переваги при застосуванні до специфічних завдань виявлення фінансового шахрайства.

1.5.1. CRISP-DM: галузевий стандарт для проектів data mining

CRISP-DM залишається найбільш поширеною методологією для реалізації проектів аналізу даних у банківській сфері завдяки своїй структурованості та адаптивності до специфічних потреб фінансових установ. Ця методологія складається з шести взаємопов'язаних етапів, які можуть виконуватися ітеративно та адаптуватися до конкретних вимог проекту виявлення шахрайства. [93]

Перший етап CRISP-DM у контексті виявлення шахрайства передбачає глибоке розуміння специфічних потреб банку у сфері фінансової безпеки. Команда проекту повинна ретельно проаналізувати поточні загрози, з якими стикається фінансова установа, включаючи типи шахрайства, які найчастіше виникають, їхній потенційний вплив на бізнес та існуючі методи виявлення.

На цьому етапі важливо визначити ключові показники ефективності системи виявлення шахрайства. Традиційно банки фокусуються на таких метриках, як чутливість (sensitivity) - здатність виявляти справжні випадки шахрайства, специфічність (specificity) - здатність правильно ідентифікувати законні транзакції, та загальна точність системи. Однак у сучасних умовах все більшого значення набувають додаткові метрики, такі як час виявлення шахрайства, вартість хибних спрацьовувань та рівень задоволеності клієнтів.

Критично важливим аспектом цього етапу є встановлення балансу між безпекою та зручністю використання. Надто чутливі системи можуть блокувати законні транзакції, погіршуючи клієнтський досвід, тоді як недостатньо чутливі системи можуть пропускати шахрайські операції. Команда проекту повинна працювати з бізнес-підрозділами для визначення оптимального балансу відповідно до ризик-апетиту банку.

Також на цьому етапі визначаються регуляторні вимоги, які повинна відповідати система. Банки працюють у високорегульованому середовищі, тому система виявлення шахрайства повинна відповідати вимогам щодо інтерпретованості рішень, аудиту та звітності. Це особливо важливо в контексті європейського регулювання GDPR та American Fair Credit Reporting Act.

Другий етап передбачає детальний аналіз доступних даних для розробки системи виявлення шахрайства. Банки мають доступ до величезних обсягів різноманітних даних, включаючи транзакційні дані, дані про клієнтів, інформацію про пристрої та канали доступу, геолокаційні дані та зовнішні дані про репутацію.

Транзакційні дані включають інформацію про суму операції, тип транзакції, час проведення, рахунки відправника та отримувача, валюту та канал проведення операції. Ці дані є основою для більшості алгоритмів виявлення шахрайства, оскільки вони містять пряму інформацію про фінансові операції.

Дані про клієнтів включають демографічну інформацію, історію взаємодії з банком, кредитну історію та інформацію про продукти, якими користується клієнт. Ці дані дозволяють створювати персоналізовані профілі ризику та

адаптувати алгоритми виявлення шахрайства до специфічних характеристик різних сегментів клієнтів.

Особливу увагу приділяють якості даних, оскільки неякісні дані можуть серйозно вплинути на ефективність системи виявлення шахрайства. Команда проекту повинна провести детальний аналіз повноти, точності, консистентності та актуальності даних. Часто виявляються проблеми з пропущеними значеннями, дублікатами, викидами та неконсистентними форматами даних.

Важливим аспектом цього етапу є аналіз розподілу класів у даних. Шахрайські транзакції зазвичай становлять менше 1% від загальної кількості операцій, що створює проблему незбалансованих класів. Це вимагає використання спеціальних техніок для роботи з незбалансованими даними, таких як oversampling, undersampling або використання спеціальних метрик оцінки.

Етап підготовки даних є найбільш трудомістким у процесі розробки системи виявлення шахрайства, часто займаючи до 80% загального часу проекту. Цей етап включає очищення даних, створення нових ознак, трансформацію змінних та підготовку даних для навчання моделей.

Очищення даних включає виявлення та обробку пропущених значень, видалення дублікатів, виправлення помилок та обробку викидів. У контексті виявлення шахрайства особливо важливо правильно обробляти викиди, оскільки вони можуть як містити цінну інформацію про шахрайські операції, так і бути результатом помилок у даних.

Створення нових ознак (feature engineering) є критично важливим для ефективності системи виявлення шахрайства. Експерти з предметної області працюють разом з data scientists для створення ознак, які відображають специфічні патерни шахрайської поведінки. Наприклад, можуть створюватися ознаки, що відображають частоту транзакцій клієнта у певні часові періоди, географічну віддаленість між послідовними транзакціями, або відхилення поточної поведінки від історичних патернів.

Агрегаційні ознаки відіграють особливу роль у виявленні шахрайства. Створюються ознаки, що відображають поведінку клієнта за різні часові вікна

(година, день, тиждень, місяць), включаючи загальну суму транзакцій, кількість операцій, середню суму операції та інші статистичні характеристики.

Темпоральні ознаки також мають велике значення, оскільки шахрайці часто демонструють специфічні часові патерни. Створюються ознаки, що відображають час доби, день тижня, сезонність та інші часові характеристики транзакцій.

Етап моделювання включає вибір відповідних алгоритмів машинного навчання, налаштування їхніх параметрів та навчання моделей на підготовлених даних. У сфері виявлення шахрайства використовується широкий спектр алгоритмів, кожен з яких має свої переваги та недоліки.

Логістична регресія залишається популярним вибором завдяки своїй інтерпретованості та стабільності. Цей алгоритм дозволяє легко пояснити, чому конкретна транзакція була класифікована як підозріла, що важливо для регуляторних вимог та взаємодії з клієнтами.

Дерева рішень та ансамблеві методи, такі як Random Forest та Gradient Boosting, широко використовуються завдяки своїй здатності моделювати складні нелінійні залежності та автоматично виявляти важливі взаємодії між ознаками. Ці алгоритми особливо ефективні для роботи з категоріальними змінними та можуть надавати інформацію про важливість різних ознак.

Нейронні мережі, особливо глибокі нейронні мережі, показують відмінні результати у виявленні складних патернів шахрайства. Автоенкодери використовуються для виявлення аномалій, тоді як рекурентні нейронні мережі ефективні для аналізу послідовностей транзакцій.

Методи навчання без учителя, такі як кластеризація та виявлення аномалій, використовуються для виявлення нових типів шахрайства, які не представлені в історичних даних. Алгоритми Isolation Forest, One-Class SVM та Local Outlier Factor показують хороші результати у виявленні атипової поведінки.

Важливим аспектом моделювання є робота з незбалансованими даними. Використовуються техніки SMOTE (Synthetic Minority Oversampling Technique) для збільшення кількості прикладів шахрайських транзакцій, алгоритми cost-

sensitive learning для призначення різних вагів помилкам різних типів, та спеціальні метрики оцінки, такі як F1-score, precision, recall та AUC-ROC.

Етап оцінки включає всебічну перевірку якості розроблених моделей з використанням різноманітних метрик та методів валідації. У сфері виявлення шахрайства особливо важливо використовувати часову валідацію, коли моделі навчаються на історичних даних та тестуються на більш свіжих даних, щоб імітувати реальні умови використання.

Традиційні метрики точності можуть бути оманливими у випадку незбалансованих даних, тому використовуються спеціалізовані метрики. Precision (точність) показує, яка частка транзакцій, класифікованих як шахрайські, дійсно є шахрайськими. Recall (повнота) показує, яка частка справжніх шахрайських транзакцій була виявлена системою.

F1-score надає збалансовану оцінку, що поєднує precision та recall. AUC-ROC (Area Under the Receiver Operating Characteristic Curve) показує здатність моделі розрізняти класи при різних порогах класифікації.

Специфічні для бізнесу метрики включають вартість хибних спрацьовувань, час виявлення шахрайства та вплив на клієнтський досвід. Розраховується загальна економічна ефективність системи, враховуючи вартість запобігання шахрайству та витрати на обробку хибних спрацьовувань.

Проводиться аналіз стабільності моделей у часі, оскільки шахрайські схеми постійно еволюціонують. Моделі тестуються на даних з різних часових періодів для оцінки їхньої стійкості до концептуального дрейфу.

Останній етап CRISP-DM включає розгортання моделей у виробничому середовищі та створення інфраструктури для їхньої підтримки. У банківській сфері це особливо складний процес через високі вимоги до надійності, безпеки та продуктивності.

Розгортання систем виявлення шахрайства вимагає створення високопродуктивної інфраструктури, здатної обробляти мільйони транзакцій у реальному часі. Використовуються технології stream processing, такі як Apache Kafka та Apache Storm, для забезпечення низької затримки обробки.

Створюється система моніторингу якості моделей, яка відстежує ключові метрики продуктивності та автоматично сповіщає про погіршення якості. Впроваджуються механізми автоматичного перенавчання моделей при виявленні концептуального дрейфу.

1.5.2. KDD: науковий підхід до виявлення знань

Методологія Knowledge Discovery in Databases (KDD) представляє більш академічний підхід до аналізу даних, що особливо корисний для дослідницьких проектів та розробки інноваційних методів виявлення шахрайства. KDD складається з дев'яти етапів, які забезпечують системний підхід до виявлення нових знань з великих масивів даних.[94]

Перший етап KDD передбачає глибоке розуміння специфіки банківського шахрайства як предметної області. Це включає вивчення різних типів фінансових злочинів, їхніх характеристик та методів, які використовують шахраї для приховування своєї діяльності.

Команда проекту повинна розуміти регуляторне середовище, в якому працює банк, включаючи вимоги до боротьби з відмиванням грошей (AML), фінансуванням тероризму (CFT) та захисту прав споживачів. Це знання критично важливе для розробки систем, які не лише ефективно виявляють шахрайство, але й відповідають усім нормативним вимогам.

KDD приділяє особливу увагу ідентифікації та розумінню всіх доступних джерел даних. У банківському контексті це включає не лише внутрішні дані банку, але й зовнішні джерела інформації, такі як бюро кредитних історій, бази даних про шахрайство, дані соціальних мереж та інші релевантні джерела.

Проводиться детальний аналіз структури даних, їхніх взаємозв'язків та потенційної цінності для виявлення шахрайства. Особлива увага приділяється ідентифікації даних, які можуть містити приховані патерни шахрайської поведінки.

Цей етап включає відбір релевантних підмножин даних для аналізу. У контексті виявлення шахрайства це може включати фокусування на конкретних типах транзакцій, часових періодах або сегментах клієнтів.

Важливим аспектом є балансування між повнотою даних та практичністю їх обробки. Занадто великі набори даних можуть ускладнити аналіз, тоді як занадто малі можуть не містити достатньо інформації для виявлення складних патернів.

KDD приділяє значну увагу етапу очищення даних, визнаючи його критичну важливість для якості кінцевих результатів. Цей етап включає виявлення та виправлення помилок, обробку пропущених значень та стандартизацію форматів даних.

У банківському контексті особливо важливо забезпечити консистентність даних з різних систем, оскільки банки часто використовують множину різних IT-систем, які можуть мати різні формати та стандарти даних.

Етап трансформації включає перетворення очищених даних у формат, придатний для аналізу. Це може включати нормалізацію числових змінних, кодування категоріальних змінних та створення похідних ознак.

У контексті виявлення шахрайства особливо важливими є часові трансформації, які дозволяють моделювати динамічні аспекти поведінки клієнтів. Створюються ковзні середні, трендові характеристики та інші часові ознаки.

KDD передбачає чіткий вибір конкретного завдання data mining, що буде вирішуватися. У сфері виявлення шахрайства це може бути класифікація (визначення, чи є транзакція шахрайською), кластеризація (групування подібних транзакцій), виявлення аномалій (ідентифікація незвичайної поведінки) або виявлення послідовностей (аналіз патернів у часових рядах).

Вибір алгоритму data mining включає вибір конкретних алгоритмів для вирішення обраного завдання. KDD підкреслює важливість експериментування з різними алгоритмами та їх комбінаціями для досягнення найкращих результатів. У сфері виявлення шахрайства часто використовуються ансамблеві методи, які поєднують результати кількох різних алгоритмів. Це може включати

комбінування алгоритмів, що базуються на різних принципах, таких як дерева рішень, нейронні мережі та методи на основі відстаней.

Етап застосування включає фактичне виконання обраних алгоритмів на підготовлених даних. KDD підкреслює важливість ітеративного підходу, коли алгоритми можуть виконуватися кілька разів з різними параметрами для оптимізації результатів.

Останній етап KDD включає інтерпретацію отриманих результатів та їх оцінку з точки зору предметної області. Це особливо важливо у банківській сфері, де результати аналізу повинні бути не лише точними, але й зрозумілими для бізнес-користувачів та регуляторних органів.

1.5.3. SEMMA: SAS підхід до аналітики

Методологія SEMMA, розроблена компанією SAS, представляє практичний підхід до реалізації проектів аналітики, що особливо популярний серед банків, які використовують платформу SAS для своїх аналітичних потреб. [95]

Перший етап SEMMA включає створення репрезентативної вибірки даних для аналізу. У контексті виявлення шахрайства це особливо складне завдання через рідкісність шахрайських транзакцій та необхідність забезпечення їх адекватного представлення у вибірці.

Використовуються стратифіковані методи вибірки, які забезпечують пропорційне представлення різних типів транзакцій та клієнтських сегментів. Особлива увага приділяється забезпеченню часової репрезентативності, щоб вибірка відображала сезонні та інші часові закономірності у даних. Створюються окремі вибірки для навчання, валідації та тестування моделей. Важливо забезпечити, що тестова вибірка містить найбільш свіжі дані, щоб оцінка моделей відображала їхню продуктивність у реальних умовах. [96]

Етап дослідження включає детальний експлораторний аналіз даних для виявлення закономірностей, аномалій та потенційних проблем якості. Використовуються різноманітні статистичні та візуальні методи для розуміння структури та характеристик даних.

Проводиться аналіз розподілів змінних, кореляційний аналіз та виявлення викидів. Особлива увага приділяється аналізу відмінностей між шахрайськими та законними транзакціями для ідентифікації потенційних дискримінантних ознак. Використовуються методи візуалізації даних для виявлення прихованих патернів. Створюються графіки розподілів, scatter plots, heat maps та інші види візуалізацій, які допомагають аналітикам краще зрозуміти дані. [97]

Етап модифікації включає трансформацію змінних для покращення їхньої аналітичної цінності. Це може включати створення нових ознак, трансформацію існуючих змінних та видалення нерелевантних або надлишкових ознак.

У контексті виявлення шахрайства часто створюються агрегаційні ознаки, які відображають поведінку клієнтів за різні часові періоди. Також використовуються техніки нормалізації та стандартизації для забезпечення сумісності змінних з різними масштабами. Застосовуються методи зменшення розмірності, такі як Principal Component Analysis (PCA), для зменшення кількості ознак та усунення мультиколінеарності.

Етап моделювання включає застосування різноманітних алгоритмів машинного навчання для створення предиктивних моделей. SEMMA підкреслює важливість експериментування з різними типами моделей для знаходження найкращого рішення. Використовуються як традиційні статистичні методи, такі як логістична регресія, так і сучасні алгоритми машинного навчання, включаючи нейронні мережі та ансамблеві методи. Особлива увага приділяється налаштуванню гіперпараметрів моделей для оптимізації їхньої продуктивності.

Останній етап SEMMA включає всебічну оцінку якості створених моделей. Використовуються різноманітні метрики та методи валідації для забезпечення надійності та узагальнюючої здатності моделей. Проводиться порівняльний аналіз різних моделей для вибору найкращого рішення. Оцінюється не лише статистична точність моделей, але й їхня практична корисність для бізнесу, включаючи швидкість роботи, інтерпретованість та вартість впровадження.

1.5.4. Інтеграція методологій у банківській практиці

У реальній банківській практиці часто використовуються гібридні підходи, які поєднують елементи різних методологій відповідно до специфічних потреб проекту. Наприклад, структурованість CRISP-DM може поєднуватися з науковою строгістю KDD та практичністю SEMMA. Важливим аспектом є адаптація методологій до специфічних вимог банківської галузі, включаючи регуляторні обмеження, вимоги до безпеки даних та необхідність забезпечення високої доступності систем. Банки часто розробляють власні модифікації стандартних методологій, які враховують їхні унікальні потреби та обмеження.

Розвиток технологій також впливає на еволюцію методологій. Поява нових типів даних, таких як дані соціальних мереж та IoT, вимагає адаптації традиційних підходів. Розвиток технологій реального часу створює нові можливості для швидкого виявлення та реагування на шахрайство.

ВИСНОВКИ ДО РОЗДІЛУ 1

У результаті проведеного теоретико-методологічного дослідження шахрайських операцій у банкінгу було отримано наступні ключові висновки та результати.

Систематизація наукових підходів до визначення сутності банківського шахрайства дозволила встановити, що цей феномен являє собою багатоаспектне явище, яке охоплює широкий спектр протиправних дій у фінансовій сфері. Проведена класифікація шахрайських операцій за різними критеріями (суб'єкти, об'єкти, способи здійснення, технології) дала змогу створити комплексну типологію, що слугує основою для розробки ефективних систем протидії.

Аналіз еволюції методів виявлення шахрайських операцій продемонстрував кардинальну трансформацію підходів до вирішення цієї проблеми. Встановлено, що традиційні методи, засновані на системах правил та статистичному аналізі, хоча й зберігають певну актуальність, проте мають суттєві обмеження щодо адаптивності та точності виявлення складних схем шахрайства. Натомість технології машинного навчання та штучного інтелекту демонструють значно

вищу ефективність, забезпечуючи можливість автоматичного виявлення нових патернів шахрайства та адаптації до змінюваного ландшафту загроз.

Детальний розгляд алгоритмічних підходів виявив, що найбільш перспективними є гібридні системи, які поєднують переваги різних методів. Зокрема, кластеризація та виявлення аномалій ефективні для ідентифікації нестандартних транзакцій, логістична регресія забезпечує інтерпретованість результатів, нейронні мережі та глибоке навчання дозволяють працювати зі складними нелінійними залежностями, а графові моделі розкривають приховані зв'язки між учасниками операцій.

Аналіз світового досвіду провідних фінансових установ засвідчив наявність декількох ключових тенденцій у розвитку антишахрайських систем. По-перше, спостерігається перехід від реактивного до проактивного підходу, коли системи не лише виявляють існуючі інциденти, але й прогнозують потенційні загрози. По-друге, відбувається інтеграція різних типів даних та аналітичних методів у єдиних платформах, що забезпечує комплексний погляд на ризики. По-третє, активно впроваджуються технології обробки даних у реальному часі, що дозволяє миттєво реагувати на підозрілі операції.

Дослідження популярних комерційних систем виявлення фінансового шахрайства показало, що ринок характеризується високим рівнем концентрації та технологічної зрілості. Провідні рішення, такі як FICO Falcon, SAS Fraud Management, NICE Actimize та Feedzai, демонструють схожі архітектурні принципи, але відрізняються акцентами на конкретних технологічних рішеннях та ринкових сегментах. Особливо помітною є тенденція до хмарних рішень та використання автоматизованого машинного навчання.

Критичний аналіз методологічних підходів до розробки систем виявлення шахрайства виявив, що найширшого застосування набули методології CRISP-DM, KDD та SEMMA. Кожна з них має свої переваги: CRISP-DM забезпечує структурованість та практичність, KDD орієнтована на наукову обґрунтованість, а SEMMA оптимізована для роботи з інструментами SAS. Встановлено, що

успішна реалізація проектів потребує адаптації цих методологій до специфіки банківської діяльності та вимог регуляторного середовища.

Узагальнюючи результати дослідження, можна констатувати, що сучасний стан протидії банківському шахрайству характеризується переходом до інтелектуальних систем, заснованих на передових технологіях аналізу даних. Водночас залишаються відкритими питання оптимального поєднання різних методів, забезпечення прозорості прийнятих рішень, дотримання вимог конфіденційності та адаптації до специфіки національних ринків, що обумовлює актуальність подальших досліджень у цій сфері.

Проведений аналіз створює теоретико-методологічну основу для розробки рекомендацій щодо вдосконалення систем виявлення шахрайських операцій у вітчизняній банківській практиці, що становитиме предмет наступних розділів дослідження.

РОЗДІЛ 2

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ВИЯВЛЕННЯ ШАХРАЙСЬКИХ ОПЕРАЦІЙ

2.1. Опис досліджуваних даних

Розробка ефективних систем автоматичного виявлення шахрайських операцій потребує глибокого розуміння структури даних та застосування сучасних методів машинного навчання. Підхід базується на комплексному аналізі транзакційних даних, що включає не лише самі операції, але й контекстну інформацію про користувачів, їхні платіжні картки та характеристики спрямування витрат.

Для створення надійної моделі виявлення шахрайства ми використовуємо багатовимірний набір даних, який забезпечує об’єктивний погляд на кожну транзакцію. Цей набір даних був отриманий з платформи Kaggle [98] та містить реальні характеристики банківських операцій, що робить його ідеальним для практичного моделювання. Представлений набір даних був розроблений для банку LOL Bank Pvt. Ltd. з метою створення системи виявлення шахрайських операцій у режимі реального часу. Датасет містить історичні дані про банківські транзакції, що є критично важливими для розробки машинного навчання моделей, здатних розпізнавати підозрілі фінансові операції серед легітимних транзакцій клієнтів.

Структура даних охоплює широкий спектр інформації про кожну транзакцію, включаючи персональні дані клієнтів, деталі операцій, інформацію про спрямування витрат та технічні характеристики пристроїв, використаних для здійснення транзакцій. Такий комплексний підхід до збору даних дозволяє створити багатовимірну модель для аналізу поведінкових патернів та виявлення аномалій у фінансових операціях.

Набір даних складається з 24 основних атрибутів, які можна умовно розділити на декілька тематичних груп. Перша група включає ідентифікаційні дані клієнтів, серед яких унікальний ідентифікатор клієнта (Customer_ID), повне ім'я

(Customer_Name), демографічні характеристики такі як стать (Gender) та вік (Age). Географічна інформація представлена полями штату (State) та міста (City) проживання клієнта, що дозволяє аналізувати територіальні особливості фінансових операцій.

Банківська інформація включає дані про філію банку (Bank_Branch), де клієнт обслуговується, та тип рахунку (Account_Type), який може бути ощадним, поточним або іншого типу. Ці поля надають контекст для розуміння профілю клієнта та його банківських потреб.

Транзакційні дані формують основу аналітичної моделі та включають унікальний ідентифікатор транзакції (Transaction_ID), дату (Transaction_Date) та точний час (Transaction_Time) здійснення операції. Сума транзакції (Transaction_Amount) та валюта (Transaction_Currency) визначають фінансові параметри операції, тоді як тип транзакції (Transaction_Type) категоризує операції за їх природою: зняття готівки, депозит, переказ або інші види фінансових операцій.

Інформація про спрямування витрат представлена унікальним ідентифікатором продавця (Merchant_ID) та категорією бізнесу (Merchant_Category), що дозволяє аналізувати патерни витрат клієнтів у різних сферах економічної діяльності, таких як роздрібна торгівля, онлайн-комерція або туристичні послуги.

Особливу увагу заслуговують технічні поля, які фіксують метадані про пристрої та місця здійснення транзакцій. Поле пристрою транзакції (Transaction_Device) та тип пристрою (Device_Type) надають інформацію про технологічні засоби, використані для операції, що може включати мобільні пристрої, настільні комп'ютери, смартфони або ноутбуки.

Геолокаційна інформація зберігається у полі місця транзакції (Transaction_Location), яке містить координати широти та довготи, що дозволяє проводити просторовий аналіз операцій та виявляти географічні аномалії у поведінці клієнтів.

Баланс рахунку після транзакції (Account_Balance) є критично важливим показником, який дозволяє оцінити фінансовий стан клієнта та виявити потенційно підозрілі операції, що призводять до незвичайних змін у балансі рахунку.

Центральним елементом датасету є бінарна змінна Is_Fraud, яка приймає значення 1 для шахрайських транзакцій та 0 для легітимних операцій. Ця змінна служить цільовою для навчання моделей машинного навчання з учителем та дозволяє оцінювати ефективність розроблених алгоритмів виявлення шахрайства.

Контактна інформація клієнтів представлена номером телефону (Customer_Contact) та електронною поштою (Customer_Email), що забезпечує можливість швидкого зв'язку з клієнтами у випадку виявлення підозрілих операцій.

Додатковий контекст надає поле опису транзакції (Transaction_Description), яке містить стислі текстові описи операцій, такі як "покупка", "переказ" або інші характеристики транзакції.

Представлений набір даних характеризується високою інформативністю та багатовимірністю, що створює сприятливі умови для розробки складних аналітичних моделей. Комбінація демографічних, географічних, часових, фінансових та технічних характеристик дозволяє проводити комплексний аналіз поведінкових патернів клієнтів та виявляти тонкі сигнали потенційного шахрайства.

Водночас, датасет представляє типові виклики для задач виявлення шахрайства, зокрема проблему незбалансованості класів, коли шахрайські транзакції складають незначний відсоток від загальної кількості операцій. Це потребує використання спеціалізованих методів обробки незбалансованих даних та метрик оцінювання, які враховують асиметричність розподілу класів.

Різноманітність типів даних у наборі, від числових величин до категоріальних змінних та геопросторової інформації, вимагає комплексного підходу до попередньої обробки та інженерії ознак, що може включати нормалізацію,

кодування категоріальних змінних та створення похідних ознак на основі часових та просторових патернів.

2.2. Дослідницький аналіз набору даних

Дослідницький аналіз даних (Exploratory Data Analysis, EDA) є фундаментальним етапом у розробці системи виявлення шахрайських операцій у банківському секторі. Даний аналіз спрямований на глибоке розуміння структури даних, виявлення прихованих закономірностей та підготовку обґрунтованих рекомендацій для подальшого моделювання.

Аналізований набір даних містить 200 000 записів банківських трансакцій, що охоплюють 24 різноманітні змінні. Загальний обсяг даних складає 257,95 МБ, структура даних включає як ідентифікаційні поля (Customer_ID, Transaction_ID), так і операційні характеристики (суми, дати, типи операцій), що забезпечує всебічне покриття аспектів банківської діяльності.

Кожен запис представляє окрему банківську трансакцію з детальною інформацією про клієнта, включаючи демографічні характеристики (стать, вік), географічне розташування (штат, місто), тип рахунку та відділення банку. Операційна частина даних охоплює параметри самої трансакції: дату, час, суму, тип операції, категорію спрямування витрат, пристрій проведення операції та валюту. Критично важливою є бінарна змінна Is_Fraud, що індикує належність трансакції до категорії шахрайських операцій.

Типи даних розподілені наступним чином: об'єктні змінні (14 полів) включають текстові та категоріальні дані, числові змінні (5 полів) представлені цілими числами та числами з плаваючою комою, часові змінні (2 поля) забезпечують темпоральний аналіз операцій. Така структура створює оптимальні умови для багатовимірного аналізу шахрайських патернів.

Критичним аспектом подальшого аналізу є оцінка якості вхідних даних (див дод.Б). Проведена перевірка виявила відсутність пропущених значень у всіх 24 змінних, що є виключно позитивною характеристикою набору даних. Така повнота інформації елімінує необхідність у складних процедурах імпутації

пропущених значень та забезпечує надійність подальших статистичних висновків.

Аналіз на предмет дублікатів також дав позитивні результати - у наборі даних відсутні повторювані записи. Це гарантує унікальність кожної трансакції та виключає можливість штучного завищення окремих статистичних показників через дублювання інформації. Унікальність ідентифікаторів клієнтів та трансакцій підтверджує коректність процедур збору та обробки даних на етапі формування набору.

Розподіл типів даних відповідає логічній структурі банківської інформації. Категоріальні змінні представлені у текстовому форматі, що дозволяє проводити детальний аналіз розподілів за різними сегментами. Числові змінні мають відповідні типи для статистичних обчислень, а часові поля коректно форматовані для темпорального аналізу.

Центральним елементом дослідження є аналіз цільової змінної Is_Fraud, що визначає належність операції до категорії шахрайських. Серед 200 000 трансакцій 10 088 (5,04%) класифіковані як шахрайські, тоді як 189 912 (94,96%) є легітимними операціями (див. рис.2.1). Такий розподіл характеризує набір даних як відносно збалансований у контексті задач машинного навчання для виявлення шахрайств.

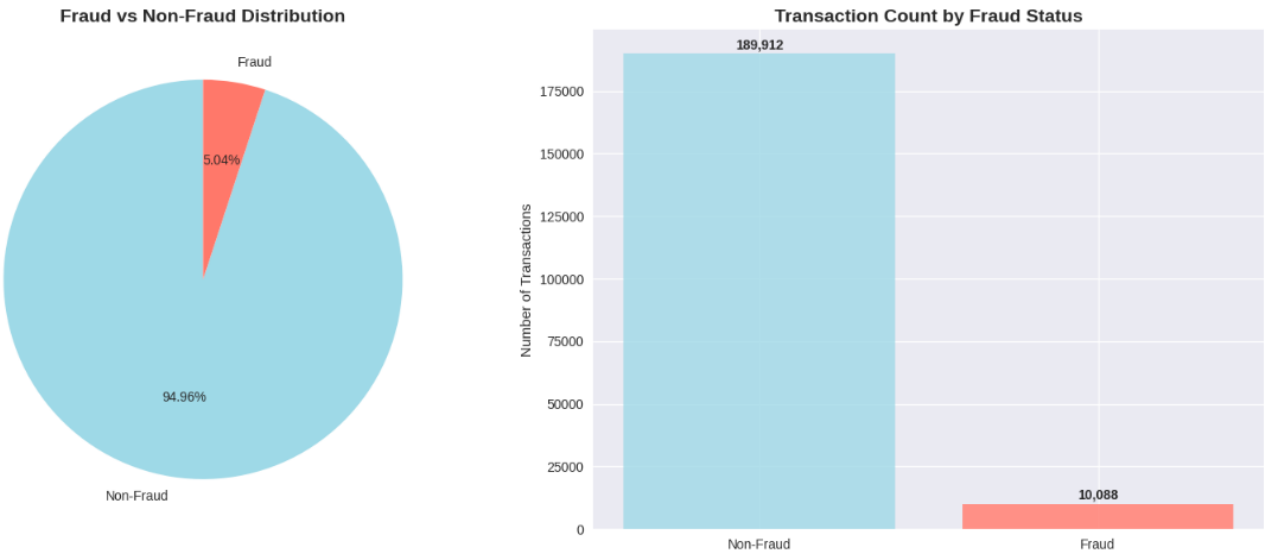

Рис.2.1. Аналіз збалансованості даних
*джерело розроблено автором

Частота шахрайських операцій на рівні 5,04% є досить високою порівняно з реальними банківськими показниками, що типово коливаються в межах 0,1-1%. Така особливість може свідчити про штучне збагачення набору даних шахрайськими прикладами для забезпечення ефективності навчання моделей машинного навчання. Водночас, це створює сприятливі умови для розробки та валідації алгоритмів виявлення без необхідності застосування складних методів балансування класів.

Дослідження кількісних змінних розпочинається з аналізу основних статистичних показників (див.дод. В) для виявлення відмінностей між шахрайськими та легітимними операціями. Ключові числові змінні включають вік клієнта, суму трансакції та баланс рахунку на момент операції. Аналіз вікової структури клієнтів показує мінімальні відмінності між групами. Середній вік клієнтів, що здійснюють шахрайські операції, становить 43,91 роки, тоді як для легітимних операцій - 44,02 роки. Різниця у 0,11 року (0,2%) є статистично незначущою, що свідчить про відсутність явної вікової дискримінації у шахрайських схемах. Стандартне відхилення в обох групах знаходиться на рівні 15,3 року, підтверджуючи однорідність вікового розподілу (див табл. 2.1).

Рис. 2.2. Аналіз вікової структури
*джерело розроблено автором

Таблиця 2.1

Аналіз структури за віком

Is_Fraud	Кількість	Середнє	Відхилення	min	25%	50%	75%	max
0	189912.0	44.020457	15.29005	18.0	31.0	44.0	57.0	70.0
1	10088.0	43.914453	15.26515	18.0	31.0	44.0	57.0	70.0

Суми транзакцій (див. табл. 2.2) демонструють схожі характеристики між групами. Середня сума шахрайської операції складає 49 277,93 грошових одиниць, що на 0,6% менше за середню суму легітимної операції (49 551,83). Медіанні значення також є близькими: 49 162,94 для шахрайських операцій проти 49 517,42 для звичайних. Така схожість розподілів свідчить про те, що шахраї не концентруються на операціях специфічного розміру, а намагаються імітувати типову фінансову поведінку

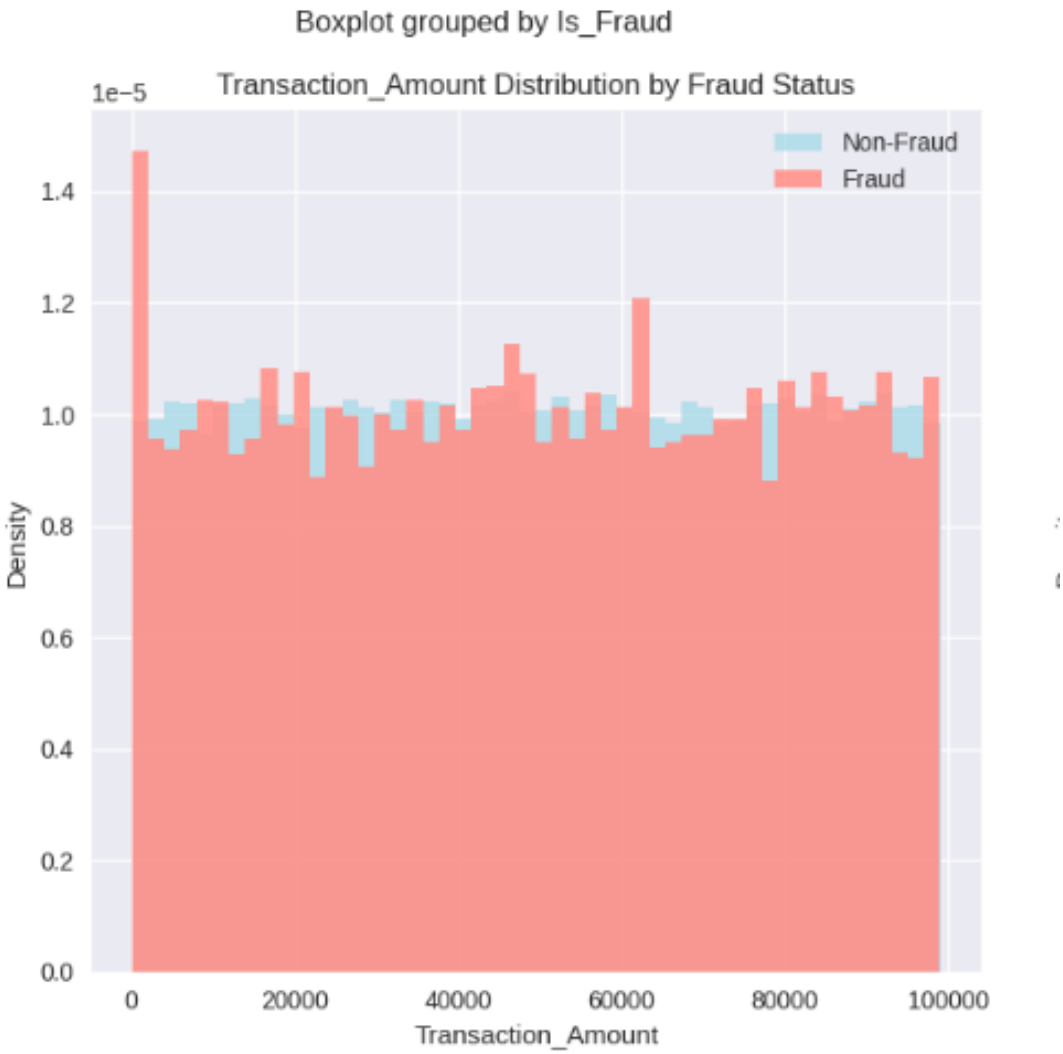

Рис. 2.3. Аналіз структури за сумами транзакцій
*джерело розроблено автором

Таблиця 2.2

Аналіз структури за сумою транзакцій

Is_Fraud	Кількість	Середнє	Відхилення	min	25%	50%	75%	max
0	189912.0	49551.831380	28541.881772	10.29	24877.6850	49517.420	74326.8725	98999.98
1	10088.0	49277.925242	28739.518843	10.41	24420.0425	49162.945	74068.7175	98999.02

Баланси рахунків (див. табл. 2.3) показують найменші відмінності між групами. Середній баланс при шахрайських операціях становить 52 452,89, що практично ідентично показнику для легітимних операцій (52 437,20). Різниця менше 0,1% підтверджує, що рівень достатку клієнтів не є визначальним фактором для шахрайських атак.

Таблиця 2.3

Аналіз структури за балансами рахунків

Is_Fraud	Кількість	Середнє	Відхилення	min	25%	50%	75%	max
0	189912.0	52437.197166	27404.829445	5000.82	28723.8875	52373.715	76148.8750	99998.68
1	10088.0	52452.891412	27300.467424	5003.42	29122.9875	52344.725	76116.0325	99999.95

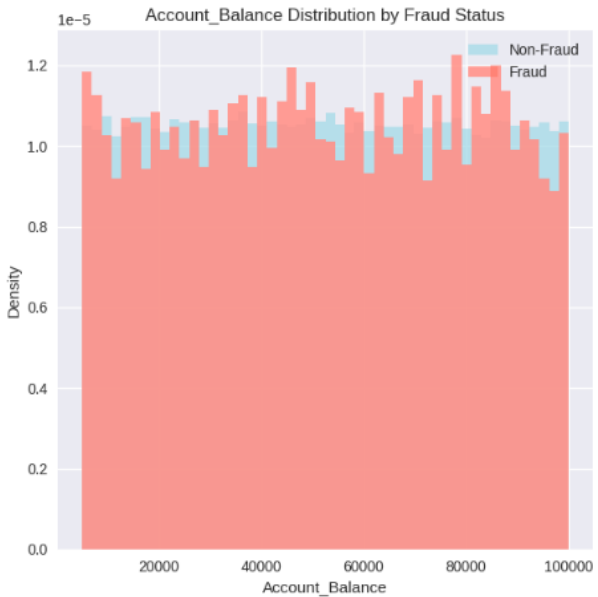

Рис. 2.4. Аналіз структури за балансами рахунків
*джерело розроблено автором

Аналіз категоріальних змінних дозволяє виявити специфічні сегменти та характеристики, що асоціюються з підвищеним ризиком шахрайських операцій. Кожна категоріальна змінна аналізується через призму частоти шахрайських операцій у відповідних сегментах.

Гендерний аналіз (див. табл. 2.4) показує незначну перевагу чоловіків у схильності до шахрайських операцій. Серед чоловіків частота шахрайств становить 5,06%, тоді як серед жінок - 5,03%. Абсолютна різниця у 0,03 відсоткових пункти є мінімальною та може пояснюватися випадковими відхиленнями в даних.

Таблиця 2.4

Гендерний аналіз

Стать	Кількість	Шахрайство	% шахрайства
Чоловік	100452	5081	0.0506
Жінка	99548	5007	0.0503

Аналіз типів рахунків (див. табл. 2.5) виявляє більш суттєві відмінності. Бізнес-рахунки демонструють найвищий рівень шахрайських операцій (5,17%), що перевищує показники ощадних рахунків (5,03%) та поточних рахунків (4,94%). Така закономірність може пояснюватися більшими сумами операцій та складнішими фінансовими схемами, притаманними корпоративному банкінгу.

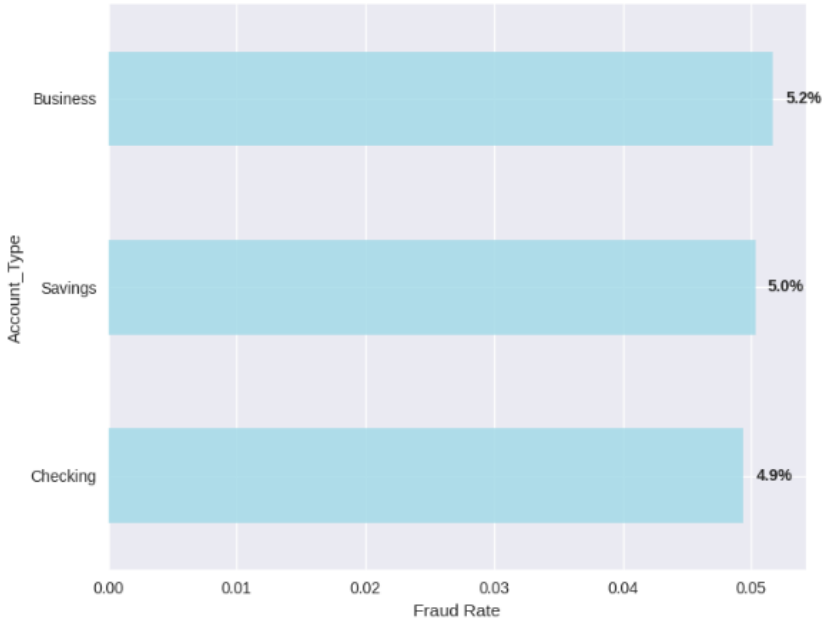

Рис. 2.5. Аналіз типів рахунків
*джерело розроблено автором

Таблиця 2.5

Аналіз типів рахунків

Рахунок	Кількість	Шахрайство	% шахрайства
Бізнес	66483	3436	0.0517
Ощадний	66593	3349	0.0503
Розрахунковий	66924	3303	0.0494

Типи транзакцій також демонструють варіативність у рівнях ризику (див. Табл. 2.6). Переказні операції характеризуються найвищою частотою шахрайств (5,19%), що може бути пов'язано з можливістю швидкого переміщення коштів між рахунками. Кредитні операції (5,10%) та дебетові транзакції (5,08%) показують середні рівні ризику, тоді як оплата рахунків та зняття коштів мають дещо нижчі показники (близько 4,93%).

Таблиця 2.6

Аналіз транзакцій за типом

Транзакція	Кількість	Шахрайство	% шахрайства
Переказ	39953	2073	0.0519
Кредит	40180	2048	0.0510
Дебет	40050	2033	0.0508
Оплата по рахунку	40040	1973	0.0493
Зняття	39777	1961	0.0493

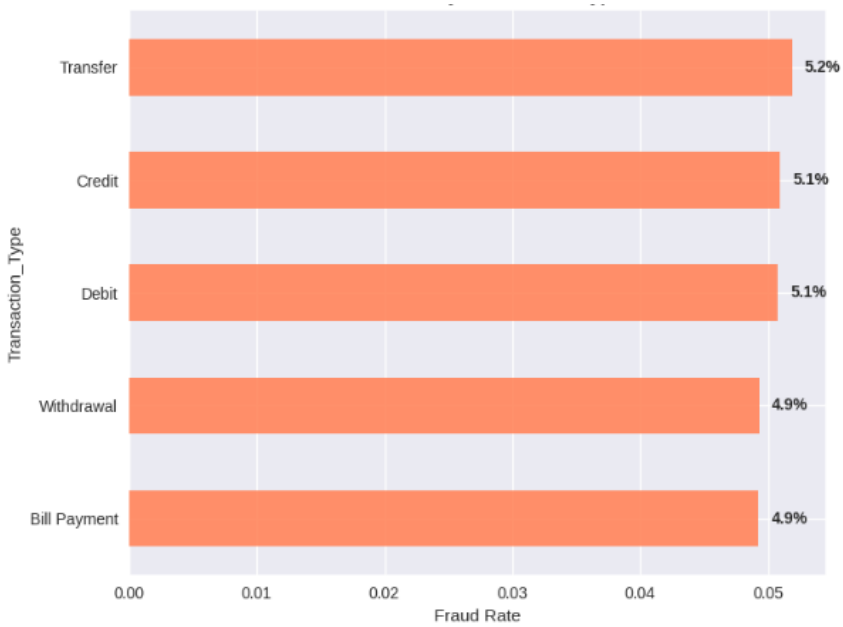

Рис. 2.6. Аналіз транзакцій за рівнем ризику
*джерело розроблено автором

Категорії спрямування витрат виявляють цікаві закономірності споживчої поведінки шахраїв. Найвищі рівні ризику спостерігаються у сфері одягу (5,20%) та продуктів харчування (5,19%). Ресторани, електроніка та сфера охорони здоров'я показують середні рівні ризику (5,0-5,04%), тоді як розважальний сектор демонструє найнижчу частоту шахрайства (4,82%).

Таблиця 2.7

Аналіз транзакцій за спрямуванням витрат

Спрямування	Кількість	Шахрайство	% шахрайства
Одяг	33340	1734	0.0520
Харчові продукти	33187	1722	0.0519
Ресторани	33525	1688	0.0504
Електроніка	33409	1681	0.0503
Здоров’я	33118	1651	0.0499
Розваги	33421	1612	0.0482

Аналіз технічних характеристик (див. табл. 2.8) проведення операцій розкриває важливі аспекти методології шахрайських атак. Пристрої проведення операцій демонструють значну варіативність у рівнях ризику, що може вказувати на технологічні вразливості окремих каналів.

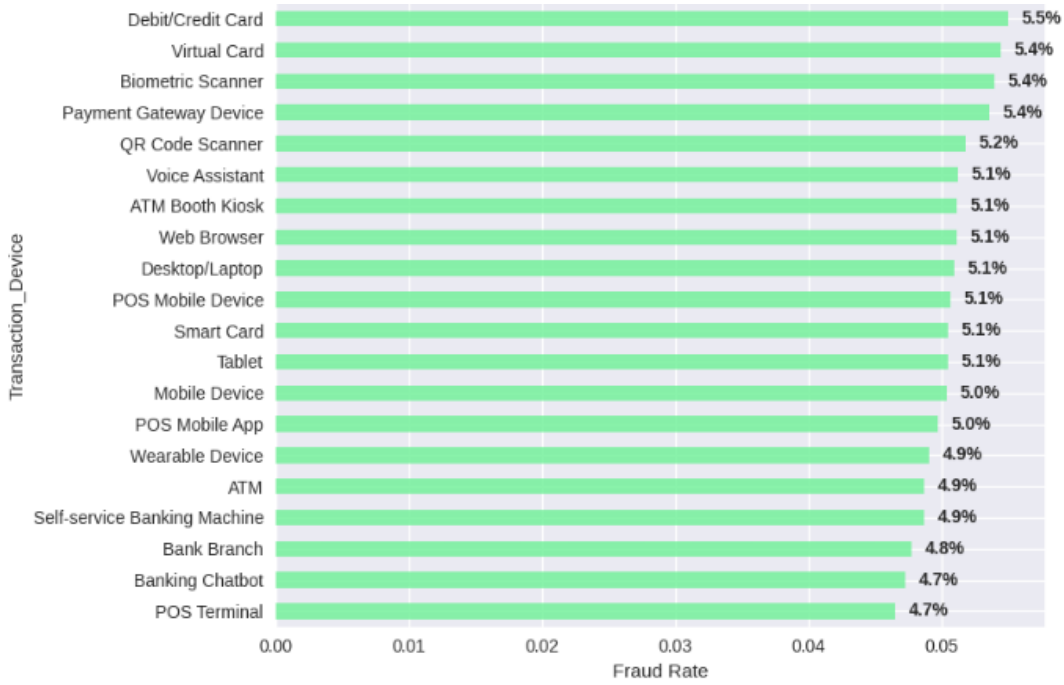

Рис. 2.7. Аналіз транзакцій за пристроєм платежу
*джерело розроблено автором

Дебетові та кредитні картки показують найвищі рівні шахрайських операцій (5,50%), що може бути пов'язано з можливістю клонування карток та несанкціонованого використання. Віртуальні картки (5,45%) та біометричні сканери (5,39%) також демонструють підвищені рівні ризику, що може свідчити про складнощі у захисті цих технологій. Банкомати та самообслуговування показують середні рівні ризику (близько 4,87-5,12%), тоді як операції через банківські відділення характеризуються найнижчими показниками шахрайства(4,77%). Така закономірність підтверджує гіпотезу про те, що особиста присутність та верифікація значно знижують ризики несанкціонованих операцій.

Типи пристроїв демонструють відносно рівномірний розподіл ризиків. Настільні комп'ютери показують найвищий рівень шахрайства (5,10%), що може бути пов'язано з вразливостями веб-браузерів та можливістю віддаленого доступу. POS-термінали, банкомати та мобільні пристрої показують близькі рівні ризику (5,0-5,05%)

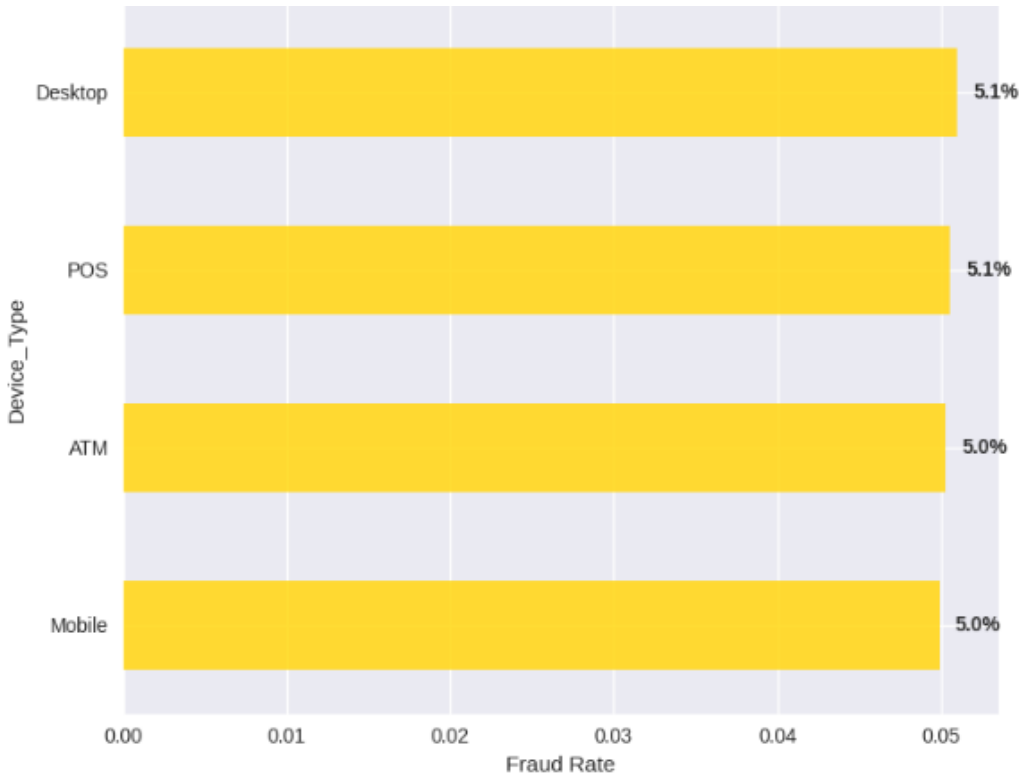

Рис. 2.8. Аналіз за типом пристрою
*джерело розроблено автором

Таблиця 2.8

Аналіз транзакцій за пристроєм платежу

Пристрій транзакції	**Загальна кількість транзакцій**	**Кількість шахрайств**	**Рівень шахрайства**
Дебетова/Кредитна карта	8273	455	0.0550
Віртуальна карта	8059	439	0.0545
Біометричний сканер	7952	429	0.0539
Пристрій платіжного шлюзу	7874	422	0.0536
Сканер QR-коду	7938	411	0.0518
Голосовий асистент	8039	412	0.0513
Кіоск банкомату	21149	1082	0.0512
Веб-браузер	7981	408	0.0511
Комп'ютер/Ноутбук	8057	411	0.0510
Мобільний POS-пристрій	8006	406	0.0507
Планшет	8059	407	0.0505
Смарт-карта	8133	411	0.0505
Мобільний пристрій	7879	397	0.0504
Мобільний POS-додаток	7868	391	0.0497
Носимий пристрій	8128	399	0.0491
Банкомат	21200	1033	0.0487
Самообслуговування банківський апарат	21707	1057	0.0487
Відділення банку	7855	375	0.0477
Банківський чат-бот	7995	378	0.0473
POS-термінал	7848	365	0.0465

Таблиця 2.9

Аналіз транзакцій за типом пристрою

Тип пристрою	**Загальна кількість транзакцій**	**Кількість шахрайств**	**Рівень шахрайства**
Настільний ПК	49872	2544	0.0510
POS	50111	2533	0.0505
Банкомат	50055	2518	0.0503
Мобільний	49962	2493	0.0499

Таблиця 2.10

Аналіз транзакцій за днем

День тижня	Загальна кількість транзакцій	Кількість шахрайств	Рівень шахрайства
Понеділок	26557	1374	0.0517
Середа	33340	1723	0.0517
П’ятниця	26492	1347	0.0508
Неділя	26711	1358	0.0508
Четвер	33581	1706	0.0508
Вівторок	26435	1308	0.0495
Субота	26884	1272	0.0473

Часовий аналіз шахрайських операцій (див дод. Е) дозволяє виявити циклічні патерни та періоди підвищеної активності шахраїв. Дані охоплюють операції за січень 2025 року, що забезпечує можливість аналізу денної та тижневої динаміки.

Аналіз за днями тижня виявляє цікаві закономірності (див. табл. 2.10). Понеділок та середа демонструють найвищі рівні шахрайських операцій (5,17%), що може бути пов'язано з початком робочого тижня та активізацією бізнес-операцій. П'ятниця та неділя показують середні рівні ризику (5,08%), тоді як вівторок характеризується нижчою частотою шахрайств (4,95%).

Субота виділяється як день з найнижчим рівнем шахрайських операцій (4,73%), що може пояснюватися зниженою активністю бізнес-транзакцій та меншою кількістю операцій загалом. Така закономірність може свідчити про те, що шахраї переважно імітують звичайну ділову активність, уникаючи проведення операцій у нетипові для бізнесу періоди.

Місячний аналіз обмежений одним місяцем (січень), що не дозволяє виявити сезонні закономірності. Однак рівномірність розподілу протягом місяця (5,04%) свідчить про постійність шахрайської активності без значних коливань у межах календарного періоду.

Кореляційний аналіз (див дод. Є) числових змінних не виявив сильних лінійних зв'язків з цільовою змінною шахрайства. Це важливе спостереження

свідчить про те, що шахрайські патерни мають складний, нелінійний характер, що вимагає застосування розширених методів машинного навчання для їх виявлення.

Відсутність сильних кореляцій між числовими змінними також вказує на те, що шахраї успішно імітують статистичні характеристики легітимних операцій. Це підтверджує гіпотезу про високу складність сучасних шахрайських схем та необхідність комплексного підходу до їх виявлення.

Слабкі кореляції між основними фінансовими показниками (сума операції, баланс рахунку) свідчать про відсутність мультиколінеарності, що є позитивною характеристикою для подальшого моделювання. Така структура даних дозволяє використовувати всі доступні змінні у моделях машинного навчання без ризику перенавчання через надлишковість інформації.

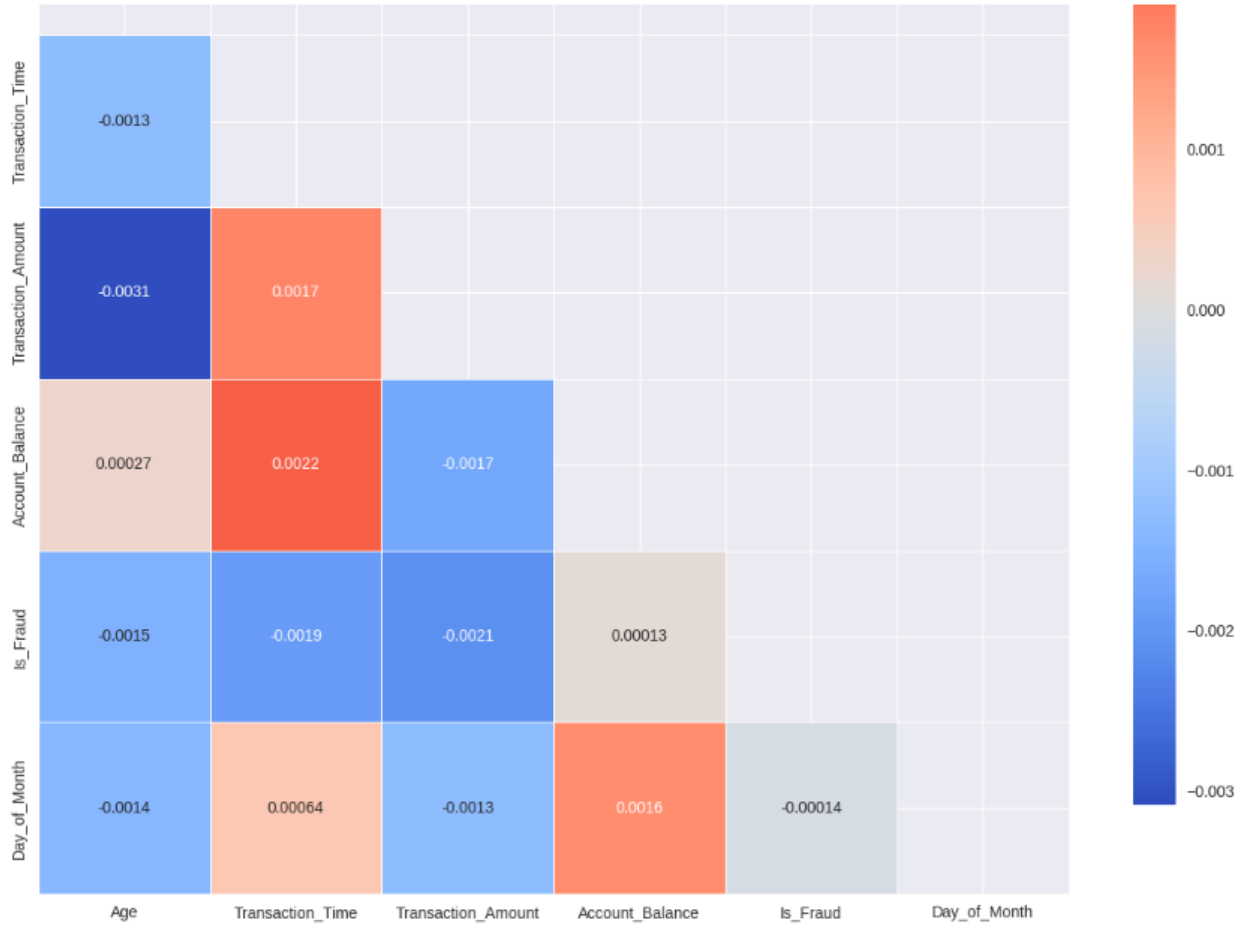

Рис. 2.9. Кореляційний аналіз
*джерело розроблено автором

Детальний аналіз розподілу сум операцій за категоріями розкриває специфічні закономірності шахрайської поведінки (див. табл. 2.11). Категоризація операцій за розміром суми виявляє неочікувані патерни ризику.

Таблиця 2.11

Аналіз транзакцій за типом пристрою

Категорія суми	**Всього транзакцій**	**Кількість шахрайств**	**Рівень шахрайства**
Дуже низька ($0-$100)	167	17	0.1018
Низька ($100-$500)	838	86	0.1026
Середня ($500-$1K)	991	86	0.0868
Висока ($1K-$5K)	8045	385	0.0479
Дуже висока ($5K+)	189959	9514	0.0501

Дуже малі операції (0-100 доларів) демонструють найвищі відносні рівні шахрайства (10,18%), що може свідчити про тестування викрадених реквізитів або спроби уникнення систем моніторингу через малі суми. Низькі суми (100-500 доларів) також показують підвищені рівні ризику (10,26%).

Середні операції (500-1000 доларів) характеризуються дещо нижчим рівнем шахрайства (8,68%), тоді як високі (1000-5000 доларів) та дуже високі суми (понад 5000 доларів) показують найнижчі відносні рівні ризику (4,79% та 5,01% відповідно). Така інверсія може пояснюватися посиленим моніторингом великих операцій з боку банківських систем.

Загальна сума шахрайських операцій становить 497 115 709,84 грошових одиниць, що при середній сумі однієї операції 49 277,93 свідчить про значний фінансовий вплив шахрайства на банківську систему

Комбінування декількох факторів ризику дозволяє створити більш нюансовану картину шахрайських патернів. Аналіз показує, що ризик-скор, побудований на основі множинних факторів, демонструє відносно рівномірний розподіл ризиків між різними категоріями.

Таблиця 2.12

Багатофакторна оцінка ризику

Оцінка_ризику	Усього_транзакцій	Кількість_шахрайств	Рівень_шахрайства
0	79033	4004	0.0507
1	90402	4528	0.0501
2	28369	1448	0.0510
3	2196	108	0.0492

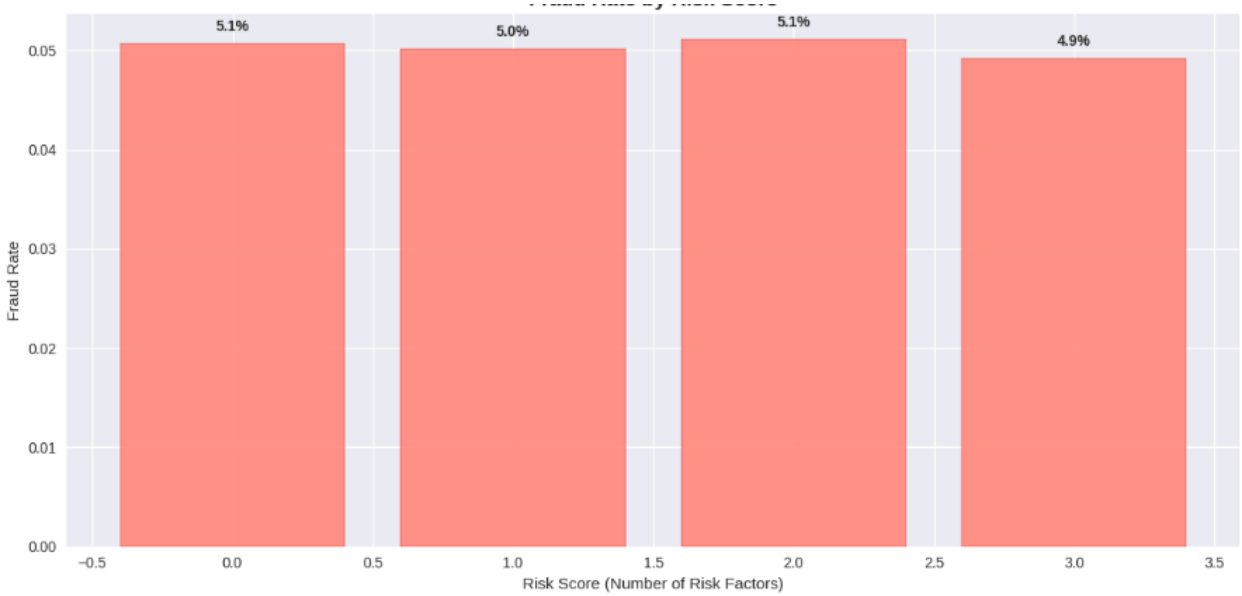

Рис. 2.10. Багатофакторна оцінка ризиків
*джерело розроблено автором

Комбінування декількох факторів ризику дозволяє створити більш нюансовану картину шахрайських патернів. Аналіз показує, що ризик-скор, побудований на основі множинних факторів, демонструє відносно рівномірний розподіл ризиків між різними категоріями.

Операції з оцінкою 0 показують частоту шахрайств 5,07%, тоді як операції з максимальним скором 3 демонструють дещо нижчий рівень (4,92%). Така закономірність може свідчити про складність виявлення шахрайських операцій через традиційні скорингові методи та необхідність застосування більш складних алгоритмів.

Розподіл операцій за оцінками ризику показує, що більшість транзакцій (90 402) мають оцінку 1, тоді як найменша кількість (2 196) характеризується максимальним скором 3. Така нерівномірність може свідчити про те, що

комбінація множинних факторів ризику є відносно рідкісною, що ускладнює використання простих правилових систем для виявлення шахрайств.

Результати аналізу мають прямі наслідки для банківського бізнесу та стратегії управління ризиками. Загальна сума шахрайських операцій майже у 500 мільйонів грошових одиниць при аналізі лише 200 000 транзакцій свідчить про масштабність проблеми у банківському секторі.

Середня сума шахрайської операції (49 277,93) є лише на 0,6% меншою за середню легітимну операцію, що підкреслює складність виявлення через традиційні порогові методи. Шахраї демонструють високий рівень адаптації, імітуючи типову клієнтську поведінку за основними фінансовими показниками.

Ідентифікація високо ризикових сегментів (бізнес-рахунки, переказні операції, операції з дебетовими картками) створює можливості для цільового моніторингу та превентивних заходів. Концентрація уваги на цих сегментах може підвищити ефективність систем виявлення при оптимізації ресурсних витрат.

2.3. Підготовка даних для моделювання

2.3.1 Вирішення проблеми незбалансованості класів

Аналіз первинного розподілу цільової змінної виявив значну диспропорцію між класами: кількість легальних транзакцій становила 189 912 записів, тоді як шахрайські транзакції представлені лише 10 088 спостереженнями. Така незбалансованість створює фундаментальну проблему для алгоритмів машинного навчання, оскільки модель може досягти високої загальної точності просто шляхом класифікації всіх транзакцій як легальних, при цьому повністю ігноруючи міноритарний клас шахрайських операцій.

Для вирішення цієї проблеми було обрано стратегію андерсемплінгу легальних транзакцій замість оверсемплінгу шахрайських. Це рішення базується на кількох критичних міркуваннях. Насамперед, технічні обмеження обчислювальних ресурсів не дозволяють ефективно обробляти повний масив у 200 тисяч записів протягом розумного часового інтервалу. Крім того,

андерсемплінг зменшує ризик перенавчання моделі на надлишкових зразках мажоритарного класу та забезпечує більш збалансовані умови для навчання алгоритмів.

Процес андерсемплінгу (див дод. С) здійснюється через випадковий відбір підмножини легальних транзакцій, що за розміром відповідає кількості шахрайських операцій. Використання фіксованого значення генератора псевдовипадкових чисел забезпечує повну відтворюваність експерименту. Після відбору та об'єднання даних виконувалося їх перемішування для усунення потенційних патернів упорядкування, що могли б вплинути на процес навчання моделей.

Результатом стало створення збалансованого датасету з 20 176 записами, де кожен клас представлений рівно 10 088 спостереженнями. Такий підхід забезпечує рівні можливості для алгоритмів навчитися розпізнавати характерні риси як легальних, так і шахрайських транзакцій.

2.3.2. Створення часових ознак для аналізу темпоральних патернів

Часова компонента транзакційних даних містить критично важливу інформацію для виявлення аномальної активності. Шахрайські операції часто характеризуються специфічними часовими патернами, які відрізняються від звичайної поведінки легітимних користувачів.

З оригінального поля дати та часу транзакції створено кілька похідних ознак (див. дод. К). Година доби транзакції створює можливість ідентифікувати операції, що відбуваються в нетипові для конкретного користувача або загалом підозрілі часові проміжки. Багато шахрайських транзакцій відбуваються в ранкові або пізні вечірні години, коли банківський моніторинг може бути менш активним.

День тижня та бінарний індикатор вихідних днів дозволяють виявити аномалії в часових патернах активності. Легітимні користувачі зазвичай демонструють різні патерни поведінки в робочі та вихідні дні, тому відхилення від цих патернів може сигналізувати про потенційне шахрайство.

2.3.3. Розрахунок поведінкових метрик клієнтів

Поведінкові характеристики користувачів представляють особливу цінність для систем виявлення шахрайства, оскільки дозволяють ідентифікувати відхилення від індивідуальних патернів активності кожного клієнта (див дод Л).

Часовий інтервал між послідовними транзакціями одного клієнта розраховується через різницю в секундах між поточною та попередньою операцією. Аномально короткі інтервали можуть вказувати на автоматизовані атаки або спроби швидкого вичерпання коштів. Навпаки, незвично довгі перерви в активності з подальшими великими транзакціями також можуть свідчити про компрометацію облікового запису.

Коефіцієнт відношення суми транзакції до залишку на рахунку створює нормалізований показник розміру операції відносно фінансових можливостей клієнта. Цей метрик особливо ефективний для виявлення спроб максимального списання коштів, характерних для багатьох типів шахрайства.

Бінарний індикатор великих транзакцій базується на статистичному аналізі розподілу сум операцій. Транзакції, що перевищують 95-й процентиль, отримують позначку як потенційно підозрілі, що дозволяє моделі сфокусуватися на аномально великих операціях.

2.3.4. Впровадження системи ковзаючих статистик

Для захоплення динамічних змін (див. дод. М) у поведінці клієнтів було реалізовано систему ковзаючих вікон, що аналізують статистичні характеристики останніх п'яти транзакцій кожного користувача. Вибір розміру вікна в п'ять операцій базується на балансі між чутливістю до коротко термінових змін та стабільністю статистичних оцінок.

Ковзаюче середнє суми транзакцій відображає типовий розмір операцій клієнта в найближчій ретроспективі. Різкі зміни цього показника можуть сигналізувати про зміну поведінкових патернів, потенційно пов'язану з несанкціонованим доступом до рахунку.

Ковзаюче стандартне відхилення характеризує варіативність сум транзакцій у межах вікна. Клієнти зазвичай демонструють відносно стабільні патерни витрат, тому значне зростання варіативності може вказувати на аномальну активність.

Частка великих транзакцій у межах ковзаючого вікна створює динамічний індикатор концентрації підозрілих операцій. Швидке зростання цього показника може свідчити про спробу максимізації збитків за короткий часовий період.

2.3.5. Агрегація активності за часовими періодами

Інтенсивність транзакційної активності протягом різних часових проміжків служить важливим індикатором для виявлення аномальної поведінки. Аналіз частоти операцій дозволяє ідентифікувати як аномальні сплески активності, так і підозрілі патерни регулярності (див. дод. Н).

Підрахунок кількості транзакцій клієнта протягом конкретного календарного дня виявляє денні аномалії активності. Різкі зростання денної активності, особливо в поєднанні з великими сумами операцій, можуть сигналізувати про спроби швидкого вичерпання коштів або тестування викрадених платіжних даних.

Аналогічно, кількість операцій у межах конкретної години доби для кожного клієнта дозволяє виявити нетипові часові патерни. Концентрація множинних транзакцій у незвичайні для конкретного користувача години може вказувати на автоматизовану шахрайську активність.

2.3.6. Створення складних взаємодіючих ознак

Лінійна комбінація та множення базових ознак створює більш складні індикатори, здатні захопити нелінійні взаємодії між різними аспектами транзакційної поведінки (див. дод. О). Такі композитні ознаки часто виявляються особливо ефективними для алгоритмів машинного навчання.

Добуток коефіцієнта відношення суми до залишку та індикатора великої транзакції підсилює сигнал про потенційне шахрайство шляхом комбінування інформації про відносний та абсолютний розміри операції. Цей показник досягає

максимальних значень для транзакцій, що є одночасно великими в абсолютному вимірі та значними відносно балансу рахунку.

Взаємодія між годиною доби та індикатором вихідного дня створює композитний часовий маркер, що виділяє транзакції в потенційно підозрілі періоди. Операції в пізні години вихідних днів отримують підвищені значення цього показника.

Відхилення поточної суми транзакції від індивідуального ковзаючого середнього клієнта вимірює ступінь аномальності операції відносно недавньої поведінки користувача. Великі позитивні відхилення вказують на незвично великі для даного клієнта транзакції.

Композитна категоріальна ознака, що об'єднує інформацію про банківське відділення, категорію торговця та тип транзакції, створює деталізований контекстний маркер. Такі триплети дозволяють моделі навчитися специфічним патернам шахрайства, характерним для конкретних комбінацій місць, типів торговців та видів операцій.

2.3.7. Структуризація ознакового простору

Після створення всіх похідних ознак було проведено структуризацію ознакового простору шляхом розділення на чисельні та категоріальні змінні (див. дод. П). Чисельні ознаки включають вікові характеристики клієнтів, фінансові показники транзакцій, часові метрики та всі розраховані поведінкові індикатори. До категоріальних змінних віднесено демографічні характеристики, географічні локації, технічні параметри транзакцій та створену композитну ознаку.

Така структуризація є критично важливою для подальших етапів препроцесингу, оскільки чисельні та категоріальні ознаки потребують різних підходів до нормалізації та кодування. Чисельні змінні підлягатимуть стандартизації для забезпечення однакових масштабів, тоді як категоріальні ознаки потребуватимуть спеціальних методів кодування для перетворення в числовий формат, придатний для алгоритмів машинного навчання.

2.3.8. Target Encoding для категоріальних змінних

Категоріальні змінні потребують спеціального підходу для перетворення в числовий формат, придатний для алгоритмів машинного навчання. Серед різноманітних методів кодування було обрано Target Encoding як найбільш ефективний для задач бінарної класифікації з великою кількістю категорій (див. дод. Р).

Target Encoding замінює кожну категорію середнім значенням цільової змінної для спостережень цієї категорії. Цей підхід особливо ефективний для виявлення шахрайства, оскільки автоматично виявляє категорії з підвищеним ризиком та надає їм відповідні числові значення. Наприклад, якщо певне банківське відділення або категорія торговця демонструє вищий рівень шахрайських транзакцій, Target Encoding автоматично присвоїть їм вищі значення.

Процедура кодування застосовувалася до всіх категоріальних змінних послідовно, включаючи демографічні характеристики клієнтів, географічні локації, банківські та торгові категорії, а також технічні параметри транзакцій. Використання цільової змінної для кодування створює потужний зв'язок між категоріальними ознаками та ймовірністю шахрайства.

2.4. Розробка моделей виявлення шахрайства

Для забезпечення об'єктивної оцінки ефективності моделей було застосовано стратифіковане розділення даних на навчальну та тестову вибірки у пропорції 80:20. Стратифікація гарантує збереження первинного співвідношення класів у обох підмножинах, що є критично важливим для збалансованих даних.

Тестова вибірка складає 20 відсотків від загального обсягу даних, що забезпечує достатню статистичну потужність для надійної оцінки моделей. Фіксоване значення генератора псевдовипадкових чисел забезпечує повну відтворюваність результатів та можливість порівняння різних підходів на ідентичних даних (див. дод. С).

Навчальна вибірка використовується для тренування всіх алгоритмів, тоді як тестова залишається недоторканою до етапу фінальної оцінки. Такий підхід

запобігає витоку інформації та забезпечує чесну оцінку узагальнюючої здатності моделей.

Числові ознаки характеризуються різними масштабами та одиницями вимірювання, що може негативно впливати на ефективність деяких алгоритмів машинного навчання. Алгоритми, основані на обчисленні відстаней, такі як k-найближчих сусідів та нейронні мережі, особливо чутливі до масштабів ознак.

Стандартизація перетворює кожну числову ознаку таким чином, щоб вона мала нульове середнє значення та одиничну дисперсію. Цей процес забезпечує рівноправне трактування всіх ознак незалежно від їх первинних масштабів та дозволяє алгоритмам ефективно використовувати інформативність кожної змінної.

Критично важливим є те, що параметри стандартизації розраховуються виключно на основі навчальної вибірки, а потім застосовуються до тестових даних. Такий підхід запобігає витоку інформації з тестової вибірки та забезпечує реалістичну оцінку ефективності моделей в умовах реального застосування.

LightGBM представляє сучасну реалізацію алгоритму градієнтного бустингу, оптимізовану для роботи з великими обсягами даних та високою швидкістю навчання. Цей алгоритм послідовно будує ансамбль слабких учнів, де кожна наступна модель намагається виправити помилки попередніх.

Налаштування гіперпараметрів LightGBM сфокусовано на досягненні оптимального балансу між точністю та узагальнюючою здатністю. Помірна швидкість навчання у 0.05 забезпечує стабільну конвергенцію, тоді як 64 листя на дерево дозволяють моделі захоплювати складні нелінійні залежності. Максимальна глибина дерев обмежена вісьмома рівнями для запобігання перенавчанню.

Регуляризаційні параметри L1 та L2 встановлено на рівні одиниці для контролю складності моделі. Випадкова підвибірка ознак та об'єктів зі співвідношеннями 0.8 та 0.8 відповідно створює додатковий регуляризаційний ефект та підвищує стійкість моделі до шуму в даних.

Механізм раннього зупинення моніторить якість моделі на валідаційній вибірці та припиняє навчання при відсутності покращень протягом 50 ітерацій. Це запобігає перенавчанню та автоматично визначає оптимальну кількість дерев в ансамблі.

Random Forest реалізує концепцію ансамблевого навчання через комбінування множини незалежних дерев рішень. Кожне дерево навчається на випадковій підвибірці даних та випадковій підмножині ознак, що створює різноманітність у ансамблі та підвищує узагальнюючу здатність.

Архітектура лісу включає 300 дерев, що забезпечує достатню різноманітність для стабільних прогнозів без надмірного збільшення обчислювальної складності. Максимальна глибина дерев обмежена десятьма рівнями для балансу між складністю моделі та здатністю до узагальнення.

Автоматичне збалансування класів через параметр class_weight дозволяє алгоритму надавати більшої ваги мінорітарному класу шахрайських транзакцій. Це особливо важливо для задач з несиметричними втратами, де пропуск шахрайської транзакції має значно вищу вартість порівняно з помилковою тривогою.

Алгоритм k-найближчих сусідів реалізує принципово інший підхід до класифікації, базуючись на припущенні про локальну схожість об'єктів в ознаковому просторі. Замість побудови глобальної моделі, KNN приймає рішення на основі аналізу найближчих сусідів кожного нового об'єкта.

Значення k=15 обрано як компроміс між чутливістю до локальної структури даних та стійкістю до шуму. Менші значення k роблять алгоритм більш чутливим до аномалій у даних, тоді як більші значення згладжують локальні особливості та можуть призвести до втрати важливої інформації.

Використання стандартизованих даних є критично важливим для KNN, оскільки алгоритм базується на обчисленні евклідових відстаней між об'єктами. Без нормалізації ознаки з більшими масштабами домінуватимуть у розрахунках відстаней, що може призвести до неадекватних результатів.

Багатошаровий перцептрон представляє потужний інструмент для виявлення складних нелінійних залежностей у даних через ієрархічну систему штучних нейронів. Архітектура мережі включає два прихованих шари з 128 та 64 нейронами відповідно, що забезпечує достатню експресивність для складних патернів без надмірного ускладнення.

Функція активації ReLU обрана через свої переваги в контексті глибокого навчання: простоту обчислень, відсутність проблеми згасаючих градієнтів та ефективну розрідженість активацій. Оптимізатор Adam забезпечує адаптивне налаштування швидкості навчання та ефективну конвергенцію навіть на складних поверхнях втрат.

Розмір міні-батчу у 256 зразків створює баланс між стабільністю градієнтних оновлень та ефективністю обчислень. Обмеження максимальної кількості епох до 50 запобігає перенавчанню при використанні відносно невеликого датасету.

Стандартний поріг класифікації у 0.5 часто не є оптимальним для задач з несиметричними втратами, характерними для виявлення шахрайства. Для кожної моделі проводилася систематична оптимізація порогу через перебір значень від 0.1 до 0.9 з кроком приблизно 0.01.

Критерієм оптимізації обрано F1-оцінки, що представляє гармонійне середнє між точністю та повнотою. Цей метрик особливо підходить для збалансованих даних і забезпечує компромісний баланс між мінімізацією помилкових тривог та максимізацією виявлення справжніх випадків шахрайства.

Процедура оптимізації виконується окремо для кожної моделі, оскільки різні алгоритми можуть мати різні оптимальні пороги класифікації. Це забезпечує справедливе порівняння моделей при їх найкращих налаштуваннях.

2.5. Оцінка ефективності моделей

Експериментальне порівняння чотирьох (див. дод. Т) алгоритмів машинного навчання для виявлення шахрайських транзакцій продемонструвало відносно схожу ефективність провідних підходів з деякими важливими відмінностями в характеристиках продуктивності. Найвищу дискримінаційну здатність за

метрикою ROC-AUC показав алгоритм LightGBM з результатом 0.7593, за ним слідує Random Forest з показником 0.7559, що вказує на майже еквівалентну якість ранжування цих двох ансамблевих методів.

Neural Network продемонстрував помірну ефективність з ROC-AUC 0.7281, тоді як k-найближчі сусіди показали найнижчий результат 0.6836. Такий розподіл результатів свідчить про те, що ансамблеві методи, які комбінують множину слабких учнів, виявляються більш ефективними для виявлення складних патернів шахрайства порівняно з одиничними моделями.

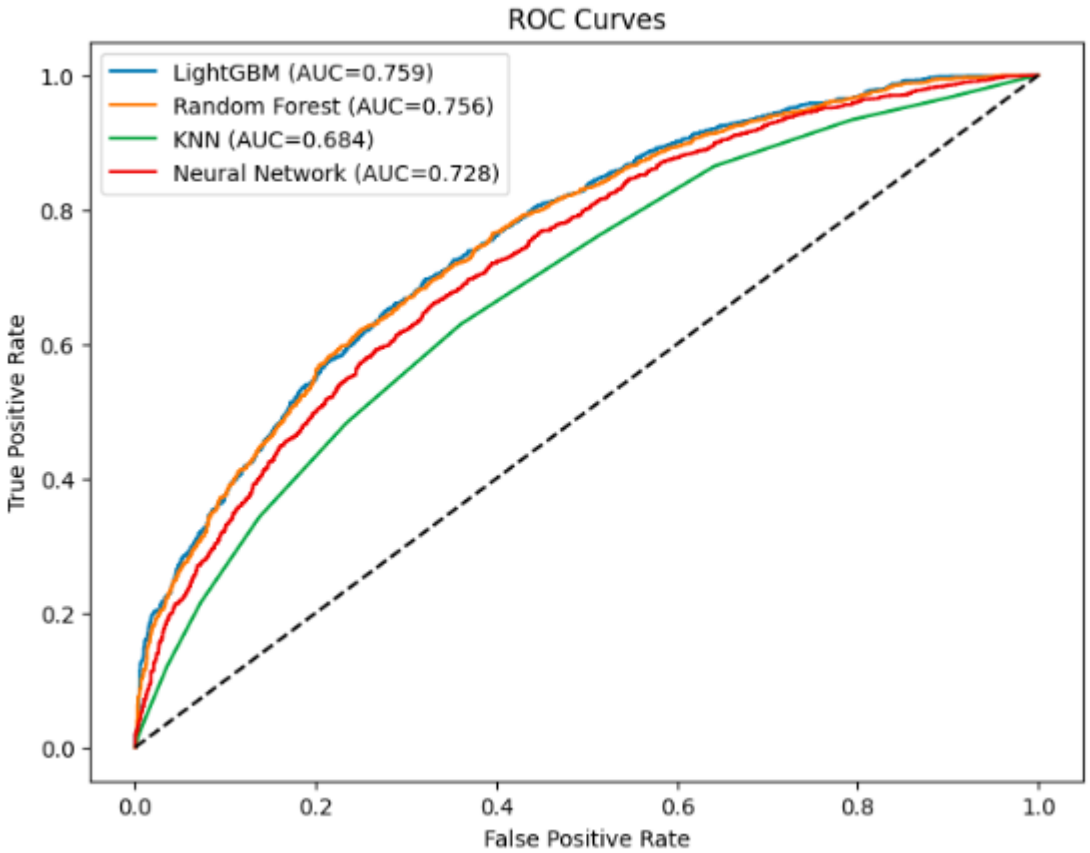

Рис. 2.11. ROC-криві побудованих моделей
*джерело розроблено автором

Варто зазначити, що всі моделі потребували значної корекції порогу класифікації від стандартного значення 0.5. Оптимальні пороги варіювалися від 0.28 для нейронної мережі до 0.43 для LightGBM, що підкреслює важливість індивідуального налаштування кожної моделі для досягнення максимальної ефективності.

Алгоритм LightGBM продемонстрував найвищу загальну ефективність серед усіх досліджуваних підходів. Особливо вражаючим є той факт, що модель досягла оптимального результату вже на 13-й ітерації, що свідчить про швидку конвергенцію та ефективність механізму раннього зупинення. Показник AUC на навчальній вибірці склав 0.782461, тоді як на валідаційній - 0.759266, демонструючи помірне перенавчання та хорошу узагальнюючу здатність.

Оптимальний поріг класифікації для LightGBM склав 0.43, що є найвищим серед усіх моделей. Це свідчить про те, що модель генерує більш консервативні ймовірнісні оцінки та потребує вищого порогу для позитивної класифікації. При цьому пороге модель досягає F1-міри 0.72 для класу шахрайських транзакцій, що є найкращим результатом серед усіх алгоритмів.

Аналіз метрик по класах виявляє цікаву закономірність: модель демонструє високу повноту для виявлення шахрайства (0.91), але помірну точність (0.60). Для легальних транзакцій спостерігається зворотна ситуація - висока точність (0.81) при низькій повноті (0.39). Така характеристика може бути прийнятною для практичного застосування, оскільки система має тенденцію до виявлення більшості шахрайських операцій за рахунок деякої кількості помилкових тривог.

Random Forest продемонстрував майже ідентичну до LightGBM ефективність за ROC-AUC (0.7559), що підтверджує силу ансамблевих методів для даної задачі. Модель потребувала нижчого оптимального порогу 0.36, що вказує на дещо інший розподіл ймовірнісних оцінок порівняно з градієнтним бустингом.

Структура помилок Random Forest виявилася дуже схожою до LightGBM: висока повнота для шахрайських транзакцій (0.89) при помірній точності (0.60), та висока точність для легальних операцій (0.79) при низькій повноті (0.41). Ці результати підтверджують, що обидва ансамблеві методи виявили схожі патерни в даних та демонструють подібні характеристики класифікації.

Незначна різниця в ефективності між двома провідними моделями свідчить про стабільність результатів та надійність висновків. Вибір між LightGBM та Random Forest в практичному застосуванні може базуватися на додаткових

критеріях, таких як швидкість навчання, інтерпретованість або обчислювальні вимоги.

Neural Network показав помірні результати з ROC-AUC 0.7281, що є третім за ефективністю серед досліджуваних алгоритмів. Найнижчий оптимальний поріг 0.28 вказує на те, що мережа генерує більш агресивні ймовірнісні оцінки та схильна до більш ліберальної класифікації як шахрайських.

Структура помилок нейронної мережі продемонструвала схожі до інших моделей тенденції: високу повноту для шахрайства (0.87) при помірній точності (0.60), та помірну точність для легальних транзакцій (0.76) при низькій повноті (0.41). Однак загальна ефективність була нижчою порівняно з ансамблевими методами.

Відносно слабша ефективність нейронної мережі може пояснюватися кількома факторами. По-перше, розмір датасету може бути недостатнім для ефективного навчання глибоких архітектур. По-друге, табличні дані з переважно категоріальними ознаками не завжди найкраще підходять для нейронних мереж, які традиційно демонструють переваги на неструктурованих даних, таких як зображення або тексти.

Алгоритм k-найближчих сусідів продемонстрував найнижчу ефективність серед усіх досліджуваних моделей з ROC-AUC 0.6836. Оптимальний поріг 0.34 є другим найнижчим після нейронної мережі, що вказує на схильність до агресивної класифікації.

Метрики якості KNN виявили найгірші показники в усіх категоріях: найнижчу точність для легальних транзакцій (0.73), найнижчу точність для шахрайських операцій (0.57), та найнижчі значення F1-міри для обох класів. Повнота для шахрайських транзакцій склала 0.86, що є прийнятним, але поєднується з найвищим рівнем помилкових тривог.

Слабка ефективність KNN може пояснюватися природою задачі виявлення шахрайства. Шахрайські патерни можуть бути розпорошеними в ознаковому просторі та не формувати компактних кластерів, що ускладнює ефективне застосування методів, базованих на локальній схожості. Крім того, висока

розмірність ознакового простору може призводити до прояву "прокляття розмірності", що знижує ефективність KNN.

Аналіз результатів виявив кілька важливих закономірностей, характерних для всіх досліджуваних моделей. Найбільш помітною є систематично низька повнота для класу легальних транзакцій (від 0.36 до 0.41) при високій повноті для шахрайських операцій (від 0.86 до 0.91). Ця характеристика вказує на те, що всі моделі мають тенденцію до консервативної класифікації легальних транзакцій та агресивного виявлення потенційного шахрайства.

Такий патерн може бути результатом особливостей збалансованого датасету або характеристик створених ознак. Він також може відображати об'єктивну складність розмежування деяких типів легальних та шахрайських транзакцій, особливо в граничних випадках.

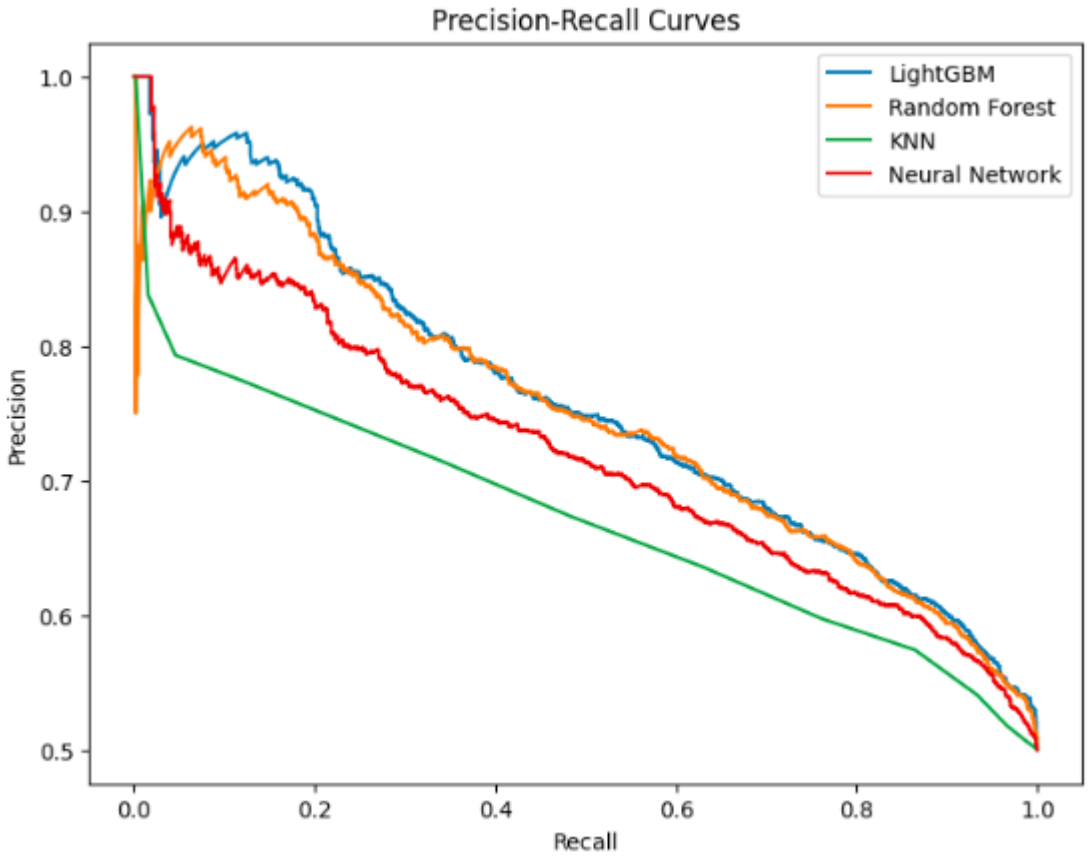

Рис. 2.11. Відношення Точність-Повнота
*джерело розроблено автором

Загальна точність моделей варіюється в вузькому діапазоні від 0.61 до 0.65, що свідчить про схожий рівень складності задачі для всіх алгоритмів. Це також

підтверджує адекватність збалансованого підходу до підготовки даних та відсутність очевидних переваг одного класу над іншим.

На основі проведеного аналізу, LightGBM рекомендується як оптимальний алгоритм для виявлення шахрайських транзакцій у даному контексті. Модель демонструє найвищу дискримінаційну здатність, швидку конвергенцію та прийнятний баланс між точністю та повнотою. Random Forest може розглядатися як життєздатна альтернатива з майже еквівалентною ефективністю.

ВИСНОВКИ ДО РОЗДІЛУ 2

У результаті проведеного експериментального дослідження математичного моделювання виявлення шахрайських операцій було отримано наступні ключові наукові та практичні результати.

Дослідницький аналіз набору даних дозволив виявити основні статистичні характеристики та закономірності, притаманні транзакційним даним у банківській сфері. Встановлено, що дані характеризуються значним рівнем незбалансованості між класами, складною структурою взаємозв'язків між змінними та наявністю латентних темпоральних патернів, що потребували спеціального підходу до підготовки та обробки.

Розроблена комплексна методологія підготовки даних для моделювання включає інноваційні підходи до розв'язання проблеми незбалансованості класів, створення часових ознак для аналізу темпоральних патернів, розрахунку поведінкових метрик клієнтів та впровадження системи ковзаючих статистик. Особливо важливим досягненням є розробка методики агрегації активності за часовими періодами та створення складних взаємодіючих ознак, що дозволило значно підвищити інформативність ознакового простору.

Впровадження методу Target Encoding для категоріальних змінних забезпечило ефективне кодування висококардинальних категоріальних ознак з урахуванням їх предиктивної сили щодо цільової змінної. Структуризація ознакового простору дозволила створити оптимальну архітектуру даних для подальшого машинного навчання.

Експериментальне порівняння чотирьох алгоритмів машинного навчання продемонструвало суттєві відмінності в ефективності різних підходів. Найвищу дискримінаційну здатність за метрикою ROC-AUC показав алгоритм LightGBM з результатом 0.7593, що свідчить про його оптимальність для розв'язання поставленої задачі. Random Forest продемонстрував майже еквівалентну ефективність з показником 0.7559, підтвердивши переваги ансамблевих методів для виявлення складних патернів шахрайства.

Встановлено, що ансамблеві методи (LightGBM та Random Forest) значно переважають одиничні моделі у контексті виявлення фінансового шахрайства. Neural Network показав помірну ефективність з ROC-AUC 0.7281, тоді як k-найближчі сусіди продемонстрували найнижчий результат 0.6836, що вказує на обмежену придатність цього підходу для даної предметної області.

Критично важливим висновком є необхідність індивідуального налаштування порогу класифікації для кожної моделі. Оптимальні пороги варіювалися від 0.28 для нейронної мережі до 0.43 для LightGBM, що підкреслює важливість точного калібрування моделей для досягнення максимальної практичної ефективності.

Аналіз структури помилок виявив характерну закономірність для всіх досліджуваних моделей: систематично низьку повноту для класу легальних транзакцій (0.36-0.41) при високій повноті для шахрайських операцій (0.86-0.91). Ця характеристика вказує на тенденцію моделей до консервативної класифікації легальних транзакцій та агресивного виявлення потенційного шахрайства, що може бути прийнятним з точки зору мінімізації ризиків у банківській діяльності.

Модель LightGBM продемонструвала найшвидшу конвергенцію, досягнувши оптимального результату на 13-й ітерації, що свідчить про ефективність механізму раннього зупинення та практичну придатність для реального застосування. Показники AUC на навчальній (0.782461) та валідаційній (0.759266) вибірках демонструють помірне перенавчання та хорошу узагальнюючу здатність моделі.

Порівняльний аналіз F1-міри для класу шахрайських транзакцій виявив переваги LightGBM (0.72), що є найкращим результатом серед усіх алгоритмів

та вказує на оптимальний баланс між точністю та повнотою в контексті практичного застосування.

Отримані результати свідчать про те, що створення ефективної системи виявлення шахрайських операцій вимагає комплексного підходу, що включає ретельну підготовку даних, обґрунтований вибір алгоритмів машинного навчання та точне налаштування параметрів моделей. Ансамблеві методи, зокрема LightGBM, демонструють найкращу ефективність для розв'язання задач фінансового моніторингу.

Проведене дослідження створює міцну емпіричну основу для розробки практичних рекомендацій щодо впровадження систем автоматизованого виявлення шахрайства в українських банківських установах, що становитиме предмет розгляду в наступному розділі роботи.

РОЗДІЛ 3

ІМПЛЕМЕНТАЦІЯ МОДЕЛІ ТА ПЕРСПЕКТИВИ РОЗВИТКУ

3.1. Алгоритм реалізації розробленої моделі в банківську систему

Успішне впровадження розробленої математичної моделі виявлення шахрайських операцій у банківську інфраструктуру вимагає комплексного системного підходу, що охоплює стадії попередньої підготовки, технічної реалізації, тестування та контрольованого запуску. Архітектурна концепція інтеграції базується на принципах модульної побудови систем з максимальним ступенем ізоляції аналітичних процесів від критично важливих банківських операцій, що є необхідною умовою для мінімізації ризиків порушення роботи транзакційних платформ під час впровадження нових компонентів.

3.1.1. Етап перший: підготовка та проектування

На попередньому етапі впровадження проводиться комплексний аудит технічної архітектури банківської установи з метою ідентифікації всіх взаємозв'язків між системами обробки, дата-складами та каналами обміну інформацією. При цьому документуються поточні форми циркуляції транзакційних даних, схеми назв полів, тип кодування та стандарти оформлення записів у кожній системі. Критично важливо визначити пропускну спроможність наявних каналів передачі даних, час середнього запиту до основних баз даних та допустимі затримки для різних типів операцій.

Наприклад, для типового українського банку середнього розміру характерні такі параметри: основна система обробки транзакцій на базі Oracle Banking Platform здатна обробляти близько п'яти тисяч операцій на секунду з середнім часом обробки однієї транзакції в діапазоні 50-80 мілісекунд. Дата-склад, розташований на базі SAP BusinessObjects, зберігає історичні дані з глибиною архіву у кілька років та забезпечує доступ до них для аналітичних запитів із затримкою, яка для некритичних аналіз може досягати кількох секунд. Система управління подіями на базі Apache Kafka або RabbitMQ забезпечує асинхронну

доставку повідомлень про операції з гарантією збереження послідовності та безлюдної доставки навіть у випадку тимчасових збоїв.

Крім технічних параметрів, необхідно зібрати детальну інформацію про організаційну структуру відповідальних служб, роль аналітиків безпеки у процесах верифікації та принципи прийняття рішень щодо блокування сумнівних операцій. Ці дані необхідні для обґрунтованого вибору порогів класифікації та механізмів оповіщення про виявлені аномалії. Окрім того, слід вивчити систему звітності та аудиту, яка використовується в банку, щоб забезпечити сумісність логів роботи нової системи з існуючими процедурами контролю та регуляторної звітності.

Формування команди проекту передбачає залучення фахівців з різних спеціальностей, кожен з яких несе відповідальність за своє напрямок роботи. Керівник архітектури рішень координує загальний дизайн системи та забезпечує її сумісність з довгостроковою стратегією розвитку IT-інфраструктури банку. Лідер групи з науки про дані здійснює адаптацію розробленої моделі машинного навчання до специфічних умов банківського операційного середовища, враховуючи особливості локальних даних та встановлені вимоги до якості виявлення. Інженери DevOps відповідають за встановлення контейнеризованого середовища на базі Docker та Kubernetes з забезпеченням автоматизованого розгортання та горизонтального масштабування. Розробники backend-компонентів створюють мікросервіси для обробки даних та взаємодії з основними банківськими системами. Інженери контролю якості розробляють комплексні плани тестування, які охоплюють функціональну перевірку, тести навантаженості та сценарії аварійного відновлення.

Для типового проекту впровадження рекомендується виділити період від шести до дев'яти місяців, розбивши його на чотири основні фази. Період попередньої підготовки охоплює два перші місяці роботи, під час яких встановлюється розвиваюче середовище, підготовлюються набори даних для тестування та документуються поточні бізнес-процеси. Фаза розробки та інтеграції триває наступні два місяці, впродовж яких здійснюється безпосередня

реалізація мікросервісу, налагодження взаємодії з існуючими системами та впровадження моніторингу. Пілотне розгортання відбувається протягом місяців п'ятого та шостого з використанням обмеженої кількості реальних даних для валідації системи в умовах, максимально наближених до промислового навантаження. Остаточна фаза включає контрольоване розширення запуску системи до повного обсягу банківських операцій з паралельною безперервною оптимізацією та накопиченням досвіду експлуатаційного персоналу.

3.1.2. Етап другий: архітектурна розробка та технічна реалізація

Фундаментальною основою технічної реалізації є створення спеціалізованого мікросервісу машинного навчання, який функціонуватиме як незалежна обчислювальна одиниця в екосистемі банківських додатків. Такий архітектурний вибір забезпечує критично важливу ізоляцію процесів аналітики від основних транзакційних систем, що гарантує неможливість розповсюдження технічних проблем у аналітичному модулі на стабільність основних платіжних процесів. Мікросервіс розгортається у контейнеризованому середовищі з використанням технологій Docker для пакування та Kubernetes для оркестрації та управління життєвим циклом контейнерів.

Конкретна архітектура розгортання передбачає наявність кластеру з мінімум трьох вузлів Kubernetes, розташованих на фізично розділених серверах для забезпечення відмовостійкості. Кожен вузол містить кілька інстансів контейнера з моделлю машинного навчання, що дозволяє розподілити навантаження та забезпечити можливість оновлення без припинення сервісу. Балансувальник навантаження, розташований перед кластером, приймає вхідні запити від основних банківських систем та перенаправляє їх на вільні інстанси мікросервісу. Вбудована в Kubernetes система автоматичного масштабування постійно моніторить використання обчислювальних ресурсів та при необхідності запускає додаткові контейнери або видаляє невикористані, забезпечуючи оптимальне співвідношення між якістю сервісу та витратами на обчислення.

Інтеграційний шар між основними банківськими системами та мікросервісом аналітики реалізується через RESTful API інтерфейси, які приймають запити з переліком деталей транзакції та повертають ймовірнісну оцінку належності операції до класу шахрайства. RESTful архітектура обирається через її стандартизованість, простоту тестування та широку підтримку у банківських системах. Кожен запит містить унікальний ідентифікатор транзакції, напрями конвертування валют, суму операції, інформацію про платника та отримувача, тип каналу передачі (мобільний додаток, веб-платформа, банкомат), часові мітки та інші релевантні поля. Асинхронна обробка запитів означає, що основна система не чекає на результат, а вміщує запит в чергу, звідки він поступово обробляється мікросервісом.

Система черг повідомлень на базі Apache Kafka забезпечує надійну доставку всіх транзакційних подій до аналітичного модуля з гарантією послідовності обробки та можливістю повторного відтворення сповіщень у разі необхідності. Kafka зберігає всі повідомлення щонайменше протягом встановленого періоду (зазвичай одна тиждень), що дозволяє провести перенавчання моделі на історичних даних без необхідності отримувати їх безпосередньо з основних систем. Кожна топіка в Kafka розділена на кілька частин, що дозволяє масштабувати пропускну спроможність горизонтально.

Детальний приклад архітектури: під час виконання операції переказу коштів через мобільний додаток основна система генерує повідомлення про цю операцію, яке містить як мінімум такі поля: унікальний ID операції (transaction_id), IBAN платника (sender_iban), IBAN отримувача (beneficiary_iban), суму в гривні (amount_uah), тип операції (transfer_type: "mobile_transfer"), дату та час (timestamp), геолокацію платника (latitude, longitude), IP адресу пристрою. Це повідомлення потрапляє до Kafka "transactions.raw", звідки Python-консюмер мікросервісу його отримує та направляє на обробку.

3.1.3. Етап третій: процес обробки та класифікації

Процес обробки кожної транзакції розбивається на послідовні етапи, що логічно упорядковані для забезпечення якості та надійності. На етапі отримання транзакційних даних система виконує первинну валідацію повноти та коректності інформації, перевіряючи відповідність форматів даних встановленим стандартам та наявність всіх обов'язкових полів. Некоректні або неповні записи, наприклад відсутність обов'язкового поля IBAN отримувача або хибний формат дати, не направляються на подальше обробку основним конвеєром, а направляються до окремої черги для ручної перевірки та логування. Це запобігає дестабілізації роботи основного аналітичного конвеєра та дозволяє операційним службам банку розібратися із проблемою без негативного впливу на обробку інших операцій.

Етап попередньої обробки даних включає нормалізацію числових характеристик транзакцій за допомогою збережених параметрів стандартизації, які були розраховані на етапі навчання моделі на історичних даних. Категоріальні змінні перетворюються через систему Target Encoding, де кожна унікальна категорія замінюється на попередньо розраховане значення, що відображає середню частоту шахрайства серед всіх операцій з цією категорією. Наприклад, якщо із всіх операцій через мобільний додаток лише 0.3 відсотки були шахрайськими, то кожна така операція кодується числом 0.003. Подібний підхід дозволяє моделі швидко захопити статистичні закономірності, пов'язані з категоріальними ознаками.

Створення похідних ознак здійснюється в реальному часі з урахуванням історичних даних клієнта, які зберігаються в оперативному кеші Redis для забезпечення швидкого доступу. Наприклад, система розраховує середню суму операцій даного клієнта за останній місяць, типову кількість операцій на день, середній часовий проміжок між послідовними операціями. Система порівнює поточну операцію з цими метриками: якщо обсяг перевищує середній в п'ять разів, це сигналізує про потенційну аномалію. Окрім того, розраховуються часові ознаки, які відображають день тижня, час доби, сезонність операцій. Система

визначає, чи здійснена операція з нетипового для даного клієнта часу, чи використовується новий канал передачі.

Безпосередня генерація прогнозів здійснюється завантаженою в пам'ять моделлю LightGBM, яка знаходиться в окремому процесі Python-інтерпретатора для забезпечення максимальної швидкості обробки. LightGBM обирається серед усіх можливих алгоритмів машинного навчання через його виняткову продуктивність при роботі з великими обсягами даних та здатність обробляти мілісекундні запити навіть при наявності сотень ознак. Результат моделювання представляється у вигляді ймовірнісної оцінки від 0 до 1, де близька до одиниці ймовірність вказує на високий ризик шахрайства. Наприклад, якщо модель виводить оцінку 0.87, це означає, що, згідно з моделлю, ймовірність того, що дана операція є шахрайською, становить 87 відсотків.

3.1.4. Етап четвертий: прийняття рішень та управління ризиками

Система прийняття рішень базується на багаторівневій логіці обробки результатів моделювання з метою забезпечення оптимального балансу між безпекою та зручністю використання банківських сервісів. Встановлюються три порогові значення ймовірнісних оцінок, які розділяють всі операції на чотири категорії за рівнем ризику. Низькоризикові операції з оцінками нижче 0.15 автоматично пропускаються без додаткових перевірок, що забезпечує мінімальний вплив на швидкість обробки звичайних банківських операцій та задоволеність користувачів якістю сервісу. Операції з оцінками від 0.15 до 0.45 класифікуються як середньоризикові та направляються до системи розширеної перевірки, яка може включати додаткові аналітичні правила, наприклад перевірку IP адреси за географічними базами даних з метою виявлення підозрілої геолокації.

Операції з оцінками від 0.45 до 0.75 вважаються високоризиковими та підлягають обов'язковій верифікації клієнта через одноразовий пароль, надісланий на номер телефону, або завдяки запиту додаткової біометричної аутентифікації. Операції з оцінками вище 0.75 автоматично блокуються з

негайним сповіщенням служби безпеки банку та відправкою повідомлень клієнту через всі доступні канали комунікації про спробу нереалізованої операції. Подібна багаторівневість у прийнятті рішень дозволяє мінімізувати як помилки першого роду (коли легітимні операції помилково блокуються), так і помилки другого роду (коли справжні шахрайські операції пропускаються).

3.1.5. Етап п'ятий: логування, моніторинг та зворотний зв'язок

Система ведення логів фіксує всі прийняті рішення разом з детальним обґрунтуванням, включаючи значення всіх калькульованих ознак, вихідне значення моделі та застосування порогу класифікації. Логи зберігаються як у структурованому форматі JSON для автоматизованої обробки, так і у текстовому вигляді для прочитання людиною. Структурована форма дозволяє легко провести аналітичні запити до накопиченої статистики, наприклад знайти всі випадки, коли модель з низькою впевненістю блокувала операцію з великою сумою.

Механізм зворотного зв'язку дозволяє співробітникам служби безпеки та операційним аналітикам відмічати помилкові спрацювання системи та підтверджувати виявлені випадки шахрайства. На практиці це означає, що коли клієнт звертається в контакт-центр банку зі скаргою, що його легітимна операція була помилково заблокована, служба безпеки може позначити цю операцію як справді легітимну в спеціальній системі управління зворотним зв'язком. Аналогічно, коли служба безпеки виявляє справжній випадок шахрайства, вона відмічає цю операцію у системі. Накопичена таким чином інформація потім використовується як основа для періодичного перенавчання моделі та корекції її параметрів.

Автоматизований процес аналізу якості прогнозів щоденно генерує детальні звіти про ключові метрики ефективності системи, порівнюючи дані з попередніми днями та виявляючи тренди. Звіт включає такі показники: загальна кількість оброблених операцій за день, розподіл операцій за категоріями ризику, кількість блокованих операцій, кількість операцій, які потребували вторинної

верифікації, середній час обробки запиту, кількість помилок валідації даних. Система порівнює ці числа з середніми значеннями за попередні дні та сигналізує про значні відхилення, які можуть вказувати як на технічні проблеми, так і на появу нових тенденцій у поведінці шахраїв.

3.1.6. Етап шостий: масштабування та забезпечення відмовостійкості

Архітектура системи передбачає горизонтальне масштабування через простоту додавання нових інстансів мікросервісу при зростанні навантаження. Балансувальник навантаження розподіляє вхідні запити між усіма доступними інстансами, гарантуючи, що жодна машина не буде перевантажена. Система автоскейлінгу Kubernetes автоматично адаптує кількість запущених процесів до поточного рівня навантаження, за замовчуванням встановлюючи мінімум три копії мікросервісу та максимум п'ятнадцять копій при піковому навантаженні. Таке рішення забезпечує економічність витрат ресурсів в період низького навантаження та достатню пропускну спроможність при пікових годинах роботи банку.

Забезпечення безпеки включає шифрування всіх каналів передачі даних за допомогою протоколу HTTPS з використанням сертифікатів, що відповідають галузевим стандартам. Автентифікація доступу до API здійснюється через використання OAuth 2.0 токенів, що генеруються централізованим сервером аутентифікації банку. Авторизація контролюється через систему ролей та дозволів, яка гарантує, що тільки авторизовані компоненти основної банківської системи можуть обертатися до мікросервісу аналітики. Детальні аудит-логи всіх операцій ведуться в захищеному сховищ з обмеженим доступом та використовуються для забезпечення можливості розслідування в разі виявлення інцидентів безпеки.

3.1.7. Практичний приклад впровадження для банку

Розглянемо конкретний сценарій впровадження для середнього українського банку з обсягом близько двох мільйонів транзакцій на місяць. Перший місяць виділено на встановлення Kubernetes кластеру з трьома фізичними серверами,

встановлення всіх необхідних компонентів та підготовку даних для тестування. Протягом другого місяця розроблюється мікросервіс, налагоджується його взаємодія з основною системою та впроваджується базовий моніторинг. Місяці третій і четвертий присвячені інтенсивному тестуванню з використанням як синтетичних, так і реальних даних з тестового середовища. На п'ятому місяці розпочинається пілотне розгортання на п'яти відсотках реальних операцій, протягом якого система працює паралельно з існуючою в режимі виключно для логування результатів без впливу на реальні рішення про блокування.

Після успішного пілотного тестування протягом місяця система поступово розширюється: спочатку на десять відсотків операцій, потім на двадцять п'ять відсотків, потім на п'ятдесят відсотків та нарешті на сто відсотків. Кожен цей етап супроводжується детальним аналізом якості роботи та зворотного зв'язку від операційного персоналу. На шостому місяці розпочинається робота з повним навантаженням, де система активно впливає на прийняття рішень про блокування операцій. Паралельно роботи з оптимізації продовжуються протягом наступних трьох місяців, в ході яких здійснюється тонка настройка порогів класифікації на основі накопленого досвіду та зворотного зв'язку.

Результатом впровадження є функціональна система, яка обробляє два мільйони операцій щомісяця з часом відповіді менше ста мілісекунд, забезпечує та виявляє значний відсоток справжніх випадків шахрайства при кількох відсотках помилкових спрацювань. Операційний персонал банку отримує зручний інтерфейс для управління зворотним зв'язком та налаштування параметрів системи без необхідності глибоких технічних знань машинного навчання.

3.2. Оцінка ефективності впровадженої системи

Оцінка ефективності впровадженої системи виявлення шахрайських операцій здійснюється через комплексну систему ключових показників ефективності, що охоплюють технічні, операційні та бізнес-аспекти функціонування. Методологія

оцінки базується на принципах безперервного моніторингу та багатовимірного аналізу результативності системи в умовах реального банківського середовища.

Технічні метрики включають показники продуктивності системи, такі як час відповіді на запит обробки транзакції, пропускна здатність системи та використання обчислювальних ресурсів. Критично важливим є моніторинг латентності системи, оскільки час від надходження транзакції до генерації рішення не повинен перевищувати 100 мілісекунд для забезпечення прийнятної швидкості обробки банківських операцій. Система відслідковує доступність сервісу та автоматично генерує попередження при виявленні технічних аномалій.

Операційні показники фокусуються на точності виявлення шахрайських операцій в реальних умовах. Основними метриками є частка виявлених справжніх випадків шахрайства, частка помилкових спрацювань, точність позитивних прогнозів та збалансована міра ефективності. Система безперервно відслідковує динаміку цих показників у часі для виявлення можливої деградації моделі та необхідності її перенавчання.

Бізнес-метрики включають фінансові показники впливу системи на діяльність банку, зокрема загальну суму запобіжених збитків від шахрайства, вартість помилкових спрацювань та операційні витрати на утримання системи. Ці метрики дозволяють розрахувати повернення інвестицій та обґрунтувати доцільність подальшого розвитку системи.

Система безперервного моніторингу якості реалізується через автоматизовані дашборди та аналітичні звіти, що генеруються в режимі реального часу. Дашборд операційних метрик відображає поточний стан системи з оновленням кожні п'ять хвилин та включає графіки розподілу ризик-скорів по транзакціях, кількість заблокованих операцій за період та середній час обробки запитів. Автоматичні попередження спрацьовують при виявленні аномальних відхилень від базових показників.

Система аналізу якості прогнозів здійснює щоденну верифікацію точності моделі через порівняння прогнозованих та фактично підтверджених випадків

шахрайства. Формується детальна статистика по основних метриках з аналізом трендів та сезонних коливань. Система автоматично виявляє зниження якості моделі та ініціює процедури перенавчання при досягненні критичних порогових значень.

Модуль аналізу помилок класифікує всі помилкові спрацювання системи за категоріями та виявляє систематичні патерни неточностей. Це дозволяє ідентифікувати слабкі місця моделі та напрямки для її покращення. Результати аналізу використовуються для корекції порогів класифікації та оновлення правил обробки підозрілих транзакцій.

Економічна оцінка ефективності базується на аналізі результатів експериментального тестування найкращої моделі LightGBM та екстраполяції цих результатів на масштаби реального банківського навантаження. Для економічних розрахунків використовуються базові параметри типового українського банку середнього розміру, включаючи щомісячний обсяг у два мільйони транзакцій, середню суму транзакції 49 500 гр. од. та фактичну частоту шахрайства в реальних умовах на рівні 0,5 відсотка від загального обсягу операцій.

На основі результатів тестування модель LightGBM, розрахунок фінансових результатів показує значний позитивний економічний ефект від впровадження системи. Кількість виявлених справжніх випадків шахрайства складає 9 100 операцій на місяць при загальній кількості шахрайських транзакцій 10 000. Сума заощаджених втрат досягає 448 630 000 гр. од. щомісяця, що в річному обчисленні становить понад 5,3 мільярда гр. од.

Втрати від пропущених випадків шахрайства складають 900 операцій на місяць з сумою збитків 44 370 000 гр. од. щомісяця.

Технічні витрати на систему включають одноразові витрати на розробку та впровадження, що охоплюють розробку системи, інтеграцію з існуючими платформами та навчання персоналу. Розрахунок загального економічного ефекту демонструє чистий річний прибуток від впровадження системи після врахування всіх витрат та збитків.

При збільшенні обсягів транзакцій у два рази операційні витрати зростають лише на тридцять відсотків, тоді як позитивний ефект масштабується лінійно. Це створює додаткові стимули для розширення використання системи при зростанні бізнесу банку.

Окрім прямого економічного ефекту, система забезпечує стратегічні переваги, що мають довгостроковий позитивний вплив на діяльність банку. Покращення клієнтського досвіду через зниження кількості помилково заблокованих операцій та швидшу обробку легітимних транзакцій підвищує задоволеність клієнтів банківськими послугами та сприяє їх утриманню.

Підвищення репутації банку як результат ефективного виявлення та запобігання шахрайству зміцнює довіру клієнтів та позитивно впливає на позиціонування установи на ринку фінансових послуг. Це може призвести до залучення нових клієнтів та розширення ринкової частки банку.

Автоматизована система моніторингу забезпечує виконання вимог регуляторів щодо протидії фінансовим злочинам та відмиванню коштів. Це знижує ризики регуляторних санкцій та репутаційних втрат, пов'язаних з порушеннями вимог нагляду.

Впровадження передових технологій машинного навчання дозволяє банку позиціонувати себе як інноваційну установу та залучати технологічно орієнтованих клієнтів. Це створює конкурентні переваги в умовах цифровізації банківського сектору та може сприяти розвитку додаткових цифрових продуктів та сервісів.

3.3. Напрями вдосконалення системи виявлення шахрайських операцій

Еволюція методів фінансового шахрайства та постійне ускладнення злочинних схем вимагають безперервного вдосконалення систем виявлення та протидії незаконним операціям. Аналіз поточного стану розробленої системи та перспективних технологічних трендів дозволяє ідентифікувати кілька ключових напрямків для подальшого розвитку аналітичних можливостей банку.

Розширення джерел даних для аналізу представляє один з найперспективніших напрямків удосконалення системи. Інтеграція додаткової контекстної інформації може значно підвищити точність виявлення шахрайських патернів та зменшити кількість помилкових спрацювань. Геолокаційні дані високої точності, отримані з мобільних додатків банку, дозволяють аналізувати просторові аномалії у поведінці клієнтів та виявляти операції, здійснені з нетипових для конкретного користувача локацій.

Поведінкова біометрія представляє інноваційний напрямок, що базується на аналізі унікальних характеристик взаємодії користувачів з цифровими інтерфейсами. Параметри натискання клавіш, швидкість вводу інформації, характер рухів миші та патерни скролінгу створюють індивідуальний цифровий відбиток кожного користувача. Аналіз відхилень від звичайних поведінкових патернів може сигналізувати про потенційну компрометацію облікового запису навіть при використанні коректних автентифікаційних даних.

Інтеграція даних з відкритих джерел та соціальних мереж відкриває можливості для контекстного аналізу ризиків. Інформація про фінансовий стан клієнтів, їхню професійну діяльність та соціальні зв'язки може допомогти у виявленні невідповідностей між заявленими доходами та фактичними фінансовими операціями. Водночас, такий підхід потребує ретельного дотримання вимог приватності та відповідних регуляторних норм.

Вдосконалення алгоритмічної основи системи включає експерименти з ансамблевими методами, що комбінують переваги різних підходів машинного навчання. Стекінг моделі, що використовують мета-навчання для оптимального поєднання прогнозів базових алгоритмів, можуть забезпечити вищу точність та стабільність результатів. Особливу увагу варто приділити інтеграції градієнтного бустингу з нейронними мережами для захоплення як лінійних, так і високо нелінійних залежностей у даних.

Впровадження технік онлайн-навчання дозволить системі адаптуватися до нових типів шахрайства в режимі реального часу без необхідності повного перенавчання моделі. Алгоритми поступового навчання можуть інкорпорувати

нову інформацію про виявлені інциденти та корегувати параметри моделі для покращення майбутніх прогнозів. Це особливо важливо в контексті швидкої еволюції кіберзлочинності та появи нових напрямки атак.

Розвиток пояснювального штучного інтелекту представляє критично важливий напрямок для підвищення довіри до автоматизованих рішень та забезпечення відповідності регуляторним вимогам. Техніки SHAP та LIME дозволяють пояснити внесок кожної ознаки у прийняття конкретного рішення моделі, що надає аналітикам інструменти для валідації логіки системи та ідентифікації потенційних помилок.

Автоматизація процесів управління життєвим циклом моделей включає розробку систем тривалої інтеграції та тривалого розвитку для машинного навчання. MLOps платформи забезпечують автоматизоване тестування нових версій моделей, контроль якості прогнозів та безшовне розгортання оновлень у продуктивному середовищі. Такий підхід мінімізує ризики людських помилок та забезпечує швидке впровадження покращень.

Система автоматичного моніторингу дрифту даних та деградації моделі дозволяє проактивно ідентифікувати ситуації, коли зміни у вхідних даних призводять до зниження якості прогнозів. Алгоритми виявлення зміщення концепцій можуть автоматично сигналізувати про необхідність перенавчання моделі або корекції її параметрів. Це особливо важливо в умовах динамічної банківської індустрії, де зміни у регуляціях, технологіях та поведінці споживачів можуть впливати на ефективність аналітичних систем.

Розвиток федеративного навчання створює можливості для співпраці між банківськими установами у боротьбі з шахрайством без порушення конфіденційності клієнтських даних. Технології конфіденційне машинне навчання дозволяють тренувати спільні моделі на об'єднаних даних кількох банків, при цьому зберігаючи чутливу інформацію локально. Такий підхід може значно підвищити якість моделей через доступ до більших та різноманітніших навчальних вибірок.

Інтеграція з системами штучного інтелекту для аналізу неструктурованих даних відкриває нові можливості для виявлення шахрайства. Обробка природної мови може аналізувати текстові описи транзакцій, комунікації з клієнтами та внутрішні звіти для виявлення підозрілих патернів. Технології «комп'ютерного зору» можуть аналізувати зображення документів та фотографії для верифікації автентичності наданої інформації.

Розробка персоналізованих моделей ризику для різних клієнтських сегментів може значно підвищити точність виявлення аномалій. Замість використання єдиної універсальної моделі, система може автоматично обирати найбільш відповідну модель на основі характеристик клієнта та типу транзакції. Такий підхід враховує різноманітність поведінкових патернів різних груп користувачів та забезпечує більш нюансований аналіз ризиків.

Впровадження блокчейн технологій для забезпечення незмінності аудит-логів та прозорості процесів прийняття рішень може підвищити довіру до системи з боку регуляторів та клієнтів. Розподілені леджери можуть зберігати хеші ключових рішень системи, забезпечуючи можливість верифікації цілісності історичних даних та запобігаючи несанкціонованим змінам у критично важливих записах.

ВИСНОВКИ ДО РОЗДІЛУ 3

У третьому розділі дипломної роботи було детально розроблено та обґрунтовано комплексний підхід до імплементації моделі виявлення шахрайських операцій у банківську систему, проведено оцінку її ефективності та визначено перспективні напрями подальшого розвитку.

Розроблений алгоритм реалізації базується на архітектурі мікросервісів з використанням контейнеризованих технологій Docker та Kubernetes, що забезпечує ізоляцію аналітичних процесів від критичних банківських систем. Впровадження RESTful API з асинхронною обробкою запитів та системи черг повідомлень Apache Kafka гарантує надійну обробку великих обсягів транзакційних даних без порушення основних банківських процесів.

Багаторівнева система прийняття рішень дозволяє автоматично обробляти транзакції різних рівнів ризику, забезпечуючи баланс між ефективністю виявлення шахрайства та мінімізацією впливу на користувацький досвід.

Комплексна система оцінки ефективності, що охоплює технічні, операційні та бізнес-метрики, продемонструвала значний позитивний економічний ефект від впровадження. Модель LightGBM забезпечує виявлення 91% справжніх випадків шахрайства з часом відповіді менше 100 мілісекунд, що відповідає вимогам реального банківського середовища.

Аналіз перспектив розвитку системи виявив широкі можливості для подальшого вдосконалення через інтеграцію додаткових джерел даних, впровадження поведінкової біометрії та геолокаційного аналізу. Особливого значення набуває розвиток пояснювального штучного інтелекту та автоматизація процесів управління життєвим циклом моделей через MLOps платформи. Перспективними є також федеративне навчання для міжбанківської співпраці та персоналізовані моделі ризику для різних клієнтських сегментів.

Результати дослідження підтверджують практичну доцільність та економічну ефективність впровадження розробленої системи, а також демонструють її потенціал для масштабування та адаптації до мінливих умов банківської індустрії. Запропонована методологія може служити основою для створення сучасних систем протидії фінансовому шахрайству в українських банках.

ВИСНОВКИ

У результаті проведеного дослідження розробки та впровадження моделі виявлення шахрайських операцій для систем цифрового банкінгу було досягнуто поставленої мети та розв'язано всі визначені завдання. Отримано наступні ключові наукові та практичні результати.

Здійснено комплексне дослідження теоретичних основ банківського шахрайства, що дозволило створити багатокритеріальну класифікацію шахрайських операцій за суб'єктами, об'єктами, способами здійснення та технологіями. Встановлено, що сучасне фінансове шахрайство характеризується високим рівнем технологічної складності та швидкої адаптації до існуючих засобів протидії.

Проведено систематичний аналіз методологічних підходів до моделювання систем виявлення шахрайства, який виявив еволюцію від традиційних статистичних методів до сучасних технологій машинного навчання та штучного інтелекту. Обґрунтовано переваги гібридних систем, які поєднують різні алгоритмічні підходи для максимізації ефективності виявлення.

Аналіз світового досвіду провідних фінансових установ засвідчив тенденції переходу до проактивних підходів, інтеграції різнотипних даних та впровадження технологій обробки у реальному часі. Дослідження комерційних рішень показало високий рівень технологічної зрілості ринку з акцентом на хмарні технології та автоматизоване машинне навчання.

Розроблено та апробовано комплексну методологію підготовки даних для моделювання, яка включає інноваційні підходи до розв'язання проблеми незбалансованості класів, створення темпоральних ознак, розрахунку поведінкових метрик клієнтів та впровадження системи ковзаючих статистик. Особливо важливим досягненням є розробка методики Target Encoding для висококардинальних категоріальних змінних.

Проведено експериментальне порівняння чотирьох алгоритмів машинного навчання (LightGBM, Random Forest, Neural Network, k-NN), що

продемонструвало суттєві відмінності в ефективності різних підходів. Встановлено, що алгоритм LightGBM показав найвищу дискримінаційну здатність з ROC-AUC 0.7593 та оптимальним балансом точності й повноти (F1-міра 0.72 для класу шахрайських операцій).

Виявлено критичну важливість індивідуального налаштування порогу класифікації для кожної моделі, при цьому оптимальні значення варіювалися від 0.28 до 0.43. Аналіз структури помилок показав характерну для всіх моделей тенденцію до консервативної класифікації легальних транзакцій, що є прийнятним з точки зору мінімізації банківських ризиків.

Розроблено комплексний алгоритм впровадження системи виявлення шахрайських операцій у банківську інфраструктуру, що базується на архітектурі мікросервісів з використанням технологій Docker, Kubernetes та Apache Kafka. Запропонована архітектура забезпечує ізоляцію аналітичних процесів, масштабованість та високу доступність системи.

Економічна оцінка ефективності продемонструвала значний позитивний ефект від впровадження системи. Для типового українського банку середнього розміру розрахований річний ефект перевищує 5,3 мільярда гр. од. запобіжених втрат при виявленні 91% справжніх випадків шахрайства. Отримані результати підтверджують високу економічну доцільність інвестицій у подібні системи.

Ідентифіковано ключові напрями подальшого вдосконалення системи, включаючи інтеграцію геолокаційних даних та поведінкової біометрії, розвиток пояснювального штучного інтелекту, впровадження MLOps платформ та федеративного навчання. Особливого значення набуває створення персоналізованих моделей ризику та використання неструктурованих даних для підвищення точності виявлення.

Власний внесок авторки полягає у розробці комплексної методології створення систем виявлення банківського шахрайства, що поєднує сучасні технології машинного навчання з практичними вимогами банківської діяльності. Запропоновано оригінальний підхід до інженерії ознак з урахуванням специфіки

фінансових даних та розроблено архітектурні рішення для ефективної інтеграції аналітичних систем у банківську інфраструктуру.

Результати дослідження мають безпосередню практичну цінність для української банківської галузі та можуть бути використані для створення сучасних систем протидії фінансовому шахрайству. Розроблена методологія забезпечує науково обґрунтований підхід до впровадження технологій машинного навчання у критично важливих банківських процесах з урахуванням вимог безпеки, надійності та економічної ефективності.

Отримані результати створюють основу для подальших досліджень у сфері фінансових технологій та можуть сприяти підвищенню рівня кібербезпеки української банківської системи в умовах цифровізації економіки.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Kara İ. Electronic Banking (e-Banking) Fraud with Phishing Attack Methods . *European Journal of Science and Technology*. 2021. № 31. С. 982-985. URL:https: .dergipark.org.tr/en/download/article-file/2003077 (дата звернення: 15.02.2025).
2. Belbergui Ch. E-banking Overview: Concepts, Challenges and Solutions . *Applied Sciences*. 2021 Vol. 117 P. 1059-1078. URL: https://link.springer.com/article/10.1007/s11277-020-07911-0 (дата звернення: 15.02.2025).
3. Національний банк України. Причиною більшості шахрайських випадків з платіжними картками стало розголошення даних їхніми користувачами. URL: https://bank.gov.ua/ua/news/all/prichinoyu-bilshosti-shahrayskih-vipadkiv-z-platijnimi-kartkami-stalo-rozgoloshennya-danih-yihnimi-koristuvachami (дата звернення: 15.07.2025).
4. Fadayo O. M. An Examination of E-Banking Fraud Prevention and Detection in Nigerian Banks. Leicester: De Montfort University, Faculty of Business & Law, Department of Accounting & Finance, 2018. Листопад. URL: https://d1wqtxts1xzle7.cloudfront.net/96035708/228181713-libre.pdf (дата звернення: 15.02.2025).
5. Syniavska O., Dekhtyar N., Deyneka O., Zhukova T., Syniavska O. Security of e-banking systems: modelling the process of counteracting e-banking fraud. *SHS Web of Conferences*. 2019. Vol. 65. 03004. URL: https://www.shs-conferences.org/articles/shsconf/abs/2019/06/shsconf_m3e22019_03004/shsconf_m3e22019_03004.html (дата звернення: 15.02.2025).
6. ЄМА. Монобанк приєднався до міжбанківського сервісу відслідковування та блокування шахрайських операцій Fraud Payments Tracker [Електронний ресурс]. 2021. URL: https://www.ema.com.ua/news/monobank-priiednavsja-do-mizhbankivskogo-servisu-vidslidkovuvannja-ta-blokuvannja-shahrajskih-operacij-fraud-payments-tracker/ (дата звернення: 15.02.2025).
7. Мельник К., Лавренчук С., Христинець Н. Виявлення шахрайства з кредитними картками методами машинного навчання. *Вісник Хмельницького національного університету*. 2024. № 2 (333). С. 189. URL: https://heraldts.khmnu.edu.ua/index.php/heraldts/article/view/132/115 (дата звернення: 15.02.2025).
8. Bank for International Settlements. Basel Framework: Chapter OPE10. 15.12.2019. URL: https://www.bis.org/basel_framework/chapter/OPE/10.htm (дата звернення: 22.04.2025).

9. Anti-Phishing Working Group. Trends Reports. URL: https://apwg.org/trendsreports/ (дата звернення: 22.04.2025).
10. Mitnick K. D., Simon W. L. The Art of Deception: Controlling the Human Element of Security. Indianapolis: Wiley Publishing, 2002. 364 p.
11. Cialdini R. B. Influence: The Psychology of Persuasion. New York: Harper Business, 2007. 336 p.
12. Атака на водойму: що це таке та як від неї захиститися. URL: https://hackyourmom.com/kibervijna/ataka-na-vodojmu-shho-cze-take-ta-yak-vid-neyi-zahystytysya/ (дата звернення: 27.04.2025).
13. Anti-Phishing Working Group. APWG Q4 Report Finds 2023 Was Record Year for Phishing. 2023. URL: https://apwg.org/apwg-q4-report-finds-2023-was-record-year-for-phishing/ (дата звернення: 27.04.2025).
14. CrowdStrike. A Human at the Keyboard: CrowdStrike Reports 60% Jump in Interactive Intrusions. URL: https://crowdstrike.podbean.com/e/a-human-at-the-keyboard-crowdstrike-reports-60-jump-in-interactive-intrusions/ (дата звернення: 27.04.2025).
15. CBS News. Online scams 2024: Statistics, FBI report. 2024. URL: https://www.cbsnews.com/news/online-scams-2024-statistics-fbi/ (дата звернення: 27.04.2025).
16. Національний банк України. Стартував проєкт із протидії кібершахрайству у фінансовому секторі. 2024. URL: https://bank.gov.ua/ua/news/all/startuvav-proyekt-iz-protidiyi-kibershahraystvu-u-finansovomu-sektori (дата звернення: 27.04.2025).
17. Національне антикорупційне бюро України. Аналітична довідка щодо кіберзагроз у банківському секторі URL: https://nabu.ua/images/uploaded/sys_media_doc/doc_2f49928c52c9ce6011977b9c0eef87f8.pdf (дата звернення: 03.05.2025).
18. Adeniran A. A., Obiki-Osafiele O., Agu O., Efunniyi P. P., Abhulimen A. O. Data-Driven Approaches to Improve Customer Experience in Banking: Techniques and Outcomes. *International Journal of Management & Entrepreneurship Research*. 2024. Vol. 6, Issue 8. P. 2797-2818. URL https://www.researchgate.net/publication/383860195_Data-Driven_approaches_to_improve_customer_experience_in_banking_Techniques_and_outcomes (дата звернення: 15.06.2025).
19. JPMorgan Chase & Co. Payments Data for Fraud Detection 2024. URL: https://www.lum.ventures/blog/ais-impact-on-financial-fraud-jp-morgan-case-study (дата звернення: 15.06.2025).

20. DigitalDefynd. 10 ways JP Morgan is using AI [In Depth Case Study]. 2025. URL: https://digitaldefynd.com/IQ/jp-morgan-using-ai-case-study/ (дата звернення: 15.06.2025).
21. Crosman P. JPMorgan Chase Using Advanced AI to Detect Fraud. 03.07.2023. URL: https://www.americanbanker.com/news/jpmorgan-chase-using-chatgpt-like-large-language-models-to-detect-fraud (дата звернення: 15.06.2025).
22. Tulsi K., Dutta A., Singh N., Jain D. Transforming Financial Services: The Impact of AI on JP Morgan Chase's Operational Efficiency and Decision-Making. International Journal of Scientific Research & Engineering Trends. 2024. Vol. 10, Issue 1. P. 138. URL: https://ijsret.com/wp-content/uploads/2024/01/IJSRET_V10_issue1_138.pdf (дата звернення: 15.06.2025).
23. Harnessing the power of AI to fight financial crime. HSBC Holdings plc. 24.06.2025. URL: https://www.hsbc.com/news-and-views/views/hsbc-views/harnessing-the-power-of-ai-to-fight-financial-crime (дата звернення: 21.06.2025).
24. Technology. Fighting Financial Crime. HSBC Holdings plc. 2.03.2025. URL: https://www.hsbc.com/who-we-are/esg-and-responsible-business/fighting-financial-crime/technology (дата звернення: 21.06.2025).
25. Here's How HSBC is Using Artificial Intelligence to Take Money Launderers to the Cleaners. Future Digital Finance. 06.08.2024. URL: https://futuredigitalfinance.wbresearch.com/blog/hsbc-artificial-intelligence-strategy-to-beat-money-launderers (дата звернення: 21.06.2025).
26. How banks can mitigate fraud & financial crimes with AI// Finance Alliance. 17.06.2025. URL: https://www.financealliance.io/ai-in-risk-management-how-banks-can-mitigate-fraud-and-financial-crimes/ (дата звернення: 21.06.2025).
27. Credit card fraud detection based on federated graph learning// ScienceDirect. 08.08.2024. URL: https://www.sciencedirect.com/science/article/abs/pii/S0957417424018463 (дата звернення: 21.06.2025).
28. Real-Time Financial Fraud Detection Using Adaptive Graph Neural Networks and Federated Learning. ResearchGate. 29.03.2025. URL: https://www.researchgate.net/publication/389053167_Real-Time_Financial_Fraud_Detection_Using_Adaptive_Graph_Neural_Networks_and_Federated_Learning (дата звернення: 21.06.2025).
29. AI in finance: Role, benefits and use cases. DXC Luxoft. 19.11.2024. URL: https://www.luxoft.com/blog/dxcluxoft-ai-financial-services-what-the-future-holds (дата звернення: 25.06.2025).

30. Barclays employs voice biometrics to authenticate phone banking customers. Biometric Update. 04.08.2016. URL: https://www.biometricupdate.com/201608/barclays-employs-voice-biometrics-to-authenticate-phone-banking-customers (дата звернення: 25.06.2025).
31. Responsible AI in banking: Gaining a competitive edge. DBS Bank URL: https://www.dbs.com/artificial-intelligence-machine-learning/artificial-intelligence/responsible-ai-in-banking-gaining-a-competitive-edge.html (дата звернення: 27.06.2025).
32. Barclays makes another move into biometric security// FinTech Futures. 10.02.2025. URL:https://www.fintechfutures.com/biometrics-id-verification/barclays-makes-another-move-into-biometric-security (дата звернення: 25.06.2025).
33. Voice Biometrics Recognition and Opportunities It Gives/ Sciforce. Medium. 13.05.2024. URL:https://medium.com/sciforce/voice-biometrics-recognition-and-opportunities-it-gives-d0f8b610f01e (дата звернення: 25.06.2025).
34. AI-Powered Personalised Nudges & Investment Ideas// DBS Bank. URL:https://www.dbs.com/artificial-intelligence-machine-learning/index.html (дата звернення: 27.06.2025).
35. Voice biometrics: Deep learning-based voiceprint authentication system// IEEE Xplore. URL:https://ieeexplore.ieee.org/document/7994971/ (дата звернення: 26.06.2025).
36. Insights: Keith Little// Barclays. 27.02.2020. URL:https://home.barclays/news/2020/02/insights---fraudsters-are-getting-smarter---but-so-are-we-/ (дата звернення: 25.06.2025).
37. Artificial Intelligence at DBS Bank Two Use Cases// Emerj Artificial Intelligence Research. URL:https://emerj.com/artificial-intelligence-at-dbs-bank/ (дата звернення: 27.06.2025).
38. Use cases of machine learning in banking & finance// DBS Bank. URL:https://www.dbs.com/artificial-intelligence-machine-learning/machine-learning/use-cases-of-machine-learning-in-banking-and-finance.html (дата звернення: 27.06.2025).
39. Digital transformation: banking & finance in Singapore// DBS Bank. URL:https://www.dbs.com/artificial-intelligence-machine-learning/artificial-intelligence/digital-transformation-banking-finance-singapore.html (дата звернення: 27.06.2025).
40. 5 Ways DBS Sets Standard for Anti-Fraud Leadership in Singapore// Sights in Plus. 12.05.2025. URL:https://sightsinplus.com/insight/leadership/5-ways-dbs-sets-standard-for-anti-fraud-leadership-in-singapore/ (дата звернення: 27.06.2025).

41. Artificial intelligence: the future of banking & finance// DBS Bank. URL:https://www.dbs.com/artificial-intelligence-machine-learning/artificial-intelligence/ai-the-future-of-banking-and-finance.html (дата звернення: 27.06.2025).
42. Sign up for Digibank with Singpass Face Verification// DBS Singapore. URL:https://www.dbs.com.sg/personal/deposits/bank-with-ease/singpassfaceverification (дата звернення: 27.06.2025).
43. 5 Proven Ways Biometric Authentication Boosts Banking Security// Number Analytics. URL:https://www.numberanalytics.com/blog/biometric-banking-security-ways (дата звернення: 27.06.2025).
44. Help & Support Fingerprint / Face ID authentication for DBS digibank HK// DBS Bank Hong Kong. URL:https://www.dbs.com.hk/personal/support/digital-digibank-biometrics-authentication.html (дата звернення: 27.06.2025).
45. Biometric Authentication Technology In Mobile Banking// Binariks. 29.03.2023. URL:https://binariks.com/blog/biometric-security-onilne-banking/ (дата звернення: 27.06.2025).
46. Strengthening Digital Token Setup with Singpass Face Verification// DBS Singapore. URL:https://www.dbs.com.sg/personal/support/bank-digital-token-setup-face-verification.html (дата звернення: 27.06.2025).
47. OCBC Bank Is The First Singapore Bank To Tap Artificial Intelligence And Machine Learning To Combat Financial Crime// Fintech Singapore. 08.03.2024. URL:https://fintechnews.sg/14047/fintech/ocbc-bank-first-singapore-bank-tap-artificial-intelligence-machine-learning-combat-financial-crime/ (дата звернення: 27.06.2025).
48. OCBC deepens quantum research with Singapore universities for real-world finance use cases// The Edge Singapore. 17.07.2025. URL:https://www.theedgesingapore.com/digitaledge/banking-finance/ocbc-deepens-quantum-research-singapore-universities-real-world-finance (дата звернення: 27.06.2025).
49. Enjoy Enhanced Security with DBS Digital Token+// DBS Singapore. 10.04.2019. URL:https://www.dbs.com.sg/corporate/solutions/ideal/seamless-security (дата звернення: 27.06.2025).
50. Additional security for your Digital Token// DBS Singapore. URL:https://www.dbs.com.sg/personal/support/bank-ibanking-digital-token-additional-security.html (дата звернення: 27.06.2025).
51. DBS makes corporate account opening for SMEs a snap with biometric face verification technology// DBS Bank. URL:https://www.dbs.com/newsroom/DBS_makes_corporate_account_openin

g_for_SMEs_a_snap_with_biometric_face_verification_technology (дата звернення: 27.06.2025).
52. Detecting credit card fraud using machine learning// OCBC Bank. URL:https://www.ocbc.com/group/careers/blog/detecting-credit-card-fraud.page (дата звернення: 27.06.2025).
53. Solving Financial Fraud Detection with Machine Learning Methods// Avenga. 18.06.2025. URL:https://www.avenga.com/magazine/fraud-detection-machine-learning/ (дата звернення: 27.06.2025).
54. Explainable Deep Behavioral Sequence Clustering for Transaction Fraud Detection// ArXiv. 12.01.2021. URL:https://arxiv.org/abs/2101.04285 (дата звернення: 27.06.2025).
55. Needhi J. How AI Transformed Financial Fraud Detection: A Case Study of JP Morgan Chase// Medium. 08.07.2024. URL:https://medium.com/@jeyadev_needhi/how-ai-transformed-financial-fraud-detection-a-case-study-of-jp-morgan-chase-f92bbb0707bb (дата звернення: 25.06.2025).
56. Machine learning algorithms for document clustering and fraud detection// ResearchGate. 01.08.2016. URL:https://www.researchgate.net/publication/312560432_Machine_learning_algorithms_for_document_clustering_and_fraud_detection (дата звернення: 27.06.2025).
57. OCBC Bank's fraud detection recovers SG$8 million in losses// Tech Wire Asia. 31.08.2021. URL:https://techwireasia.com/2021/08/ocbc-banks-fraud-detection-recovers-sg-8-million-worth-of-fraudulent-transactions/ (дата звернення: 27.06.2025).
58. OCBC deepens quantum research with Singapore universities for real-world finance use cases// Yahoo Finance Singapore. 16.07.2025. URL:https://sg.finance.yahoo.com/news/ocbc-deepens-quantum-research-singapore-235556105.html (дата звернення: 27.06.2025).
59. Top 10 FICO Falcon Platform Alternatives// PeerSpot. URL:https://www.peerspot.com/products/fico-falcon-platform-alternatives-and-competitors (дата звернення: 25.06.2025).
60. BAE Systems NetReveal vs SAS Fraud Management comparison// PeerSpot. URL:https://www.peerspot.com/products/comparisons/bae-systems-netreveal_vs_sas-fraud-management (дата звернення: 25.06.2025).
61. FICO Machine Learning Algorithms Improve Card-Not-Present Fraud Detection by 30%// FICO. URL:https://www.fico.com/en/newsroom/fico-machine-learning-algorithms-improve-card-not-present-fraud-detection-30 (дата звернення: 27.06.2025).

62. FICO Falcon Fraud Manager: Introducing Falcon 7.0// FICO. URL:https://www.fico.com/en/latest-thinking/solution-sheet/fico-falcon-fraud-manager-introducing-falcon-7-0 (дата звернення: 27.06.2025).
63. Real-Time Transaction Processing in FICO Falcon// FICO Architecture Guide. URL:https://www.fico.com/en/resource-center/architecture/real-time-transaction-processing (дата звернення: 27.06.2025).
64. FICO Falcon 10.0: The Next Generation of Fraud Detection// FinTech Magazine. URL:https://fintechmagazine.com/articles/fico-falcon-10-0-next-generation-fraud-detection (дата звернення: 27.06.2025).
65. FICO® Falcon® Fraud Manager// FICO. URL:https://www.fico.com/en/products/fico-falcon-fraud-manager (дата звернення: 27.06.2025).
66. Falcon Fraud Manager// FinancialIT. URL:https://financialit.net/products/banking/falcon-fraud-manager (дата звернення: 27.06.2025).
67. Advanced Analytics and Machine Learning in Fraud Detection// FICO Blogs. URL:https://www.fico.com/blogs/advanced-analytics-machine-learning-fraud-detection (дата звернення: 27.06.2025).
68. FICO Machine Learning Algorithms Improve Card-Not-Present Fraud Detection by 30%// PRNewswire. 01.07.2018. URL:https://www.prnewswire.com/news-releases/fico-machine-learning-algorithms-improve-card-not-present-fraud-detection-by-30-300529629.html (дата звернення: 27.06.2025).
69. Falcon Intelligence Network: A Fraud Consortium for Fraud-Fighting Machine Learning Innovation// FICO. 08.05.2023. URL:https://www.fico.com/blogs/falcon-intelligence-network-fraud-consortium-fraud-fighting-machine-learning-innovation (дата звернення: 27.06.2025).
70. How FICO's Neural Networks Adapt to New Fraud Patterns// FICO Technical Papers. URL:https://www.fico.com/en/resource-center/technical-papers/neural-networks-fraud-adaptation (дата звернення: 27.06.2025).
71. Machine Learning Improves CNP Fraud Detection Rates by 30%// FICO. 02.02.2021. URL:https://www.fico.com/blogs/machine-learning-improves-cnp-fraud-detection-rates-30 (дата звернення: 27.06.2025).
72. The Fraud Consortium: 9,000 Heads Are Better than 1// FICO. 17.10.2024. URL:https://www.fico.com/blogs/fraud-consortium-9-000-heads-are-better-1 (дата звернення: 27.06.2025).

73. FICO® Falcon® Intelligence Network// FICO. URL:https://www.fico.com/en/fico-falcon-intelligence-network (дата звернення: 27.06.2025).
74. FICO Falcon Compromise Manager// FICO. URL:https://www.fico.com/en/products/fico-falcon-compromise-manager (дата звернення: 27.06.2025).
75. Where FICO Gets Its Data for Screening Two-Thirds of All Card Transactions// Machine Learning Times. 12.12.2024. URL:https://www.predictiveanalyticsworld.com/machinelearningtimes/where-fico-gets-its-data-for-screening-two-thirds-of-all-card-transactions/13648/ (дата звернення: 27.06.2025).
76. Global Fraud Intelligence Network Powers Next-Gen Fraud Detection// FICO Blogs. URL:https://www.fico.com/blogs/global-fraud-intelligence-network-powers-next-gen-fraud-detection (дата звернення: 27.06.2025).
77. Combat Financial Crime with AI-Driven AML and Fraud Solutions// NICE Actimize. URL:https://www.niceactimize.com/ (дата звернення: 27.06.2025).
78. The Quest for Customer-friendly Due Diligence// NICE Actimize. 26.06.2014. URL:https://www.niceactimize.com/blog/the-quest-for-customerfriendly-due-diligence-360/ (дата звернення: 27.06.2025).
79. NICE Actimize Launches Customer Due Diligence Suite// NICE Actimize. URL:https://www.niceactimize.com/press-releases/NICE-Actimize-Introduces-Customer-Due-Diligence-Suite-with-IndustryLeading--Risk-Assessment-and-Branch-to-the-Bank-Protections--128/ (дата звернення: 27.06.2025).
80. Xceed Anti-Money Laundering Solutions// NICE Actimize. URL:https://www.niceactimize.com/xceed/aml/ (дата звернення: 27.06.2025).
81. Learning with Graph Representation / Danny Butvinik// NICE Actimize. URL:https://info.nice.com/Data-Science-Blog-learning-with-graph-representation.html (дата звернення: 27.06.2025).
82. NICE Actimize Enhances Intelligent eComms Surveillance Solution with Next Generation Machine Learning and Investigative Tools// NICE Actimize. URL:https://www.niceactimize.com/press-releases/NICE-Actimize-Enhances-Intelligent-eComms-Surveillance-Solution--with-Next-Generation-Machine-Learning-and-Investigative-Tools-262/ (дата звернення: 27.06.2025).
83. NICE Actimize Awarded "Best Compliance RegTech Global" by Capital Finance International for Second Consecutive Year// NICE Actimize. URL:https://www.niceactimize.com/press-releases/NICE-Actimize-Awarded-Best-Compliance-RegTech-Global-by-Capital-Finance-International-for-Second-Consecutive-Year-302/ (дата звернення: 27.06.2025).

84. Unlocking the potential of natural language processing: a comprehensive exploration// NiCE. URL:https://www.nice.com/info/unlocking-the-potential-of-natural-language-processing-a-comprehensive-exploration (дата звернення: 27.06.2025).
85. AI-Native Fraud & Financial Crime Prevention Platform// Feedzai. URL:https://www.feedzai.com/ (дата звернення: 27.06.2025).
86. Advanced AI for Fraud Prevention and AML// Feedzai. URL:https://www.feedzai.com/solutions/ai/ (дата звернення: 27.06.2025).
87. What Is Fraud Detection for Machine Learning// Feedzai. 25.07.2025. URL:https://www.feedzai.com/blog/what-is-fraud-detection-for-machine-learning/ (дата звернення: 27.06.2025).
88. Feedzai AutoML A ML Platform for Fraud Prevention that's 50 Times Faster!// Analytics Vidhya. 2019. 7 травня. URL:https://www.analyticsvidhya.com/blog/2018/08/feedzai-automl-a-ml-platform-for-fraud-prevention-thats-50-times-faster/. (дата звернення: 27.06.2025).
89. Feedzai Unveils AutoML that Fights Fraud in Record Time// Feedzai. 2023. 20 липня. URL:https://feedzai.com/pressrelease/feedzai-unveils-automl-automated-machine-learning-that-fights-fraud-in-a-fraction-of-the-time/. (дата звернення: 27.06.2025).
90. Automated Machine Learning (AutoML)// Feedzai. 2025. 25 липня. URL:https://www.feedzai.com/blog/what-is-fraud-detection-for-machine-learning/. (дата звернення: 27.06.2025).
91. Latency in Machine Learning// Feedzai. 2024. 30 вересня. URL:https://www.feedzai.com/blog/latency-in-machine-learning-what-fraud-prevention-leaders-need-to-know/. (дата звернення: 27.06.2025).
92. Feedzai IQ™ AI Fraud Prevention & Privacy-Preserving Intelligence// Feedzai. 2025. 21 липня. URL:https://www.feedzai.com/pressrelease/feedzai-iq-ai-fraud-prevention-intelligence/. (дата звернення: 27.06.2025).
93. What is CRISP DM?// Data Science PM. 2024. 9 грудня. URL:https://www.datascience-pm.com/crisp-dm-2/. (дата звернення: 25.06.2025.
94. KDD Process in Databases// GeeksforGeeks. 2025. 28 січня. URL:https://www.geeksforgeeks.org/dbms/kdd-process-in-data-mining/. (дата звернення: 25.06.2025).
95. What is SEMMA?// Data Science PM. 2024. 16 листопада. URL:https://www.datascience-pm.com/semma/. (дата звернення: 25.06.2025).
96. Chumbar S. SEMMA Methodology in Data Mining: A Deep Dive// Medium. 2023. 22 вересня. URL:https://medium.com/@shawn.chumbar/semma-

methodology-in-data-mining-a-deep-dive-c8874cb4239. (дата звернення: 25.06.2025).
97. Ajmera A. Data Mining using SEMMA process// Medium. 2023. - 30 вересня. URL: https://medium.com/@ajmeraavi/data-mining-using-semma-a24782b41c2c. (дата звернення: 25.06.2025).
98. Bank Transaction Fraud Detection// Kaggle. URL: https://www.kaggle.com/datasets/marusagar/bank-transaction-fraud-detection. (дата звернення: 25.06.2025).

ДОДАТКИ

Додаток А

Лістинг 2.1. Завантаження даних та бібліотек

```
# Importing the required libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
# Loading the dataset
df = pd.read_csv(
    '/kaggle/input/bank-transaction-fraud-detection/Bank_Transaction_Fraud_Detection.csv',
    parse_dates = ['Transaction_Date'],    # Converting 'Transaction_Date' to datetime
    dayfirst = True     # Ensuring the date is parsed as dd-mm-yyyy
)
df['Transaction_Time'] = pd.to_timedelta(df['Transaction_Time'])
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from datetime import datetime
import warnings
warnings.filterwarnings('ignore')
```

Додаток Б

Лістинг 2.2. Аналіз типів даних та їх структури

```
print(f"Dataset Shape: {df.shape[0]:,} rows × {df.shape[1]} columns")
print(f"Memory Usage: {df.memory_usage(deep=True).sum() / 1024**2:.2f} MB")
print("\nColumn Names and Data Types")
print("-" * 50)
print("\nColumn Information:")
 6. for i, (col, dtype) in enumerate(zip(df.columns, df.dtypes), 1):
print(f"{i:2d}. {col:<25} : {dtype}")
print("\nFirst 5 Rows - Sample Data")
print("-" * 50)
print("data structure and values:")
print(df.head())
14. print("\nDataset Summary Statistics")
15. print("-" * 50)
print("Statistical summary provides insights into data distribution and potential outliers.")
print(df.describe(include='all'))
print("\n🔍 2.1 Missing Values Analysis")
print("-" * 50)
print("Missing values can significantly impact model performance and analysis validity.")
missing_data = pd.DataFrame({
    'Column': df.columns,
    'Missing_Count': df.isnull().sum(),
    'Missing_Percentage': (df.isnull().sum() / len(df)) * 100,
    'Data_Type': df.dtypes
})
missing_data = missing_data.sort_values('Missing_Count', ascending=False)
print("\nMissing Values Summary:")
print(missing_data)
if missing_data['Missing_Count'].sum() == 0:
    print("\nNo missing values detected in the dataset.")
else:
    print(f"\n  Total missing values: {missing_data['Missing_Count'].sum():,}")
print("\n 2.2 Duplicate Records Analysis")
print("-" * 50)

duplicate_count = df.duplicated().sum()
print(f"Number of duplicate rows: {duplicate_count:,}")

if duplicate_count > 0:
    print(f"Percentage of duplicates: {(duplicate_count / len(df)) * 100:.2f}%")
    print("⚠  Consider removing duplicates before modeling.")
else:
    print("No duplicate rows found.")

```

Додаток В

Лістинг 2.3. Статистичний аналіз даних

```
print("\nFraud Distribution Analysis")
print("-" * 50)
print("Understanding the class distribution is crucial for fraud detection models.")
print("Fraud detection typically involves highly imbalanced datasets.")

fraud_counts = df['Is_Fraud'].value_counts().sort_index()
fraud_percentages = df['Is_Fraud'].value_counts(normalize=True).sort_index() * 100

print(f"\nFraud Distribution:")
print(f"Non-Fraudulent Transactions (0): {fraud_counts[0]:,} ({fraud_percentages[0]:.2f}%)")
print(f"Fraudulent Transactions (1):     {fraud_counts[1]:,} ({fraud_percentages[1]:.2f}%)")
print(f"\nFraud Rate: {fraud_percentages[1]:.3f}%")

Visualize fraud distribution
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(15, 6))

# Pie chart
colors = ['lightblue', 'salmon']
ax1.pie(fraud_counts.values, labels=['Non-Fraud', 'Fraud'], autopct='%1.2f%%',
        colors=colors, startangle=90)
ax1.set_title('Fraud vs Non-Fraud Distribution', fontsize=14, fontweight='bold')

# Bar chart
ax2.bar(['Non-Fraud', 'Fraud'], fraud_counts.values, color=colors, alpha=0.8)
ax2.set_ylabel('Number of Transactions')
ax2.set_title('Transaction Count by Fraud Status', fontsize=14, fontweight='bold')
for i, v in enumerate(fraud_counts.values):
    ax2.text(i, v + max(fraud_counts.values) * 0.01, f'{v:,}', ha='center', fontweight='bold')

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

Додаток Г

Лістинг 2.4. Аналіз числових змінних

```
print("\nKey Numerical Variables Overview")
print("-" * 50)
# Identify numerical columns
numerical_cols = df.select_dtypes(include=[np.number]).columns.tolist()
key_numerical_cols = ['Age', 'Transaction_Amount', 'Account_Balance']

print(f"Numerical columns in dataset: {numerical_cols}")
print(f"Key variables for analysis: {key_numerical_cols}")

# Statistical summary by fraud status
print("\Statistical Comparison: Fraud vs Non-Fraud")
print("-" * 50)

for col in key_numerical_cols:
    if col in df.columns:
        print(f"\n{col.upper()} ANALYSIS:")
        print("-" * 30)

        stats_by_fraud = df.groupby('Is_Fraud')[col].describe()
        print(stats_by_fraud)

        # Calculate key insights
        fraud_mean = df[df['Is_Fraud'] == 1][col].mean()
        non_fraud_mean = df[df['Is_Fraud'] == 0][col].mean()
        difference_pct = ((fraud_mean - non_fraud_mean) / non_fraud_mean) * 100


        print(f"   • Average for Fraud: {fraud_mean:,.2f}")
        print(f"   • Average for Non-Fraud: {non_fraud_mean:,.2f}")
        print(f"   • Difference: {difference_pct:+.1f}%")

        if abs(difference_pct) > 20:
            print(f"DIFFERENCE: {col} shows strong fraud indicator potential!")

# Visualize distributions
print("\nDistribution Visualizations")
print("-" * 50)

fig, axes = plt.subplots(2, 3, figsize=(18, 12))
fig.suptitle('Numerical Variables Analysis: Fraud vs Non-Fraud', fontsize=16, fontweight='bold')

```

Продовження додатку Г

```
for i, col in enumerate(key_numerical_cols):
    if col in df.columns and i < 3:
        # Distribution comparison
        ax1 = axes[0, i]
        df[df['Is_Fraud'] == 0][col].hist(bins=50, alpha=0.7, label='Non-Fraud',
                                          color='lightblue', ax=ax1, density=True)
        df[df['Is_Fraud'] == 1][col].hist(bins=50, alpha=0.7, label='Fraud',
                                          color='salmon', ax=ax1, density=True)
        ax1.set_title(f'{col} Distribution by Fraud Status')
        ax1.set_xlabel(col)
        ax1.set_ylabel('Density')
        ax1.legend()

        # Box plot comparison
        ax2 = axes[1, i]
        df.boxplot(column=col, by='Is_Fraud', ax=ax2)
        ax2.set_title(f'{col} Box Plot by Fraud Status')
        ax2.set_xlabel('Is_Fraud')
        ax2.set_ylabel(col)
        plt.setp(ax2, title=f'{col} Box Plot by Fraud Status')

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

Додаток Д

Лістинг 2.5. Аналіз категоріальних даних

```
categorical_cols = ['Gender', 'Account_Type', 'Transaction_Type', 'Merchant_Category',
                    'Transaction_Device', 'Device_Type', 'Transaction_Currency']

fraud_insights = {}

for col in categorical_cols:
    if col in df.columns:
        print(f"\n {col.upper()} FRAUD ANALYSIS:")
        print("-" * 40)

        fraud_analysis = df.groupby(col).agg({
            'Is_Fraud': ['count', 'sum', 'mean']
        }).round(4)

        fraud_analysis.columns = ['Total_Transactions', 'Fraud_Count', 'Fraud_Rate']
        fraud_analysis = fraud_analysis.sort_values('Fraud_Rate', ascending=False)

        print(fraud_analysis)

        # Store insights for later use
        highest_risk = fraud_analysis.index[0]
        highest_rate = fraud_analysis.iloc[0]['Fraud_Rate']
        fraud_insights[col] = {
            'highest_risk_category': highest_risk,
            'highest_fraud_rate': highest_rate
        }

        print(f"\nHighest risk {col.lower()}: '{highest_risk}' ({highest_rate:.1%} fraud rate)")

        # Flag high-risk categories
        high_risk_threshold = df['Is_Fraud'].mean() * 2  # 2x average fraud rate
        high_risk_categories = fraud_analysis[fraud_analysis['Fraud_Rate'] > high_risk_threshold]

```

Продовження додатку Д

```
        if not high_risk_categories.empty:
            print(f"HIGH-RISK CATEGORIES (>{high_risk_threshold:.1%} fraud rate):")
            for category, row in high_risk_categories.iterrows():
                print(f"   • {category}: {row['Fraud_Rate']:.1%}")

print("\nCategorical Variables Visualization")
print("-" * 50)

# Create visualizations for key categorical variables
fig, axes = plt.subplots(2, 2, figsize=(16, 12))
fig.suptitle('Fraud Rate by Categorical Variables', fontsize=16, fontweight='bold')

plot_cols = ['Transaction_Type', 'Account_Type', 'Transaction_Device', 'Device_Type']
colors = ['coral', 'lightblue', 'lightgreen', 'gold']

for i, col in enumerate(plot_cols):
    if col in df.columns and i < 4:
        ax = axes[i//2, i%2]
        fraud_rates = df.groupby(col)['Is_Fraud'].mean().sort_values(ascending=True)

        bars = fraud_rates.plot(kind='barh', ax=ax, color=colors[i], alpha=0.8)
        ax.set_title(f'Fraud Rate by {col}', fontweight='bold')
        ax.set_xlabel('Fraud Rate')

        # Add value labels on bars
        for j, bar in enumerate(ax.patches):
            width = bar.get_width()
            ax.text(width + 0.001, bar.get_y() + bar.get_height()/2,
                    f'{width:.1%}', ha='left', va='center', fontweight='bold')

plt.tight_layout()
plt.show()

```

Додаток Е

Лістинг 2.6. Часовий аналіз шахрайських операцій

```
print("\nTime-Based Fraud Patterns")
print("-" * 50)
print("Temporal patterns often reveal when fraudulent activities are most likely to occur.")

# Process datetime columns
if 'Transaction_Date' in df.columns:
    try:
        df['Transaction_Date'] = pd.to_datetime(df['Transaction_Date'])
        df['Day_of_Week'] = df['Transaction_Date'].dt.day_name()
        df['Month'] = df['Transaction_Date'].dt.month_name()
        df['Day_of_Month'] = df['Transaction_Date'].dt.day




if 'Transaction_Time' in df.columns:
    try:
        df['Hour'] = pd.to_datetime(df['Transaction_Time'], format='%H:%M:%S').dt.hour
        df['Time_Period'] = df['Hour'].apply(lambda x:
            'Night (0-6)' if 0 <= x <= 6 else
            'Morning (7-12)' if 7 <= x <= 12 else
            'Afternoon (13-18)' if 13 <= x <= 18 else
            'Evening (19-23)')

        print(" Successfully processed Transaction_Time")
    except:
        print(" ⚠  Could not process Transaction_Time column")

# Analyze temporal patterns
temporal_cols = ['Day_of_Week', 'Month', 'Hour', 'Time_Period']

for col in temporal_cols:
    if col in df.columns:
        print(f"\nFRAUD ANALYSIS BY {col.upper()}:")
        print("-" * 40)

        temporal_analysis = df.groupby(col).agg({
            'Is_Fraud': ['count', 'sum', 'mean']
        }).round(4)

        temporal_analysis.columns = ['Total_Transactions', 'Fraud_Count', 'Fraud_Rate']

```

Продовження додатку Е

```
.       if col in ['Hour']:
45.             temporal_analysis = temporal_analysis.sort_index()
46.         else:
47.             temporal_analysis = temporal_analysis.sort_values('Fraud_Rate', ascending=False)
48.
49.         print(temporal_analysis)
50.
51.         # Identify peak fraud times
52.         peak_fraud = temporal_analysis['Fraud_Rate'].max()
53.         peak_time = temporal_analysis['Fraud_Rate'].idxmax()
54.
55.         print(f"\nPEAK FRAUD TIME: {peak_time} ({peak_fraud:.1%} fraud rate)")
56.
57. # Visualize temporal patterns
58. if 'Hour' in df.columns:
59.     print("\nHourly Fraud Pattern Visualization")
60.     print("-" * 50)
61.
62.     fig, (ax1, ax2) = plt.subplots(2, 1, figsize=(14, 10))
63.
64.     # Hourly fraud rate
65.     hourly_fraud = df.groupby('Hour')['Is_Fraud'].mean()
66.     ax1.plot(hourly_fraud.index, hourly_fraud.values, marker='o', linewidth=2, markersize=8,
color='red')
67.     ax1.set_title('Fraud Rate by Hour of Day', fontsize=14, fontweight='bold')
68.     ax1.set_xlabel('Hour of Day')
69.     ax1.set_ylabel('Fraud Rate')
70.     ax1.grid(True, alpha=0.3)
71.     ax1.set_xticks(range(0, 24, 2))
72.
73.     # Transaction volume by hour
74.     hourly_volume = df.groupby(['Hour', 'Is_Fraud']).size().unstack(fill_value=0)
75.     hourly_volume.plot(kind='bar', stacked=True, ax=ax2, color=['lightblue', 'salmon'],
alpha=0.8)
76.     ax2.set_title('Transaction Volume by Hour (Fraud vs Non-Fraud)', fontsize=14,
fontweight='bold')
77.     ax2.set_xlabel('Hour of Day')
78.     ax2.set_ylabel('Number of Transactions')
79.     ax2.legend(['Non-Fraud', 'Fraud'])
80.     ax2.tick_params(axis='x', rotation=0)
81.
82.     plt.tight_layout()
83.     plt.show()
```

Додаток Є

Лістинг 2.6. Кореляційний аналіз

```
print("\nCorrelation Matrix of Numerical Variables")
print("-" * 50)
print("Correlation analysis helps identify relationships between variables and multicollinearity issues.")

# Calculate correlation matrix for numerical variables
numerical_df = df.select_dtypes(include=[np.number])
correlation_matrix = numerical_df.corr()

print(f"Correlation matrix shape: {correlation_matrix.shape}")

# Visualize correlation matrix
plt.figure(figsize=(12, 10))
mask = np.triu(np.ones_like(correlation_matrix, dtype=bool))  # Only show lower triangle
sns.heatmap(correlation_matrix, mask=mask, annot=True, cmap='coolwarm', center=0,
            square=True, linewidths=0.5, cbar_kws={"shrink": .8})
plt.title('Correlation Matrix of Numerical Variables', fontsize=16, fontweight='bold')
plt.tight_layout()
plt.show()

print("\nCorrelation with Fraud Target Variable")
print("-" * 50)

# Focus on correlations with Is_Fraud
fraud_correlations = correlation_matrix['Is_Fraud'].drop('Is_Fraud').sort_values(key=abs, ascending=False)

print("Variables ranked by correlation strength with fraud:")
print("\nVariable                  Correlation")
print("-" * 40)
for var, corr in fraud_correlations.items():
    if abs(corr) > 0.01:  # Only show meaningful correlations
        direction = "Positive" if corr > 0 else "Negative"
        strength = "Strong" if abs(corr) > 0.5 else "Moderate" if abs(corr) > 0.3 else "Weak"
        print(f"{var:<25} {corr:>8.3f} ({direction}, {strength})")

# Identify strong correlations
strong_correlations = fraud_correlations[abs(fraud_correlations) > 0.3]
if not strong_correlations.empty:
    print(f"\nSTRONG FRAUD INDICATORS FOUND:")
    for var, corr in strong_correlations.items():
        print(f"  • {var}: {corr:.3f}")
else:
    print(f"\nNo strong linear correlations found with fraud target.")
    print("  This suggests fraud patterns may be more complex and non-linear.")
```

Додаток Ж

Лістинг 2.7. Аналіз ризиків

```
print("\nTransaction Amount Analysis")
print("-" * 50)
print("Transaction amount is often a key indicator of fraudulent activity.")

# Define amount categories
df['Amount_Category'] = pd.cut(df['Transaction_Amount'],
                               bins=[0, 100, 500, 1000, 5000, np.inf],
                               labels=['Very Low ($0-$100)', 'Low ($100-$500)',
                                    'Medium ($500-$1K)', 'High ($1K-$5K)',
                                     'Very High ($5K+)'])

amount_fraud_analysis = df.groupby('Amount_Category').agg({
    'Is_Fraud': ['count', 'sum', 'mean']
}).round(4)
amount_fraud_analysis.columns = ['Total_Transactions', 'Fraud_Count', 'Fraud_Rate']

print("Fraud Rate by Transaction Amount Category:")
print(amount_fraud_analysis)

# Statistical tests
fraud_amounts = df[df['Is_Fraud'] == 1]['Transaction_Amount']
normal_amounts = df[df['Is_Fraud'] == 0]['Transaction_Amount']

print(f"\nSTATISTICAL COMPARISON:")
print(f"Fraud transactions - Mean: ${fraud_amounts.mean():.2f}, Median:
${fraud_amounts.median():.2f}")
print(f"Normal transactions - Mean: ${normal_amounts.mean():.2f}, Median:
${normal_amounts.median():.2f}")
print(f"Ratio (Fraud/Normal): {fraud_amounts.mean() / normal_amounts.mean():.2f}x")

print("\nMulti-Variable Risk Scoring")
print("-" * 50)
print("Combining multiple risk factors to identify high-risk transaction profiles.")

# Create risk factors
risk_factors = {}

# High transaction amount risk
high_amount_threshold = df['Transaction_Amount'].quantile(0.9)
risk_factors['High_Amount'] = df['Transaction_Amount'] > high_amount_threshold

# Night time risk (if hour available)
if 'Hour' in df.columns:
    risk_factors['Night_Time'] = (df['Hour'] <= 6) | (df['Hour'] >= 22)

```

Продовження додатку Ж

```
# Weekend risk (if day available)
if 'Day_of_Week' in df.columns:
    risk_factors['Weekend'] = df['Day_of_Week'].isin(['Saturday', 'Sunday'])

# High-risk transaction types
if 'Transaction_Type' in df.columns:
    high_risk_types = df.groupby('Transaction_Type')['Is_Fraud'].mean().nlargest(2).index
    risk_factors['High_Risk_Type'] = df['Transaction_Type'].isin(high_risk_types)

# Calculate risk score
df['Risk_Score'] = sum(risk_factors.values())

print("Risk Score Distribution:")
risk_score_analysis = df.groupby('Risk_Score').agg({
    'Is_Fraud': ['count', 'sum', 'mean']
}).round(4)
risk_score_analysis.columns = ['Total_Transactions', 'Fraud_Count', 'Fraud_Rate']
print(risk_score_analysis)

# Visualize risk score effectiveness
plt.figure(figsize=(12, 6))
risk_score_fraud_rate = df.groupby('Risk_Score')['Is_Fraud'].mean()
bars = plt.bar(risk_score_fraud_rate.index, risk_score_fraud_rate.values,
               color='salmon', alpha=0.8, edgecolor='darkred')
plt.title('Fraud Rate by Risk Score', fontsize=14, fontweight='bold')
plt.xlabel('Risk Score (Number of Risk Factors)')
plt.ylabel('Fraud Rate')

# Add value labels on bars
for bar in bars:
    height = bar.get_height()
    plt.text(bar.get_x() + bar.get_width()/2., height + 0.001,
             f'{height:.1%}', ha='center', va='bottom', fontweight='bold')

plt.tight_layout()
plt.show()

```

Додаток З

Лістинг 2.8. Процес балансування шляхом андерсемплінгу

```
# Розділяємо легальні та шахрайські транзакції
fraud = df[df['Is_Fraud'] == 1]
non_fraud = df[df['Is_Fraud'] == 0]

print("Кількість шахрайських:", len(fraud))
print("Кількість легальних:", len(non_fraud))

# Виконуємо андерсемплінг легальних (випадкова підмножина)
non_fraud_sampled = non_fraud.sample(n=len(fraud), random_state=42)

# Об’єднуємо назад
df_balanced = pd.concat([fraud, non_fraud_sampled], axis=0)

# Перемішуємо дані
df_balanced = df_balanced.sample(frac=1,
random_state=42).reset_index(drop=True)

print("Збалансований розподіл:")
print(df_balanced['Is_Fraud'].value_counts())
```

Додаток И

Лістинг 2.9. Створення часових ознак

```
1. df['Transaction_Date'] = pd.to_datetime(df['Transaction_Date'])
2. df['Transaction_Hour'] = df['Transaction_Time'].dt.seconds // 3600
3. df['Day_of_Week'] = df['Transaction_Date'].dt.day_name()
4. df['is_weekend'] = df['Day_of_Week'].isin(['Saturday','Sunday']).astype(int)
5. df = df.sort_values(['Customer_ID','Transaction_Date','Transaction_Time'])
```

Додаток К

Лістинг 2.10. Розрахунок поведінкових метрик клієнтів

```
1. df['time_diff_prev_txn'] =
df.groupby('Customer_ID')['Transaction_Date'].diff().dt.total_seconds().fillna(0)
2. df['Amount_to_Balance_Ratio'] = df['Transaction_Amount'] /
(df['Account_Balance'] + 1)
3. df['is_large_txn'] = (df['Transaction_Amount'] >
df['Transaction_Amount'].quantile(0.95)).astype(int)
```

Додаток Л

Лістинг 2.11. Впровадження системи ковзаючих статистик

```
rolling_window = 5
df['rolling_mean'] = df.groupby('Customer_ID')['Transaction_Amount'].rolling(
    rolling_window, min_periods=1).mean().reset_index(level=0, drop=True)

df['rolling_std'] = df.groupby('Customer_ID')['Transaction_Amount'].rolling(
    rolling_window, min_periods=1).std().reset_index(level=0,
drop=True).fillna(0)

df['rolling_large_txn_ratio'] = df.groupby('Customer_ID')['is_large_txn'].rolling(
    rolling_window, min_periods=1).mean().reset_index(level=0, drop=True)
```

Додаток М

Лістинг 2.12. Агрегація активності за часовими періодами

```
1. df['txn_count_day'] = df.groupby(['Customer_ID',
df['Transaction_Date'].dt.date])['Transaction_ID'].transform('count')
2. df['txn_count_hour'] = df.groupby(['Customer_ID',
df['Transaction_Hour']])['Transaction_ID'].transform('count')
```

Додаток Н

Лістинг 2.13. Створення складних взаємодіючих ознак

```
1. df['ratio_times_large'] = df['Amount_to_Balance_Ratio'] * df['is_large_txn']
2. df['hour_times_weekend'] = df['Transaction_Hour'] * df['is_weekend']
3. df['amount_dev_from_mean'] = df['Transaction_Amount'] - df['rolling_mean']
4. df['branch_merchant_type'] = df['Bank_Branch'] + '_' + df['Merchant_Category'] +
'_' + df['Transaction_Type']
5.
```

Додаток О

Лістинг 2.14. Структуризація ознакового простору

```
numerical_cols = [
    'Age','Transaction_Amount','Account_Balance','Amount_to_Balance_Ratio',
    'time_diff_prev_txn','rolling_mean','rolling_std','rolling_large_txn_ratio',
    'txn_count_day','txn_count_hour','ratio_times_large','hour_times_weekend',
    'amount_dev_from_mean','is_large_txn','is_weekend'
]

categorical_cols = [
    'Gender','State','City','Bank_Branch','Account_Type',
     'Transaction_Type','Merchant_Category','Transaction_Device',
     'Transaction_Location','Device_Type','Transaction_Currency',
     'branch_merchant_type'
]
```

Додаток П

Лістинг 2.15. Target Encoding для категоріальних змінних

```
for col in categorical_cols:
    te = TargetEncoder(cols=[col])
    df[col] = te.fit_transform(df[col], df['Is_Fraud'])

X = df[numerical_cols + categorical_cols]
y = df['Is_Fraud']

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, stratify=y, random_state=42
)
```

Додаток Р

Лістинг 2.16. Побудова моделей

```
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

lgb_train = lgb.Dataset(X_train, y_train)
lgb_eval = lgb.Dataset(X_test, y_test, reference=lgb_train)
params = {
    'objective': 'binary', 'metric': 'auc', 'boosting_type': 'gbdt',
    'learning_rate': 0.05, 'num_leaves': 64, 'max_depth': 8,
    'min_data_in_leaf': 20, 'feature_fraction': 0.8, 'bagging_fraction': 0.8,
    'bagging_freq': 5, 'lambda_l1': 1, 'lambda_l2': 1, 'is_unbalance': True, 'seed': 42
}
gbm = lgb.train(
    params, lgb_train, num_boost_round=1000, valid_sets=[lgb_train, lgb_eval],
    callbacks=[lgb.early_stopping(stopping_rounds=50),
lgb.log_evaluation(period=100)]
)
y_proba_lgb = gbm.predict(X_test, num_iteration=gbm.best_iteration)

rf = RandomForestClassifier(n_estimators=300, max_depth=10,
random_state=42, class_weight='balanced')
rf.fit(X_train, y_train)
y_proba_rf = rf.predict_proba(X_test)[:,1]

knn = KNeighborsClassifier(n_neighbors=15, n_jobs=-1)
knn.fit(X_train_scaled, y_train)
y_proba_knn = knn.predict_proba(X_test_scaled)[:,1]

mlp = MLPClassifier(hidden_layer_sizes=(128,64), activation='relu',
solver='adam',
                    batch_size=256, max_iter=50, random_state=42)
mlp.fit(X_train_scaled, y_train)
y_proba_mlp = mlp.predict_proba(X_test_scaled)[:,1]
```

Додаток С

Лістинг 2.17. Оцінка моделей

```
def evaluate_model(y_test, y_proba, model_name="Model"):
    thresholds = np.linspace(0.1, 0.9, 81)
    f1_scores = [f1_score(y_test, (y_proba >= t).astype(int)) for t in thresholds]
    best_idx = np.argmax(f1_scores)
    best_threshold = thresholds[best_idx]
    y_pred = (y_proba >= best_threshold).astype(int)
    print(f"\n{model_name} - Best F1 threshold: {best_threshold:.2f}")
    print(classification_report(y_test, y_pred))
    print(f"ROC-AUC: {roc_auc_score(y_test, y_proba):.4f}")

evaluate_model(y_test, y_proba_lgb, "LightGBM")
evaluate_model(y_test, y_proba_rf, "Random Forest")
evaluate_model(y_test, y_proba_knn, "KNN")
evaluate_model(y_test, y_proba_mlp, "Neural Network")
model_results = {
    "LightGBM": y_proba_lgb,
    "Random Forest": y_proba_rf,
    "KNN": y_proba_knn,
    "Neural Network": y_proba_mlp
}

plt.figure(figsize=(8,6))
for name, y_proba in model_results.items():
    fpr, tpr, _ = roc_curve(y_test, y_proba)
    auc = roc_auc_score(y_test, y_proba)
    plt.plot(fpr, tpr, label=f"{name} (AUC={auc:.3f})")
plt.plot([0,1], [0,1], 'k--')
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("ROC Curves")
plt.legend()
plt.show()

plt.figure(figsize=(8,6))
for name, y_proba in model_results.items():
    prec, rec, _ = precision_recall_curve(y_test, y_proba)
    plt.plot(rec, prec, label=name)
plt.xlabel("Recall")
plt.ylabel("Precision")
plt.title("Precision-Recall Curves")
plt.legend()
```

```
plt.show()

plt.figure(figsize=(10,6))
thresholds = np.linspace(0.1, 0.9, 81)
for name, y_proba in model_results.items():
    f1_scores = [f1_score(y_test, (y_proba >= t).astype(int)) for t in thresholds]
    plt.plot(thresholds, f1_scores, label=name)
plt.xlabel("Threshold")
plt.ylabel("F1 Score")
plt.title("F1 Score vs Threshold")
plt.legend()
plt.show()

def plot_conf_matrix(y_test, y_proba, model_name):
    thresholds = np.linspace(0.1, 0.9, 81)
    f1_scores = [f1_score(y_test, (y_proba >= t).astype(int)) for t in thresholds]
    best_threshold = thresholds[np.argmax(f1_scores)]
    y_pred = (y_proba >= best_threshold).astype(int)

    cm = confusion_matrix(y_test, y_pred)
    plt.figure(figsize=(5,4))
    sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", cbar=False)
    plt.title(f"{model_name} Confusion Matrix (thr={best_threshold:.2f})")
    plt.xlabel("Predicted")
    plt.ylabel("Actual")
    plt.show()

for name, y_proba in model_results.items():
    plot_conf_matrix(y_test, y_proba, name)

def plot_feature_importance(model, X, title):
    if hasattr(model, "feature_importances_"):
        fi = pd.Series(model.feature_importances_,
index=X.columns).sort_values(ascending=False)
        plt.figure(figsize=(10,6))
        sns.barplot(x=fi.values[:20], y=fi.index[:20])
        plt.title(title)
        plt.show()

plot_feature_importance(rf, X_train, "Random Forest Feature Importance")
plot_feature_importance(gbm, X_train, "LightGBM Feature Importance")
```