МІНІСЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет *інформаційних технологій та електроніки*
Кафедра *інформаційних технологій та програмування*

КВАЛІФІКАЦІЙНА РОБОТА

Магістра

На тему : *Розробка адаптивної моделі виявлення шахрайських транзакцій у режимі реального часу на основі глибоких нейронних мереж та аналізу Big Data.*

Виконав: студент 2 курсу, групи_ІСТ-24зм
126 «Інформаційні системи та технології
Надтока. А.А.
Керівник _*д.т.н.,проф. Захожаі.О.І*_
Рецензент_*д.т.н.,проф. Меняйленко.О.С*

Київ-2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ.
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОДИМИРА ДАЛЯ

Факультет *інформаційних технологій та електроніки*

Кафедра *інформаційних технологій та програмування*

Освітній ступень _____*магістр*_____

Спеціальність *126 Інформаційні системи та технології*

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____*д.т.н.,проф.,Захожаі.О.І*

«___» _____ 2025року

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

Надтоки Аркадія Анатолійовича

1.Тема роботи: *Розробка адаптивної моделі виявлення шахрайських транзакцій у режимі реального часу на основі глибоких нейронних мереж та аналізу Big Data.*

керівник роботи *д.т.н..проф.Захожай О.І.*

затверджені наказом по університету від «28» листопада 2025р. № 241/17.03

2.Строк подання студентом роботи до захисту 20 грудня 2025 року

3.Вихідні данні *Матеріали науково-дослідної практики, науково-методична література; дані інтернет-мережі;*

4.Зміст розрахунково-пояснювальної записки(перелік питань, які потрібно розробити) 4.1.1*Вступ.*4.1.2*Актуальність теми.*4.1.3*Мета завдання дослідження.*4.1.4*Об'єкт і предмет дослідження.*4.1.5*Методи дослідження.*4.1.6*Наукова новизна.*4.1.7*Практичне значення.*4.2*Теорітичні основи виявлення шахрайства в банківських системах.*4.3*Тепхнологічні засади Big Data та машинного навчання*.4.4*Розробка алгоритму для виявлення шахрайських*

*транзакцій.*4.5*Висновки.*4.6*Список використаних джерел.*4.7*Додатки.*4.7.1*Фрагмент програмного коду.*4.7.2*Приклади шахрайських транзакцій.*

5.Перелік графічного матеріалу(з точним значенням обов'язкових креслень)

6.Консультанти розділів

Розділ	Прізвище,ініціали та посада консультана	Підпис,дата	
		завд. видав	завд.прийняв

7.Дата видачі завдання «10» листопада 2025року

КАЛЕНДАРНИЙ ПЛАН

№п/п	назва етапів кваліфікаційнї роботи	строк виконання роботи	примітка
	Аналіз предметної галузі	10.11.2025-14.11.2025	
	Аналіз літератури з досліджуваної проблеми	15.11.2025-19.11.2025	
	Аналіз технічних засобів	20.11.2025-24.112025	
	Аналіз існуючих систем	25.112025-30.11.2025	
	Написання, оформлення та узгодження пояснювальної записки з керівником	01.12.2025-11.12.2025	
	Оформлення пояснювальной записки та презентації	12.12.2025-19.12.2025	

Здобувач _____ *Надтока А.А*

Керівник _____ *Захожай О.І*

РЕФЕРАТ

Робота містить 83 сторінок основного тексту, 8 сторінок додатку, 7 рисунків, 4 таблиці, 31 посилання на джерела.

У даній магістерській роботі розглядається проблема виявлення шахрайських транзакцій у банківських системах, то мета спрямована на створення та обґрунтування алгоритмічного підходу, який забезпечуватиме високу ефективність і точність виявлення підозрілих фінансових операцій в умовах динамічного зростання обсягів даних.

У цьому контексті використання технологій Big Data та машинного навчання відкриває нові можливості, адже дозволяє виявляти складні нелінійні закономірності у великих масивах інформації, які недоступні для традиційних методів аналізу.

У процесі дослідження було розглянуто ключові аспекти, що стосуються поняття та видів фінансового шахрайства, класифікації банківських транзакцій і ризиків, методів виявлення шахрайських операцій у традиційних банківських системах, а також сучасних підходів до виявлення аномалій у фінансових процесах. Проведений аналіз дозволяє зробити низку узагальнень, які мають як теоретичне, так і практичне значення.

ABSTRACT

The work contains 83 pages of the main text, 8 pages of appendix, 7 figures, 4 tables, 31 references to sources.

This master's thesis considers the problem of detecting fraudulent transactions in banking systems, the goal is to create and substantiate an algorithmic approach that will ensure high efficiency and accuracy of detecting suspicious financial transactions in conditions of dynamic growth in data volumes.

In this context, the use of Big Data and machine learning technologies opens up new opportunities, as it allows you to detect complex nonlinear patterns in large amounts of information that are inaccessible to traditional analysis methods.

In the process of research, key aspects related to the concept and types of financial fraud, the classification of banking transactions and risks, methods for detecting fraudulent transactions in traditional banking systems, as well as modern approaches to detecting anomalies in financial processes were considered. The analysis conducted allows you to make a number of generalizations that have both theoretical and practical significance.

ЗМІСТ

ВСТУП

Актуальність теми

У сучасних умовах глобалізації та цифровізації фінансових послуг питання забезпечення безпеки банківських операцій набуває особливого значення. Банківські системи є однією з ключових ланок світової економіки, адже вони забезпечують обслуговування мільярдів користувачів, управління грошовими потоками та підтримку економічної стабільності держав. Разом із тим, активне впровадження електронних платіжних систем, онлайн-банкінгу, мобільних додатків і міжнародних трансакцій призвело до стрімкого зростання кількості фінансових злочинів, зокрема шахрайських операцій.

За даними міжнародних досліджень, щорічні збитки від фінансового шахрайства у світі сягають сотень мільярдів доларів, причому значна їх частина припадає саме на банківський сектор. Шахрайські транзакції можуть проявлятися у вигляді крадіжки даних платіжних карток, фішингових атак, підроблених електронних переказів, використання скомпрометованих облікових записів тощо. Зважаючи на зростаючу складність і різноманітність таких загроз, традиційні методи виявлення шахрайства, засновані на фіксованих правилах і простих статистичних підходах, виявляються недостатньо ефективними. Вони не здатні своєчасно реагувати на нові сценарії атак, які постійно змінюються та удосконалюються.

В умовах стрімкого розвитку цифрової економіки особливої актуальності набувають методи, що дозволяють обробляти великі обсяги транзакційних даних у режимі реального часу. Тут на перший план виходять технології Big Data та машинного навчання, які дають змогу створювати алгоритми адаптивного виявлення аномалій. Використання машинного навчання дозволяє не тільки аналізувати історичні дані, але й навчати моделі розпізнавати нові, невідомі раніше патерни шахрайства. Це відкриває можливість для створення систем, здатних автоматично оновлюватися та вдосконалюватися разом із розвитком загроз.

Особливу роль у цьому процесі відіграє можливість інтеграції різних джерел даних. Сучасні банківські системи генерують терабайти інформації щодня:

- транзакції;
- логфайли;
- поведінкові профілі клієнтів;
- геолокаційні дані;
- дані пристроїв тощо.

Технології Big Data дозволяють збирати, зберігати й аналізувати такі різнорідні масиви інформації, а алгоритми машинного навчання – знаходити в них приховані закономірності. У результаті можна формувати більш точні моделі оцінки ризиків, здатні виявляти навіть складні багаторівневі схеми шахрайства.

Актуальність теми також обумовлюється глобальними тенденціями у сфері кібербезпеки. Згідно з прогнозами експертів, у найближчі роки обсяги фінансових шахрайств будуть лише зростати через збільшення кількості безготівкових операцій, поширення електронних гаманців і криптовалют, розвиток міжбанківських електронних розрахунків. Відповідно, перед фінансовими установами стоїть завдання не лише зберегти конкурентоспроможність, а й забезпечити високий рівень довіри клієнтів. Виявлення шахрайських операцій у режимі реального часу стає ключовим елементом стратегії захисту банківських систем.

У науковому плані проблема є надзвичайно цікавою, адже поєднує кілька перспективних напрямів досліджень: аналіз великих даних, штучний інтелект, кібербезпека, моделювання поведінки користувачів. Синергія цих підходів дозволяє формувати нове покоління інтелектуальних систем фінансової безпеки. Зокрема, застосування ансамблевих алгоритмів (Random Forest, Gradient Boosting), глибинних нейронних мереж, а також методів навчання з підкріпленням відкриває нові можливості для виявлення прихованих взаємозв’язків у даних і прогнозування потенційно небезпечних транзакцій.

Для України ця тема має додаткове практичне значення. Зростання цифровізації банківських послуг, активне впровадження дистанційних каналів

обслуговування клієнтів, інтеграція у міжнародні платіжні системи вимагають від вітчизняних банків впровадження сучасних рішень у сфері безпеки. Використання передових алгоритмів на основі Big Data та машинного навчання дозволить не лише підвищити стійкість фінансового сектору до шахрайства, а й сприятиме його конкурентоспроможності на міжнародному рівні.

Таким чином, актуальність теми роботи визначається:

- необхідністю захисту банківських систем від зростаючих загроз шахрайства;

- недостатньою ефективністю традиційних методів виявлення підозрілих транзакцій;

- потребою у використанні сучасних технологій Big Data для обробки великих масивів інформації;

- високим потенціалом алгоритмів машинного навчання для підвищення точності та швидкості виявлення шахрайства;

- практичною значущістю результатів дослідження для розвитку банківської сфери України та світу.

Усе це підкреслює важливість розробки ефективного алгоритму для виявлення шахрайських транзакцій на основі інтелектуальних методів обробки даних, що становить як наукову, так і прикладну цінність.

Мета і завдання дослідження

Метою магістерської роботи є обґрунтування концептуального алгоритмічного підходу до виявлення шахрайських транзакцій у банківських системах у режимі реального часу на основі аналізу великих масивів даних (Big Data) та методів глибокого навчання, що забезпечує підвищення ефективності та точності ідентифікації підозрілих фінансових операцій в умовах динамічного зростання обсягів транзакційної інформації.

Об’єкт і предмет дослідження

Об’єкт досліджень: інформаційні системи електронних платежів

Предмет дослідження: методи та моделі виявлення шахрайських транзакції в системах електронних платежів.

Методи дослідження

У роботі використано комплекс взаємопов'язаних теоретичних, емпіричних та прикладних методів дослідження, що забезпечують системний аналіз застосування технологій Big Data та штучного інтелекту у фінансовій сфері, зокрема в екосистемі децентралізованих фінансів (DeFi).

Теоретико-методологічну основу дослідження становлять загальнонаукові методи аналізу і синтезу, індукції та дедукції, які застосовувалися для опрацювання наукових публікацій, формування понятійного апарату та узагальнення підходів до використання Big Data й AI у фінансовій аналітиці. Системний підхід використано для розгляду DeFi як складної динамічної системи, що поєднує технологічні, економічні та поведінкові компоненти.

Для дослідження технологічних аспектів обробки великих масивів даних застосовано методи розподіленої та потокової обробки даних, зокрема з використанням екосистем Apache Kafka, Hadoop і Apache Spark. Порівняльний аналіз використано для зіставлення характеристик брокерів повідомлень та оцінки їхньої придатності для високонавантажених фінансових систем.

У межах емпіричної частини дослідження використано економіко-статистичні методи обробки даних, зокрема аналіз пропущених значень, медіанну імпутацію, вінзоризацію на основі міжквартильного розмаху, стандартизацію числових ознак та методи кодування категоріальних змінних (One-Hot Encoding, Target Encoding). Для подолання дисбалансу класів у вибірках застосовано методи SMOTE та Tomek Links.

Для побудови моделей оцінки ризиків та виявлення аномалій використано методи машинного навчання, включаючи алгоритми класифікації, ансамблеві підходи та нейронні мережі глибинного навчання. З метою підвищення адаптивності моделей до змінних ринкових умов розглянуто застосування

гібридних моделей прогнозування та елементів навчання з підкріпленням (reinforcement learning).

Для врахування інформаційних і поведінкових чинників у фінансових процесах застосовано методи обробки природної мови (NLP) та контент-аналіз інформаційних потоків з криптовалютних медіа та соціальних мереж.

Практична перевірка ефективності запропонованих підходів здійснювалася за допомогою обчислювального експерименту, що передбачав тестування різних конфігурацій потокової обробки даних і оцінювання продуктивності систем за показниками затримки та пропускної здатності. Якість моделей оцінювалася шляхом розподілу даних на тренувальну, валідаційну та тестову вибірки, що забезпечило коректність і відтворюваність отриманих результатів.

Наукова новизна

Наукова новизна результатів роботи полягає в такому:

Обґрунтовано концептуальний алгоритмічний підхід до виявлення шахрайських транзакцій у банківських системах у режимі реального часу, що базується на аналізі великих масивів транзакційних даних (Big Data) та застосуванні методів глибокого навчання.

Набули подальшого розвитку підходи до протидії фінансовому шахрайству шляхом обґрунтування доцільності використання гібридної адаптивної моделі, яка поєднує автоенкодери, рекурентні (LSTM) та згорткові (CNN) нейронні мережі для підвищення точності виявлення підозрілих операцій.

Удосконалено підхід до адаптації алгоритмів виявлення шахрайських транзакцій в умовах динамічного зростання обсягів даних шляхом використання поведінкових профілів клієнтів та потокової обробки транзакційної інформації.

Систематизовано напрями застосування технологій Big Data та штучного інтелекту в банківських системах з позицій підвищення ефективності фінансового моніторингу та точності ідентифікації шахрайських операцій.

Практичне значення

Розвиток цифрових банківських послуг супроводжується зростанням кількості шахрайських транзакцій, які набувають дедалі більш витончених форм. Це створює серйозні ризики як для самих фінансових установ, так і для їхніх клієнтів. Тому практичне значення роботи визначається можливістю підвищення ефективності систем фінансової безпеки завдяки впровадженню алгоритму, розробленого з використанням технологій Big Data та машинного навчання.

Значення для банківського сектору

Використання розробленого алгоритму дозволяє банківським установам:

Зменшити кількість шахрайських транзакцій. Алгоритм здатний виявляти підозрілі операції у режимі, наближеному до реального часу, що знижує ризик втрат фінансових ресурсів.

Підвищити рівень довіри клієнтів. Банки, які демонструють високу якість захисту транзакцій, формують позитивну репутацію, що сприяє зростанню клієнтської бази.

Оптимізувати витрати на кібербезпеку. На відміну від ручного аналізу чи статичних правил, інтелектуальні алгоритми автоматизують процес виявлення аномалій, зменшуючи потребу в ручній перевірці.

Адаптуватися до нових загроз. Алгоритм, що базується на машинному навчанні, може навчатися на нових даних та оновлювати свої моделі у відповідь на появу нових шахрайських схем.

Технологічні переваги

Завдяки використанню сучасних інструментів Big Data розроблений алгоритм має низку важливих характеристик:

Масштабованість. Система здатна працювати з мільйонами транзакцій щодня, що особливо важливо для великих комерційних банків і міжнародних платіжних систем;

Висока швидкість обробки. Використання Apache Spark та інших розподілених обчислювальних платформ дозволяє здійснювати аналіз транзакцій практично в реальному часі;

Гнучкість. Алгоритм можна інтегрувати в різні банківські системи, адаптуючи його до локальних умов та специфіки фінансових операцій;

Прогнозувальна здатність. Алгоритм здатен не лише виявляти шахрайські транзакції, а й прогнозувати потенційно небезпечні дії клієнтів на основі їхньої поведінки.

Соціально-економічний ефект

Розробка та впровадження алгоритму має значний вплив на соціально-економічну сферу:

Захист клієнтів від фінансових втрат. Надійна система безпеки мінімізує ризик несанкціонованого списання коштів;

Зменшення навантаження на служби безпеки. Завдяки автоматизації процесів виявлення шахрайства співробітники можуть зосередитися на складних кейсах;

Підвищення стабільності фінансової системи. Ефективне виявлення шахрайства сприяє зменшенню системних ризиків у банківському секторі.

Покращення міжнародного іміджу банків. Використання передових технологій у сфері кібербезпеки сприяє інтеграції національних банків у світову фінансову систему.

Використання результатів у суміжних сферах

Практичне значення роботи не обмежується лише банківською сферою. Розроблений підхід може бути застосований у:

Страхових компаніях – для виявлення підозрілих страхових випадків;

Електронній комерції – для виявлення шахрайських замовлень або платежів;

Державному секторі – у боротьбі з відмиванням грошей і фінансуванням незаконної діяльності;

Телекомунікаційних компаніях – для запобігання шахрайським схемам з тарифами та послугами;

Значення для України

Особливої актуальності дослідження набуває у контексті розвитку фінансової системи України:

активне впровадження онлайн-банкінгу та мобільних додатків;

розширення доступу населення до безготівкових розрахунків;

інтеграція у міжнародні фінансові та платіжні системи;

необхідність протидії кіберзлочинності в умовах воєнного стану та посиленої кіберзагрози;

Впровадження алгоритмів машинного навчання у банківську практику підвищить конкурентоспроможність вітчизняних фінансових установ, забезпечить захист клієнтів і сприятиме стабільності національної економіки.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ВИЯВЛЕННЯ ШАХРАЙСТВА В БАНКІВСЬКІХ СИСТЕМАХ

1.1 Поняття та види фінансового шахрайства

У сучасному світі науковий та технічний прогрес призводить до того, що проблема шахрайства стає все більш актуальною. Хоча розвиток технологій покращує якість життя, він також сприяє поширенню негативних явищ, зокрема шахрайства, яке охоплює різні сфери суспільства. Зараз шахрайство стає все складнішим, оскільки злочинці поєднують передові технології з сучасними психологічними методами, що робить його ще більш небезпечним. Фінансове шахрайство може впливати як на державні, так і на приватні фінанси. Під його дію потрапляють як фізичні, так і юридичні особи, а також сама держава, які в процесі своєї діяльності стикаються з різноманітними ризиками, що можуть безпосередньо чи опосередковано вплинути на досягнення бажаних фінансових результатів. У зв'язку з цим стає все важливішим вміння виявляти та зменшувати можливі ризики. Управління ризиками дозволяє приймати зважені управлінські рішення, враховуючи фактори невизначеності та ймовірність виникнення непередбачених

ситуацій, що дає змогу оцінити їхній вплив на досягнення поставлених цілей. У процесі формування правової держави в Україні, реалізації демократичних змін та зміцнення принципу верховенства права у всіх сферах державного управління роль права постійно зростає. Останні десятиліття цифрові технології розвиваються дуже швидко, створюючи основу для інновацій в усіх аспектах сучасного життя. Впровадження фінансових технологій в різних країнах залежить від особливостей законодавчого регулювання, рівня економічного розвитку, культурних особливостей і потреб суспільства. Останнім часом в Україні відбулося суттєве оновлення законодавства, що стосується використання фінансових технологій. Водночас варто зазначити, що сучасне законодавство України має певні недоліки, що дає можливість шахраям розробляти нові схеми обману, сподіваючись уникнути покарання. Тому важливим завданням є вдосконалення державного контролю в сфері боротьби з фінансовим шахрайством.[4]

1.2 Класифікація банківських транзакцій та ризиків

У сучасних умовах глобалізації та цифровізації фінансових ринків банки виконують ключову роль у забезпеченні руху капіталу, інвестицій та кредитування економіки. Основним інструментом реалізації банківських послуг є банківські транзакції, що відображають фінансові операції клієнтів та самого банку. Водночас здійснення транзакцій завжди супроводжується певними ризиками, які можуть мати як внутрішній, так і зовнішній характер. Тому питання класифікації банківських транзакцій і ризиків набуває особливої актуальності, адже дозволяє ефективно управляти фінансовими потоками, забезпечувати надійність банківської системи та підвищувати довіру клієнтів.

Класифікація банківських транзакцій

Банківські транзакції – це фінансові операції, які відображають рух грошових коштів чи зобов’язань між суб’єктами фінансової діяльності. Вони можуть бути класифіковані за різними ознаками.

За економічним змістом:

кредитні транзакції – надання чи отримання позик, кредитних ліній, факторинг, лізинг;

депозитні транзакції – відкриття та поповнення депозитів, виплата відсотків, пролонгація строків;

Розрахункові транзакції – платежі за товари та послуги, грошові перекази, міжбанківські розрахунки.

Інвестиційні транзакції – операції з цінними паперами, валютні угоди, хеджування.

За суб'єктами операцій:

Клієнтські транзакції – ініційовані фізичними чи юридичними особами (платежі, перекази, депозити);

Внутрішньобанківські транзакції – рух коштів між рахунками банку, управління ліквідністю;

Міжбанківські транзакції – розрахунки між банками, операції на міжбанківському ринку;

За способом здійснення:

Традиційні транзакції – через касу банку, письмові доручення;

Електронні транзакції – через інтернет-банкінг, мобільні додатки, платіжні системи;

Автоматизовані транзакції – регулярні платежі, автосписання, standing orders;

За територіальною ознакою:

Внутрішні – у межах однієї країни;

Міжнародні – транскордонні перекази, валютні операції, SWIFT-платежі;

За часовими характеристиками:

Миттєві – здійснюються в реальному часі (перекази через систему «швидких платежів»);

Відкладені – мають часовий лаг між ініціюванням та завершенням (наприклад, кліринг);

Таким чином, банківські транзакції охоплюють широкий спектр фінансових операцій, що відрізняються за змістом, технологіями та учасниками.

Способи запобігання фінансовому шахрайству Фізичною особою Юридичною особою Державою ніколи не ділитися даними платіжних карток з іншими особами встановити ефективну комунікацію з працівниками щодо попередження ризиків шахрайства забезпечувати контроль за цільовим і раціональним використанням коштів державного та місцевих бюджетів уникати відповіді на підозрілі повідомлення та не переходити за сумнівними лінками розробити систему, яка забезпечує належну підготовку та перевірку фінансових звітів протидіяти ухиленню від сплати податків і зборів. завжди зберігати конфіденційність одноразових паролів розподіляти обов'язки, повноваження та напрямки внутрішнього контролю перевіряти законність, обґрунтованість та ефективність управління державними майновими правами пройти ідентифікацію через свого оператора мобільного зв'язку сприяти розвитку корпоративної культури здійснювати нагляд за використанням державних резервів і запасів підвищувати рівень фінансової обізнаності створювати та публікувати зрозумілу та відкриту програму боротьби з фінансовим шахрайством в організації моніторити цільове використання та своєчасне повернення кредитів і позик, отриманих під державні чи місцеві гарантії тісно співпрацювати з правоохоронними органами розробляти власну стратегію для отримання інформації про можливі шахрайства сприяти прозорості та відкритості фінансових операцій брати участь в навчальних та інформаційних заходах щодо боротьби з фінансовим шахрайством і покращення фінансової грамотності створювати прозору політику санкцій щодо осіб, які здійснюють шахрайські дії налагоджувати активну взаємодію з урядами інших країн і міжнародними організаціями у сфері боротьби з фінансовим шахрайство. [5]

Класифікація банківських ризиків

Банківські ризики – це потенційні загрози, які можуть призвести до втрат банку або його клієнтів у процесі здійснення фінансових операцій. Їх класифікація допомагає систематизувати можливі загрози та розробити стратегії управління.

За джерелом виникнення:

- Внутрішні ризики – пов’язані з організацією діяльності банку (недосконалість внутрішнього контролю, помилки персоналу, шахрайство);
- Зовнішні ризики – обумовлені макроекономічними, політичними, правовими чи ринковими факторами (інфляція, кризи, воєнні дії);

За характером впливу:

- Фінансові ризики – кредитний, процентний, валютний, ліквідності;
- Операційні ризики – технічні збої, кібератаки, помилки в системах;
- Правові ризики – зміни в законодавстві, судові позови, регуляторні санкції;
- Репутаційні ризики – втрата довіри клієнтів унаслідок скандалів чи невиконання зобов’язань;

За ймовірністю виникнення:

- Високоймовірні ризики – пов’язані з масовими операціями (помилки при платежах, кіберзагрози);
- Середньої ймовірності – коливання процентних ставок, зміна курсу валют;
- Маловірогідні, але критичні – банкрутство контрагентів, дефолт держави;

За можливим рівнем втрат:

- Мінімальні ризики – втрати незначні, швидко компенсуються;
- Суттєві ризики – можуть вплинути на прибутковість банку;
- Катастрофічні ризики – ставлять під загрозу фінансову стабільність установи;

Взаємозв’язок транзакцій та ризиків

Кожна транзакція супроводжується певним набором ризиків. Наприклад, кредитна операція несе кредитний ризик неповернення боргу, депозитна – ризик ліквідності, міжнародна транзакція – валютний ризик та ризик комплаєнсу. Електронні операції потребують посиленого захисту від кібератак, тоді як традиційні – від внутрішніх зловживань.

Тому банки застосовують системи управління ризиками, що включають:

- ідентифікацію ризиків;
- їх кількісну та якісну оцінку;
- моніторинг транзакцій у режимі реального часу;
- створення резервів під можливі втрати;
- впровадження технологій захисту (AML-системи, KYC-процедури, багатофакторна аутентифікація);

Класифікація банківських транзакцій дозволяє впорядкувати широкий спектр операцій, що здійснюються банківськими установами, з урахуванням їх економічної сутності, суб'єктів, способу здійснення та часових характеристик. Водночас кожна транзакція супроводжується певними ризиками, які необхідно чітко ідентифікувати, класифікувати та контролювати.

Системний підхід до класифікації ризиків – внутрішніх і зовнішніх, фінансових та нефінансових – є передумовою ефективного управління банківською діяльністю. Це сприяє зниженню ймовірності збитків, підвищенню фінансової стійкості та збереженню довіри клієнтів.

Таким чином, класифікація банківських транзакцій і ризиків не є лише теоретичним інструментом, а має практичне значення для забезпечення стабільності банківської системи України в умовах сучасних викликів.

1.3 Методи виявлення шахрайських операцій у традиційних банківських системах

Шахрайські операції у банківській сфері є одним із найнебезпечніших викликів для фінансових установ. Вони підривають довіру клієнтів, знижують стабільність фінансової системи та можуть призвести до значних збитків. Особливої актуальності проблема набуває в умовах зростання цифровізації, але навіть у традиційних банківських системах (операції через відділення, каси, класичні платіжні документи) існує високий рівень ризику шахрайства. Для його мінімізації банки застосовують різні методи виявлення шахрайських операцій, які ґрунтуються на поєднанні організаційних, аналітичних та технологічних підходів.

Загальні підходи до виявлення шахрайства

Методи боротьби з шахрайством у банківських системах можна умовно поділити на три групи:

- Попереджувальні – спрямовані на створення умов, які ускладнюють або роблять неможливим здійснення шахрайства;
- Детективні (виявлення) – покликані швидко знаходити підозрілі операції під час або після їх здійснення;
- Коригувальні – застосовуються для ліквідації наслідків шахрайства і відновлення роботи системи;

У даній роботі основна увага приділяється саме методам виявлення шахрайських операцій у традиційних банківських системах.

Класифікація методів виявлення шахрайства

Організаційно-контрольні методи

До них належать:

- Розподіл повноважень – жоден працівник не повинен мати можливості одноосібно виконувати весь цикл банківської операції (наприклад, оформлення кредиту і його видача);
- Внутрішній аудит – регулярні перевірки операцій і документації дозволяють виявляти підозрілі дії співробітників чи клієнтів;
- Подвійна перевірка документів – усі фінансові документи (чеки, платіжні доручення) перевіряються кількома посадовими особами;
- Ротація персоналу – періодична зміна співробітників на ключових ділянках знижує ризик змови та приховування шахрайських схем;

Аналітичні методи:

- Фінансовий моніторинг – аналіз руху коштів клієнтів, пошук нетипових або підозрілих операцій (наприклад, значні суми за короткий проміжок часу без економічного обґрунтування);
- Метод порівняння – виявлення аномалій шляхом зіставлення поточних операцій клієнта з його типовою фінансовою поведінкою.

- Методи статистичного аналізу – застосування середніх значень, відхилень та коефіцієнтів для виявлення підозрілих операцій (наприклад, нетипово високий обіг готівки);
- Сценарний аналіз – створення типових сценаріїв шахрайства і перевірка операцій на їх відповідність;

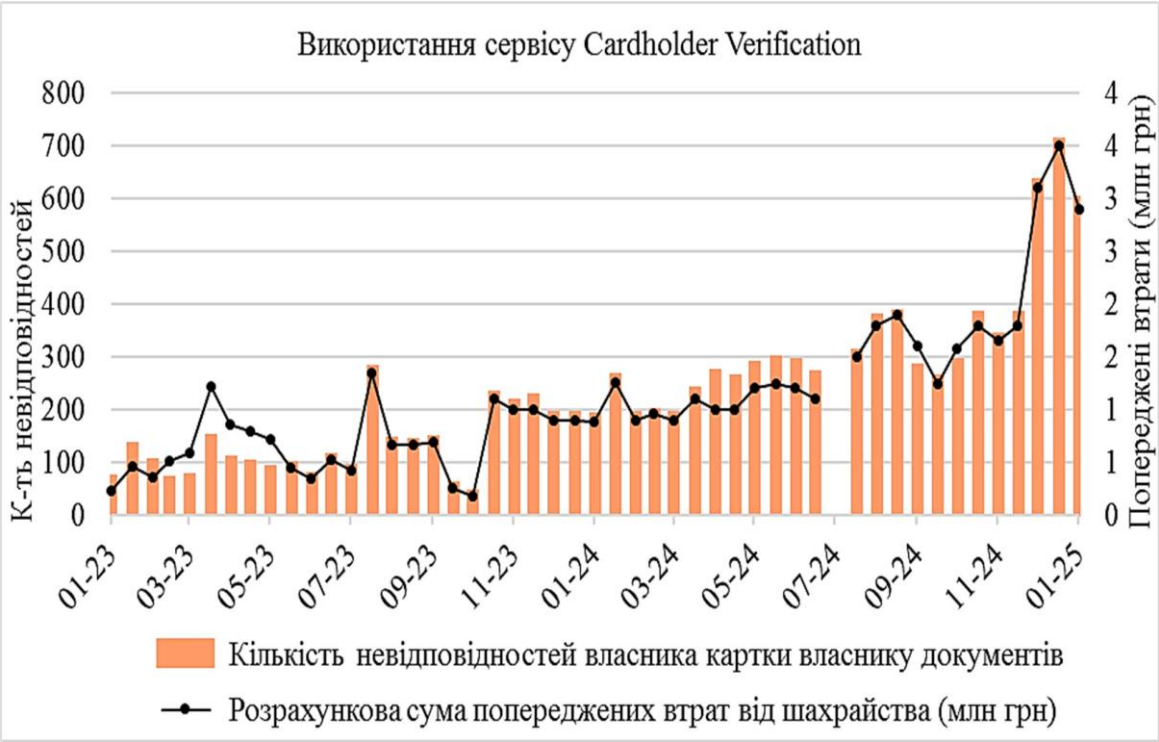


Рис.1 Використання сервісу Cardholder Verification :кількість власника картки власнику документа та розрахункова сума попереджених втрат від шахрайства

Джерело :побудовано на основі статистичних даних Української міжбанківської Асоціації членів платіжних систем[7;8]

Технологічні методи

Навіть у традиційних банківських системах використовується технічний інструментарій для виявлення шахрайства:

- Системи верифікації підписів – електронні або візуальні методи звірки підпису клієнта з еталоном;

- Перевірка достовірності документів – використання спеціальних пристроїв для ідентифікації підроблених купюр, печаток, бланків;
- Контроль операцій у реальному часі – навіть у відділеннях банки можуть застосовувати програмні комплекси, які автоматично сигналізують про ризикові транзакції;
- Використання «чорних списків» – перевірка клієнтів і контрагентів за базами даних шахраїв чи компаній із сумнівною репутацією;

Приклади шахрайських схем та методів їх виявлення

Підробка документів

У традиційних банківських системах поширеними є підроблені паспорти, довідки про доходи, гарантійні листи.

Методи виявлення:

- Перевірка документів через державні реєстри;
- Експертиза паперу й друку, запит до роботодавця клієнта;

«Касове шахрайство»

Співробітники банку можуть зловживати повноваженнями, привласнюючи частину коштів під час проведення операцій.

Методи виявлення:

- Подвійний контроль касових операцій;
- Регулярні інвентаризації готівки;
- Відеоспостереження у касових залах;

Шахрайство з кредитами

Клієнти можуть отримувати кредити на підставі фіктивних даних або за підробленими документами.

Методи виявлення:

- Ретельна перевірка кредитної історії;
- Підтвердження доходів;
- Співбесіда з клієнтом;
- Запровадження скорингових моделей;

Змови працівників і клієнтів

Небезпечні схеми, коли банківський працівник сприяє шахраю в оформленні неправомірних операцій.

Методи виявлення:

- Система «поділу обов'язків»;
- Внутрішні розслідування;
- Контроль доступу співробітників до різних сегментів інформації;

Сучасні тенденції у розвитку методів виявлення шахрайства

Хоча традиційні банківські системи поступово модернізуються, ключовим викликом залишається поєднання класичних методів контролю з новими технологіями. Серед актуальних тенденцій можна виділити:

-Впровадження систем штучного інтелекту для аналізу поведінки клієнтів;

-Використання машинного навчання для прогнозування шахрайських дій;

-Посилення регуляторних вимог (KYC – «знай свого клієнта», AML – «боротьба з відмиванням грошей»);

створення централізованих баз даних підозрілих операцій, якими обмінюються банки та регулятори.

Шахрайські операції у традиційних банківських системах становлять серйозну загрозу для стабільності фінансових установ. Методи їх виявлення охоплюють широкий спектр інструментів – від організаційно-контрольних заходів до сучасних технологій моніторингу. Найефективнішими вважаються комплексні підходи, які поєднують внутрішній контроль, аналітику транзакцій, використання спеціалізованого програмного забезпечення та співпрацю.

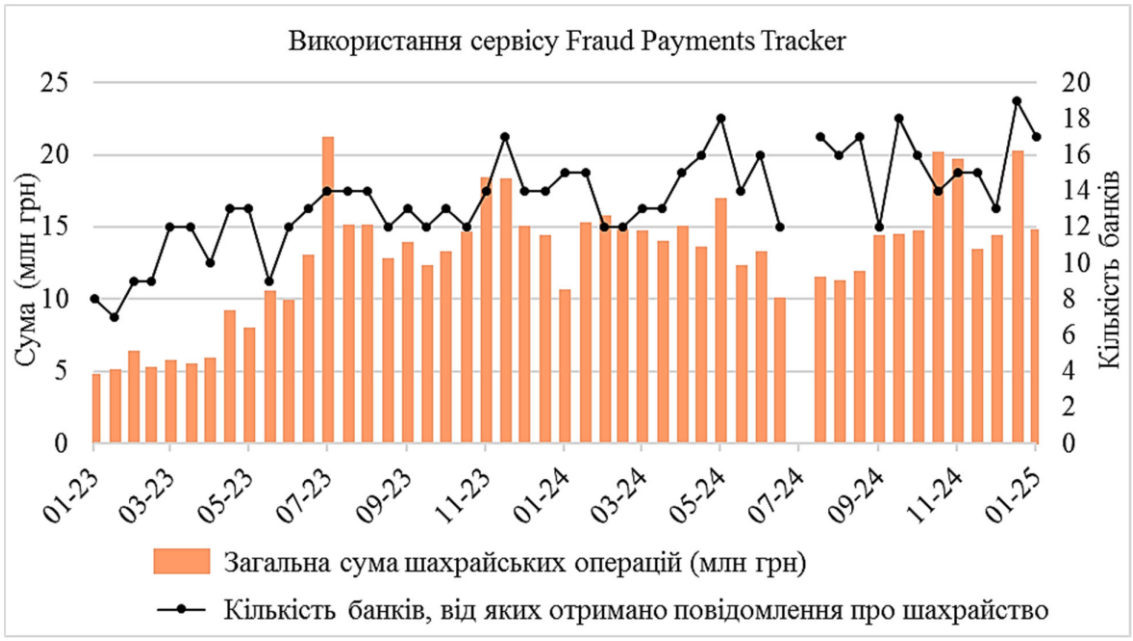


Рис.2 Загальна сума попереджених операцій за алертами та кількість банків ,які про них повідомляли ,за даними сервісу Frand Payments Tracker

Джерело : побудовано на основі статистичних даних Української міжбанківської Асоціації платіжних систем [7;8]

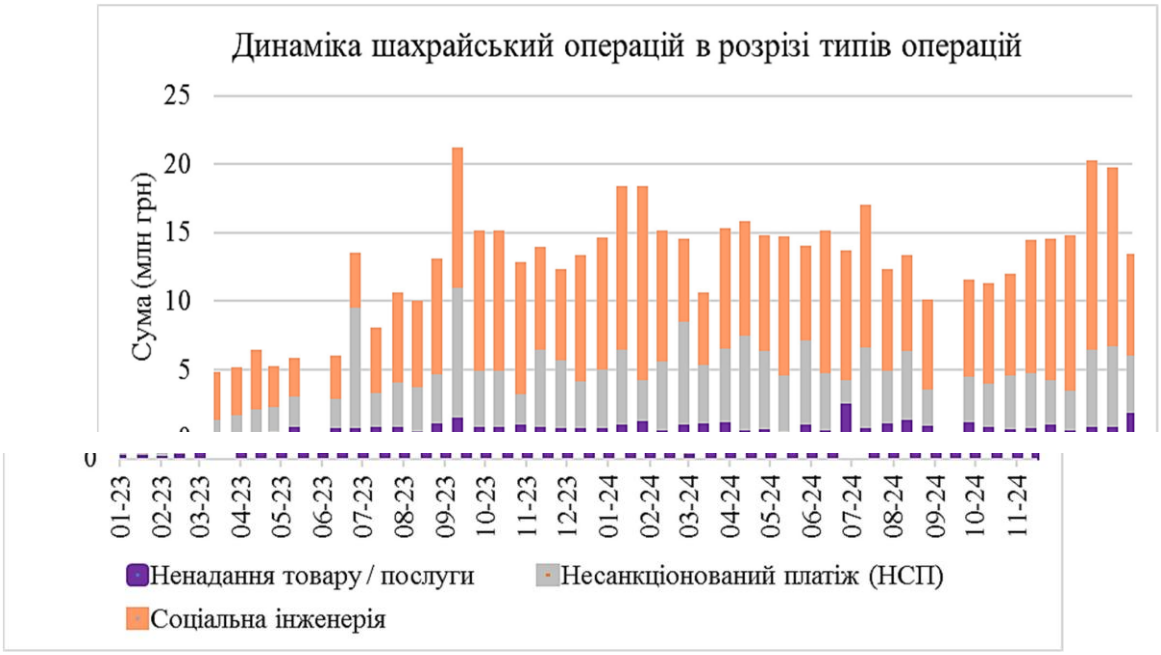


Рис.3 Динаміка шахрайських операцій в розрізі типів операцій за даними сервісу Fraud Payments Tracker

Джерело : побудовано на основі статистичних даних Української міжбанківської Асоціації платіжних систем [7;8]

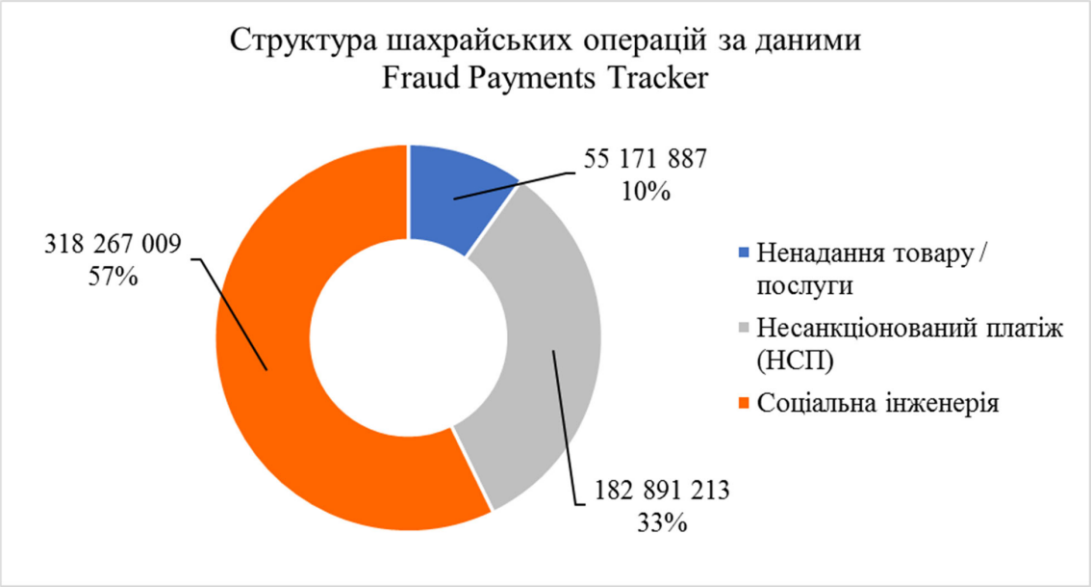


Рис.4 Структура суми шахрайських операцій в розрізі типів за даними сервісу Fraud Payments Tracker

Джерело: побудовано на основі статистичних даних Української міжбанківської Асоціації членів платіжних систем [7;8]

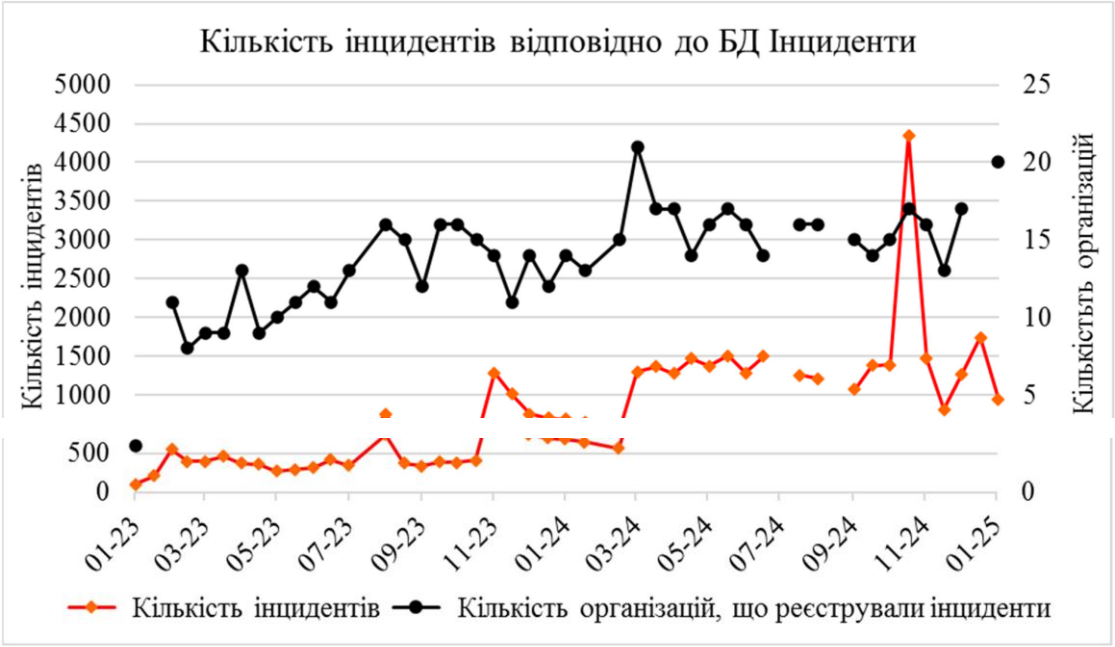


Рис. 5. Кількість інцидентів та кількість організацій, що реєстрували інциденти, в БД Інциденти

Джерело: побудовано на основі статистичних даних Української міжбанківської Асоціації членів платіжних систем[7;8]

1.4 Порівняльний аналіз сучасних підходів

Виявлення аномалій у мікросервісних системах є складним завданням, що потребує вибору оптимального методу залежно від особливостей даних, вимог до продуктивності та точності. Серед найпоширеніших підходів у цій сфері використовуються автоенкодери (AE), рекурентні нейронні мережі (RNN), довготривало-короткочасна пам’ять (LSTM) та згорткові нейронні мережі (CNN). Technical sciences ISSN 2307-5732 358 Herald of Khmelnytskyi national university, Issue 3, part 1, 2025 (351) Автоенкодери є ефективними для виявлення відхилень у багатовимірних даних. Вони навчаються реконструювати нормальні зразки, і якщо похибка реконструкції перевищує певний поріг, об'єкт вважається аномальним. Цей метод працює добре, якщо аномалії значно відрізняються від нормальних даних, проте у випадку поступових змін або аномалій, схожих на нормальні патерни, AE можуть не виявити відхилення. До того ж вони не є адаптивними: при зміні

поведінки системи автоенкодери потребують перевчання, що є ресурсомістким процесом. Рекурентні нейронні мережі, зокрема LSTM, враховують часовий контекст, що робить їх корисними для аналізу метрик, що змінюються з часом. Вони ефективно прогнозують майбутні стани системи, порівнюють очікувані та фактичні значення і, у разі значного розходження, позначають спостереження як аномальне. Це робить LSTM одним із найкращих методів для виявлення закономірних відхилень, проте він має високу обчислювальну складність, що ускладнює використання у реальному часі без відповідної оптимізації. Згорткові нейронні мережі використовуються для аналізу багатовимірних часових рядів, особливо у випадках, коли аномалії проявляються як локальні відхилення у даних. Вони добре працюють завдяки можливості автоматичного виділення особливостей у сигналах, що може бути складним для інших методів. CNN забезпечують високу швидкість обробки завдяки паралельним обчисленням і є придатними для роботи у реальному часі, особливо у разі використання графічних процесорів (GPU). Проте цей підхід менш ефективний для довготривалих часових залежностей, оскільки він аналізує лише локальні патерни, що може призвести до пропуску певних типів аномалій. З точки зору роботи у реальному часі, найкраще підходять CNN та автоенкодери, оскільки вони швидко обробляють вхідні дані. LSTM мають вищу обчислювальну складність, що може створювати затримки у великих системах. Проте гібридні моделі, наприклад комбінація CNN+LSTM або AE+LSTM, дозволяють досягти кращого балансу між продуктивністю та точністю. Вибір конкретного підходу залежить від типу аномалій, які потрібно виявляти, та вимог до швидкості роботи системи. Далі в таблиці 1 наведено порівняльну характеристику основних методів виявлення аномалій за ключовими параметрами. Таблиця 1 Порівняльна характеристика основних методів виявлення аномалій Критерій Автоенкодери (AE) LSTM CNN Точність Висока, але залежить від якості вибору порогу реконструкції Висока для часових рядів, добре прогнозує зміни Висока, якщо аномалії мають характерні локальні патерни Стійкість до шуму Помірна, може бути підвищена використанням шумостійких автоенкодерів Висока, добре фільтрує випадкові відхилення Висока, оскільки згорткові шари усереднюють нерелевантні деталі

Адаптивність Низька, потребує перевчання при зміні характеристик системи Висока, здатність адаптуватися до змін у часових рядах Помірна, нові шаблони можуть вимагати донавчання Обчислювальна складність Помірна, швидка обробка після навчання Висока, особливо для довгих часових послідовностей Низька, ефективна робота на GPU Можливість роботи у реальному часі Висока після навчання, швидке обчислення похибки Середня, залежить від складності моделі та довжини послідовності Висока, добре підходить для потокового аналізу Чутливість до типу даних Найкраще працює на стаціонарних наборах даних Підходить для часових рядів із довготривалими трендами Найкраще працює на просторових та короткочасних залежностях Складність налаштування Середня, необхідний підбір порога аномалій Висока, потребує великої кількості гіперпараметрів Низька, оскільки стандартні архітектури працюють добре Придатність до великих даних Висока, обчислення реконструкції масштабуються добре Середня, складність зростає з довжиною послідовностей Висока, особливо при використанні GPU Загалом, ефективність кожного підходу визначається специфікою даних та вимогами до продуктивності. Автоенкодери добре підходять для виявлення загальних аномалій у стаціонарних наборах даних, але потребують повторного навчання у разі змін у поведінці системи. LSTM забезпечують високу точність при аналізі часових рядів, але вимагають значних обчислювальних Технічні науки ISSN 2307-5732 Herald of Khmelnytskyi national university, Issue 3, part 1, 2025 (351) 359 ресурсів, що може ускладнювати їх використання у реальному часі. CNN працюють швидко та ефективно, особливо при потоковому аналізі багатовимірних даних, але можуть не враховувати довготривалі залежності, якщо вони не представлені у локальних шаблонах. Комбіновані підходи, такі як AE+LSTM або CNN+LSTM, можуть компенсувати слабкі сторони окремих моделей і забезпечити кращу загальну продуктивність. Наприклад, згорткові шари можуть швидко виділяти основні особливості у часових рядах, після чого LSTM-мережа аналізуватиме довготривалі взаємозв’язки між ними. Так само AE можуть використовуватися для попереднього зменшення розмірності вхідних даних перед обробкою їх LSTM, що знижує обчислювальне навантаження. Вибір конкретного методу залежить від того,

що є критично важливим у системі моніторингу: швидкість обробки чи точність виявлення складних аномалій. Якщо потрібна швидка оцінка великих потоків даних у реальному часі, CNN будуть найкращим варіантом. Для виявлення складних закономірностей у часових рядах найефективнішими будуть LSTM. Автоенкодери є хорошим компромісним варіантом для загального виявлення відхилень, але потребують ретельного налаштування параметрів порогу аномалій. З урахуванням усіх аспектів, використання гібридних підходів залишається найбільш перспективним напрямком розвитку методів виявлення аномалій у мікросервісних системах. Подальші дослідження можуть бути спрямовані на створення адаптивних ансамблевих моделей, що комбінують переваги кожного підходу та здатні самостійно оновлюватися у процесі роботи [11].

Висновки до розділу 1

У процесі дослідження було розглянуто ключові аспекти, що стосуються поняття та видів фінансового шахрайства, класифікації банківських транзакцій і ризиків, методів виявлення шахрайських операцій у традиційних банківських системах, а також сучасних підходів до виявлення аномалій у фінансових процесах. Проведений аналіз дозволяє зробити низку узагальнень, які мають як теоретичне, так і практичне значення.

По-перше, встановлено, що фінансове шахрайство є багатогранним явищем, яке охоплює як класичні, так і новітні форми зловживань. У сучасних умовах його поширенню сприяє стрімкий розвиток цифрових технологій, поєднання яких із психологічними методами маніпуляції створює нові виклики для фінансової безпеки. Жертвами фінансового шахрайства можуть стати як окремі громадяни, так і юридичні особи, фінансові установи та навіть держава в цілому. Тому проблема протидії шахрайству вимагає системного підходу, що передбачає поєднання правових, організаційних, технологічних та освітніх інструментів.

По-друге, аналіз банківських транзакцій показав, що вони є фундаментальним інструментом функціонування фінансової системи. Їх класифікація за економічним змістом, суб'єктами, способом здійснення, територіальною ознакою та часовими характеристиками дає змогу впорядкувати різні види операцій і забезпечити більш ефективне управління ними. Важливо підкреслити, що кожен вид транзакцій супроводжується власним набором ризиків, серед яких особливе місце займають кредитний, валютний, операційний та репутаційний ризики. Усвідомлення взаємозв'язку між транзакціями та ризиками дозволяє банкам своєчасно впроваджувати механізми захисту й забезпечувати стабільність діяльності.

По-третє, вивчення методів виявлення шахрайських операцій у традиційних банківських системах засвідчило необхідність комплексного підходу до їх застосування. Організаційно-контрольні методи (поділ повноважень, внутрішній аудит, ротація персоналу) забезпечують мінімізацію внутрішніх зловживань, тоді як аналітичні методи (фінансовий моніторинг, сценарний аналіз, статистичні моделі) допомагають ідентифікувати підозрілі дії клієнтів. Технологічні методи, зокрема використання систем верифікації підписів, перевірка документів та моніторинг операцій у реальному часі, дають можливість швидко реагувати на шахрайські дії. Важливо, що ефективність цих методів зростає у разі їх інтеграції в єдину систему фінансової безпеки.

По-четверте, приклади шахрайських схем показують, що найчастіше зловмисники використовують підробку документів, касові маніпуляції, кредитні шахрайства та змови із працівниками банку. Це доводить, що внутрішній контроль у банках має бути не формальністю, а дієвим механізмом, який дозволяє своєчасно виявляти відхилення та запобігати зловживанням.

По-п'яте, сучасні тенденції свідчать про зростання ролі штучного інтелекту та машинного навчання у сфері протидії фінансовому шахрайству. Завдяки цим технологіям банки отримують можливість аналізувати поведінкові патерни клієнтів, виявляти приховані закономірності та прогнозувати ризики ще до того, як відбудеться шахрайська операція. Такі підходи підсилюються регуляторними

вимогами, зокрема запровадженням процедур KYC («знай свого клієнта») та AML («боротьба з відмиванням доходів, отриманих злочинним шляхом»). У цьому контексті особливо важливим є створення централізованих баз даних шахрайських операцій, доступних для всіх учасників фінансової системи.

По-шосте, проведений порівняльний аналіз сучасних методів виявлення аномалій у мікросервісних системах засвідчив, що кожен підхід має власні переваги та недоліки. Автоенкодери є ефективними для загального виявлення відхилень, рекурентні мережі (LSTM) добре працюють із часовими рядами, тоді як згорткові мережі (CNN) дозволяють швидко обробляти великі обсяги даних у реальному часі. Найперспективнішим напрямом визнано гібридні моделі, які поєднують різні методи та забезпечують баланс між точністю і швидкістю.

Отже, розділ 1 дозволив дійти висновку, що ефективна боротьба з фінансовим шахрайством потребує інтегрованого підходу. Лише поєднання правового регулювання, внутрішнього контролю в банках, впровадження сучасних технологій аналізу та міжнародної співпраці здатне забезпечити надійний захист фінансової системи. У цьому контексті ключовими завданнями для України залишаються вдосконалення законодавчої бази, підвищення фінансової грамотності населення, розвиток інфраструктури фінансового моніторингу та тісна взаємодія банків із правоохоронними органами.

Загалом, протидія фінансовому шахрайству є безперервним процесом, який потребує постійного оновлення методів та інструментів відповідно до нових викликів. Тільки за умови комплексного підходу можливо гарантувати стабільність банківської системи та зберегти довіру суспільства до фінансових інститутів.

РОЗДІЛ 2. ТЕХНОЛОГІЧНІ ЗАСАДИ BIG DATA ТА МАШИННОГО НАВЧАННЯ

2.1 Концепція Big Data та її застосування у фінансовій сфері

Сучасні децентралізовані фінанси (DeFi) зазнають значних трансформацій під впливом новітніх цифрових технологій, зокрема Big Data та штучного інтелекту (AI). Використання цих технологій створює можливості для поглибленого аналізу ринкових процесів, підвищення точності прогнозів та мінімізації інвестиційних ризиків. Зважаючи на високу динаміку DeFi-екосистеми та її значну волатильність, питання коректної оцінки вартості проєктів є критично важливим для інвесторів, аналітиків та розробників фінансових продуктів. У даній роботі проводиться аналіз основних механізмів застосування технологій Big Data та AI у процесі оцінки DeFiпроектів, визначаються ключові виклики та перспективи їх впровадження в сучасну економічну реальність. Технологія Big Data відіграє ключову роль у сучасних аналітичних системах, що використовуються у фінансовій сфері, особливо в межах DeFi. Масиви даних, отримані з блокчейн-мереж, містять значну кількість корисної інформації, яка може бути проаналізована з метою ідентифікації основних ринкових трендів та прогнозування майбутньої динаміки активів. До найбільш релевантних джерел даних належать транзакційні записи з основних блокчейнів, таких як Ethereum, Binance Smart Chain та Solana, аналітичні метрики, зокрема Total Value Locked (TVL), показники ліквідності активів, історичні патерни торгівлі, а також соціальноекономічні фактори, що впливають на інтерес інвесторів до конкретних проєктів. Штучний інтелект є потужним інструментом, що дозволяє автоматизувати

обробку та аналіз великих обсягів даних. Зокрема, глибоке навчання та нейронні мережі можуть використовуватися для виявлення складних закономірностей у поведінці DeFi-активів, створення точних прогнозів цінових змін, а також оцінки рівня ризику, пов'язаного з інвестиціями в той чи інший проєкт. Особливу увагу слід приділити розробці моделей машинного навчання, здатних адаптуватися до швидко мінливих ринкових умов, а також інтеграції алгоритмів Natural Language Processing (NLP) для аналізу інформаційних потоків у

криптовалютних медіа та соціальних мережах. На сьогодні існує низка методів та підходів до застосування AI у фінансовій аналітиці, включаючи такі концепції, як reinforcement learning (навчання з підкріпленням), adversarial networks (змагальні нейронні мережі), hybrid forecasting models (гібридні моделі прогнозування). У поєднанні з технологією блокчейн та smart contracts вони дозволяють створювати більш прозорі та ефективні системи оцінки ринкової вартості активів. Однак, попри значний потенціал, використання Big Data та AI у DeFiаналітиці стикається з низкою викликів. Одним із ключових питань є проблема якості даних, оскільки велика частина блокчейн-інформації є неструктурованою, а її верифікація потребує застосування додаткових механізмів перевірки. Масштабованість також є важливим фактором, оскільки швидкість обробки транзакцій постійно зростає, що створює потребу у високопродуктивних обчислювальних потужностях. Крім того, значний вплив на оцінку вартості DeFiпроєктів мають регуляторні фактори, що визначають правові умови для їхнього функціонування, а також рівень довіри з боку інституційних інвесторів. Подальший розвиток Big Data та AI у DeFi-аналізі відкриває перспективи для значного покращення оцінки вартості цифрових активів. Використання адаптивних алгоритмів, що поєднують методи машинного навчання, аналіз блокчейн-даних та поведінкових фінансів, дозволить підвищити точність прогнозів і знизити ризики інвестування. Зростаюча популярність DeFi серед роздрібних та інституційних інвесторів робить необхідним удосконалення методик оцінки, що враховують не лише класичні фінансові показники, а й поведінкові аспекти прийняття рішень. Таким чином, інтеграція Big Data та AI у процес оцінки DeFi-проектів є важливим напрямом сучасних фінансових досліджень. Вдосконалення методів обробки блокчейн-даних, підвищення точності прогнозування та оптимізація алгоритмів управління ліквідністю дозволять сформувати більш стабільну та передбачувану DeFi-екосистему. Це, у свою чергу, сприятиме подальшому зростанню довіри до цифрових фінансів та розширенню їхнього використання у глобальній економіці. [12]

2.2 Технології обробки (Hadoop ,Spark ,Kafka тощо)

Apache Kafka — це розподілена платформа обміну повідомленнями, яка була спочатку розроблена в LinkedIn для обробки подій у реальному часі. Її архітектура була створена з урахуванням потреб високонавантажених систем. Kafka підтримує зберігання потоків подій, розподілення навантаження між численними консьюмерами та роботу у кластері з кількох брокерів [1, 2]. Kafka реалізує модель publish-subscribe: продюсери надсилають повідомлення до певних topic, а консьюмери підписуються на них і отримують дані. Ключовою особливістю Kafka є те, що повідомлення зберігаються в партиційованих логах. Кожен споживач самостійно веде облік зчитаних повідомлень за допомогою offset — це дозволяє повертатися до обробки з попереднього місця у разі збою, а також зменшує навантаження на систему доставки. Kafka відрізняється високою масштабованістю: нові брокери або консьюмери можуть підключатися до системи без зупинки сервісу. Повідомлення в Kafka зберігаються не в оперативній пам’яті, а на диску, що дозволяє працювати з великими обсягами історичних даних без втрати продуктивності [8]. Ще однією перевагою Kafka є гнучка інтеграція з іншими технологіями. Існують модулі Kafka Streams, Kafka Connect, а також готові коннектори для інтеграції з базами даних, файловими системами, аналітичними платформами (Spark, Flink), що дозволяє будувати повноцінні потокові пайплайни обробки даних.

Apache Kafka функціонує як розподілена платформа для обміну повідомленнями, у центрі якої лежить модель публікації-підписки (publishsubscribe). Цей принцип дозволяє системі ефективно забезпечувати обмін подіями між численними незалежними компонентами — від мікросервісів до зовнішніх аналітичних систем — без потреби в жорсткому зв’язуванні їх між собою. Таке роз’єднання компонентів підвищує масштабованість, надійність і гнучкість всієї архітектури. У моделі publish-subscribe є три основні ролі:
- Producer (видавець) — додаток або сервіс, який генерує повідомлення та надсилає їх у певний топік (topic);
-

Broker — сервер Kafka, який приймає, зберігає та передає повідомлення; Один кластер Kafka може містити кілька брокерів для горизонтального масштабування. Consumer (підписник) — додаток або сервіс, який підписується на певні топіки й отримує повідомлення, обробляючи їх у потрібний спосіб. Процес комунікації в Kafka виглядає наступним чином: продюсери публікують повідомлення у топіки, брокери зберігають ці повідомлення в логах, а консьюмери, що підписані на відповідні топіки, читають та обробляють повідомлення у своєму темпі. Завдяки цьому система досягає асинхронності, тобто компоненти не зобов'язані працювати одночасно — кожен з них може обробляти дані незалежно у своєму темпі. Ключовою особливістю Kafka є логічна структура зберігання повідомлень. Кожен топік ділиться на партиції — впорядковані послідовності записів (логи), кожен з яких має унікальний номер — offset. Offset визначає позицію повідомлення в межах конкретної партиції. Консьюмер, зчитуючи повідомлення, самостійно відстежує offset, що дає змогу: зберігати прогрес обробки даних (наприклад, після збою повернутись до останнього успішно прочитаного повідомлення);

реалізовувати паралельну обробку — кілька консьюмерів можуть читати з різних партицій одного топіка; уникати дублювання або втрати даних у разі повторного зчитування. Для забезпечення високої продуктивності Kafka реалізує лог-сегментацію. Повідомлення в партиціях зберігаються у вигляді файлів — логів, які поділяються на сегменти. Кожен сегмент є окремим файлом на диску, який автоматично закривається при досягненні певного розміру або часу. Завдяки такому підходу Kafka: забезпечує постійну швидкість запису, оскільки нові повідомлення додаються лише в кінець останнього сегмента; може зберігати великі обсяги історичних даних без втрати продуктивності; дає можливість швидко видаляти старі повідомлення, не змінюючи структуру логів (наприклад, через політику retention). Кожен брокер Kafka може обслуговувати кілька партицій та зберігати копії (репліки) партицій з інших брокерів. Таким чином забезпечується відмовостійкість: якщо один брокер виходить з ладу, його партиції можуть бути оброблені іншими брокерами, що мають актуальні репліки. Ще однією важливою

перевагою є гнучка конфігурація доставки повідомлень. Kafka дозволяє налаштовувати: acknowledgment (підтвердження) — чи повинен продюсер чекати підтвердження доставки повідомлення на всі репліки; delivery semantics — гарантії доставки (at-most-once, at-least-once або exactly-once); retention policy — політику зберігання повідомлень (за часом або за розміром). Завдяки поєднанню моделі publish-subscribe, лог-сегментації та масштабованої партиційної архітектури Apache Kafka стає надзвичайно ефективним інструментом для побудови систем обробки подій у реальному часі, що працюють із великим навантаженням.[13]

Таблиця 1,1 Різниця характеристик різних брокерів повідомлень .Джерело :[13]

Характеристика	Apache Kafka	RabbitMQ/Active MQ	ZeroMQ
Модель передачі	Publish-Subskribe (логічна черга)	Черги(queue-based fanout)	Peer to peer
Сховище	Дискове лог-зберегання з сегментацією	У пам'яті з опційним дисковим записом	Без збереження (in-memory)
Затримка обробки	Низька (для великих обсягів)	Низька-середня	Мінімальна (але не стійка)
Підтримка історії	Повна до заданого періоду	Часткова	Вітсутня
Масштабування	Горизонтальне (через партиції)	Вертикальне/обмежене горізонтальне	Вручну через логіку програми

Kafka створена для високонавантажених середовищ, де обсяг переданих повідомлень може сягати мільйонів на секунду. Її архітектура оптимізована для великого throughput (пропускної здатності) при низьких затратах на апаратні ресурси завдяки запису даних у журнали та зчитуванню по offset-ах. RabbitMQ і ActiveMQ добре працюють у невеликих або середніх системах, особливо там, де важлива гнучкість маршрутизації повідомлень. Проте при зростанні обсягу даних вони швидко вичерпують свої ресурси й потребують складних підходів до масштабування. ZeroMQ вирізняється максимальною швидкістю, оскільки не має брокера як такого. Але ця перевага нівелюється тим, що відсутнє зберігання повідомлень і будь-який збій у мережі веде до втрат. ZeroMQ потребує значно більших зусиль для створення стійкої інфраструктури. Надійність і гарантії доставки Kafka має гнучкі механізми гарантування доставки повідомлень: можливість вибору між доставкою "принаймні один раз" (at-least-once), "не більше одного разу" (at-most-once) або "рівно один раз" (exactly-once). Це дозволяє адаптувати систему до потреб конкретного бізнес-процесу. RabbitMQ теж підтримує підтвердження доставки, проте залежить від використаної черги та конфігурацій. Активне використання транзакцій і підтверджень може знижувати продуктивність. ActiveMQ подібний до RabbitMQ, але орієнтований переважно на Java-середовище. ZeroMQ практично не має гарантій доставки, покладаючись на логіку в коді користувача. Це робить його дуже швидким, але водночас — ризикованим для критичних систем.

Kafka має потужну екосистему:

- Kafka Streams для обробки потоку даних ;
- Kafka Connect для підключення джерел/сховищ ;
- Численні клієнтські бібліотеки для Java, Python, Go, Node.js тощо. Її активно інтегрують з Apache Flink, Spark, Hadoop, що робить Kafka основою для аналітичних і ETL-систем.

RabbitMQ має широке API і численні плагіни, але поступається Kafka в продуктивності та масштабованості. ActiveMQ добре інтегрується з Java EE, але менш активний у підтримці сучасних стеків. ZeroMQ — це радше транспортна

бібліотека, ніж повноцінний брокер, тому інтеграція повністю залежить від розробника і не має централізованої підтримки або інструментів моніторингу. [13]

Кількість повідомлень	Конфігурація (consumer-partition)	Середня затримка (мс)	Загальний час (сек)	Пропускна здатність (msg/сек)
10	1-1	5	0,06	167
	1-2	5	0,06	167
	2-1	5	0,06	167
	2-2	4	0,05	200
100	1-1	8	1,4	71
	1-2	7	1,2	83
	2-1	8	1,4	71
	2-2	6	0,9	111
1000	1-1	12	15	66
	1-2	10	12	83
	2-1	11	15	66
	2-2	7	8,5	117

Таблиця 1.2 Затримка в Kafka за різних конфігурацій обробки повідомлень

Джерело : [13]

Розмір буфера (MB)	Середня затримка (мс)	Пропускна здатність (msg/сек)	Випадки очикування буфера	Помилки
1	27	370	Часті	3
16	14	680	Рідко	0
64	9	1030	Відсутні	0
128	8	1090	Відсутні	0

Таблиця 1.3 Залежність затримки від параметру buffer.memory у Kafka Producer

Джерело : [13]

2.3 Алгоритм машинного навчання для задач класифікації та аномалій

Однією з найвідоміших кіберзагроз є програми-вимагачі через їхні серйозні наслідки для ресурсів жертви. Причиною таких наслідків є глобальне поширення шкідливого ПЗ через інтернет-сервіси. Одним із факторів є запуск концепції "Ransomware as a Service" (RaaS) [18]. Розробники програм-вимагачів і зловмисники використовують RaaS для здійснення атак на жертв. Найпоширенішим методом зараження є електронна пошта — понад 64% усіх активацій програм-вимагачів відбувається онлайн [19]. Жертві надсилається фішинг-лист із шкідливим вкладенням або посиланням. Після взаємодії з листом, програма-вимагач запускається, передає дані системи на сервер командування й управління (C&C) для отримання ключа шифрування. Після отримання ключа — дані жертви шифруються, а на екрані з’являється повідомлення з вимогою викупу в криптовалюті (наприклад, Bitcoin). Одним із рішень є моніторинг мережевої активності — важливий компонент виявлення вторгнень. Системні адміністратори мають постійно стежити за аномальною активністю. Пасивні атаки мають на меті непомітне збирання інформації про систему для подальшої активної атаки. Проте мережевий моніторинг допомагає запобігти як пасивним, так і активним загрозам. Контрольні показники, зокрема географічні дані, час підключення та кількість

з’єднань, відслідковуються та порівнюються з прогнозованими значеннями. Якщо виявлено відхилення — запускаються профілактичні заходи.[17]

2.4 Методи підвищення точності моделей (ансамблеви методи ,нейронні мережі)

Було залучено три репрезентативні публічні набори, які всебічно висвітлюють аспекти фінансових і страхових ризиків: Home Credit Default Risk (індикатори дефолту), Motor Vehicle Insurance Portfolio (ретроспектива збитковості полісів) та Auto Insurance Claims 2024 (фіксація страхового шахрайства). У таблиці 1.4 наведені докладні кількісні параметри

Напрям ризику	Відкритий набір даних	Роки спостережень	Кількість рядків	Початкова кількість ознак	Бінарна ціль
Кредитний	Home Credit Default Risk (Kaggle)	2015-2018	307511	122	TARGET-90денний дефолт
Страховий (портфель)	Motor Vehicle Insurance Portfolio (Mendeley Data)	2015-2018	105555	30	claim-настання збитку
Страховий (вимоги)	Auto Snsurance Claims Ubdated to 2024 (Kaggle)	2020-2024	9134	34	fraud_reported-шахрайство

Таблиця 1.4 Кількісні параметри.

Сукупний обсяг перевищує чотири сотні тисяч спостережень, а первинна розмірність ознакового простору становить 186 полів. Обрані дані досить нові, що

забезпечує відповідність моделі сучасним тенденціям та динаміці ринку страхування та фінансів. Велика кількість ознак дозволяє побудувати гнучку та інформативну модель, здатну враховувати широкий спектр факторів ризику. Після консолідації вибірок першим аналітичним кроком стала систематична інвентаризація пропусків. Для кожного атрибута xj було розраховано показник:

$$missratioj = 1\ N\ N\ i{=}1 \sum I(xij = \emptyset)\ . \qquad (1.1)$$

Ця метрика визначає питомий внесок відсутніх значень. Якщо частка пропусків не перевищувала п’яти відсотків, ознака класифікувалася як так звана «чиста». У числових колонах із вищим рівнем неповноти застосовувалася медіанна імпутація x:

$$xij\ new = \text{median}\{xkj \mid xkj \neq \emptyset\}. \qquad (1.2)$$

Вона забезпечує стійкість до одиничних екстремальних відхилень. Категоріальні поля з пропусками доповнювалися синтетичним рівнем Missing, що слугував явним маркером невизначеності для подальшого машинного навчання. Наступним етапом було усунення екстремальних спостережень. Для кожної числової змінної спершу обчислювався міжквартильний розмах $IQR = Q3{-}Q1$,після чого встановлювалися межі допустимого інтервалу $lower = Q1\ {-}1.5$ IQR та $upper = Q3\ {+}1.5$ IQR. Значення xij, що виходили за зазначені бар’єри, підлягали вінзоризації до відповідної границі, що нормалізувало статистичні характеристики дисперсії без втрати інформативності [21]. Далі здійснювалось шкалювання числових атрибутів за допомогою стандартного z-перетворення.

$$zij = xij{-}\mu j\ 24\ \sigma j\ . \qquad (1.3)$$

Середні значення μj та стандартні відхилення σj обчислювалися виключно на тренувальній підвибірці, що унеможливлювало інформаційний витік на етапах валідації та тестування. Для кодування категоріальних змінних використовувались

два підходи: при кардинальності не більш як десять класів застосовувався One-Hot, у протилежному випадку – таргет-енкодер, формально заданий виразом: $\sum tj(c) = i{:}xij{=}c\ yi + \alpha\mu\ nc + \alpha$, з глобальним середнім та параметром згладжування

$$\alpha{=}20.\ \mu = 1\ N\ N\ \sum yi\ i{=}1\ . \qquad (1.4)$$

з глобальним середнім та параметром згладжування $\alpha{=}20.\ \mu = 1\ N\ N\ \sum yi\ i{=}1$. (1.4) (1.5) Така техніка мінімізувала ризик перенавчання на рідкісних категоріях. Попри виконані маніпуляції, співвідношення класів залишалося дисбалансованим (8,07 % дефолтів, 11,9 % збиткових полісів, 9,6 % шахрайських вимог). Для корекції диспропорції в навчальній підмножині було інтегровано комбінацію SMOTE і Tomek Links: перший метод генерував синтетичні позитивні зразки у ε-околах, тоді як другий усував пограничні пари, що знижували чіткість розділової поверхні. Валідаційний і тестовий сегменти залишилися недоторканими, що гарантувало об’єктивність подальших оцінок. Підсумкова матриця ознак формувалася через конкатенацію оброблених числових та енкодованих категоріальних підматриць: числові послідовно проходили «Impute → Winsorize → Scale», а категоріальні – «Impute → Encode». Отримані вектори досягали розмірності від 71 до 241, відповідаючи вхідним вимогам базової нейронної мережі, що містить близько 2,7 ×104 параметрів і, отже, не створює обчислювального навантаження на середньосегментний GPU.

Висновки до розділу 2.

Сучасний розвиток фінансових технологій відбувається в умовах глибокої цифрової трансформації, де ключову роль відіграють Big Data та штучний інтелект (AI). Їхнє застосування у сфері децентралізованих фінансів (DeFi) відкриває нові можливості для аналітики, прогнозування та прийняття управлінських рішень. Особливістю DeFi є висока волатильність та відсутність централізованих механізмів регулювання, що підсилює потребу у використанні точних і масштабованих інструментів оцінки ринкової вартості активів та ризиків. У цьому

контексті технології Big Data та AI стають фундаментом для побудови ефективних моделей, які враховують як фінансові, так і поведінкові чинники.

Використання Big Data у фінансовій сфері дає змогу обробляти величезні масиви транзакційних даних із блокчейн-мереж, таких як Ethereum, Binance Smart Chain та Solana. Аналітика Total Value Locked (TVL), показників ліквідності, історичних патернів торгівлі та поведінкових характеристик інвесторів створює цілісну картину функціонування DeFi-екосистеми. Інструменти обробки даних, зокрема Apache Kafka, Hadoop і Spark, забезпечують обробку потоків у реальному часі, що дозволяє швидко реагувати на зміни ринку. Це особливо важливо у високонавантажених середовищах, де затримка навіть у кілька секунд може мати суттєвий вплив на інвестиційні рішення.

Штучний інтелект, у свою чергу, підсилює аналітичні можливості Big Data. Методи машинного та глибинного навчання, нейронні мережі та reinforcement learning застосовуються для виявлення складних закономірностей, прогнозування динаміки активів та оцінки інвестиційних ризиків. Використання алгоритмів Natural Language Processing (NLP) дозволяє враховувати інформаційні потоки з криптовалютних медіа та соціальних мереж, що є важливим чинником поведінкових фінансів. Таким чином, AI робить фінансові моделі більш гнучкими та здатними адаптуватися до швидких змін у DeFi-просторі.

Разом із тим, інтеграція Big Data та AI у фінансову сферу має низку викликів. Однією з головних проблем є якість даних: значна частина блокчейн-інформації є неструктурованою або дубльованою, що потребує додаткових механізмів очищення й верифікації. Другим важливим фактором є масштабованість — зростання кількості транзакцій у блокчейнах вимагає потужних обчислювальних ресурсів і спеціалізованих алгоритмів оптимізації. Також слід враховувати правові аспекти: регуляторна невизначеність у різних країнах може впливати на рівень довіри до DeFi-проєктів та обмежувати залучення інституційних інвесторів.

Перспективи розвитку Big Data та AI у DeFi залишаються надзвичайно широкими. Подальша розробка адаптивних алгоритмів прогнозування, інтеграція гібридних моделей та використання інноваційних методів обробки блокчейн-даних

здатні значно покращити точність фінансових оцінок. Це дозволить не лише знизити інвестиційні ризики, але й сформувати більш прозору та передбачувану фінансову екосистему. Зростання популярності DeFi серед роздрібних та інституційних інвесторів зумовлює необхідність удосконалення методів аналітики, що враховують не лише класичні фінансові індикатори, а й соціально-поведінкові аспекти.

Отже, інтеграція Big Data та AI у процеси аналізу та оцінки DeFi-проєктів є одним із ключових напрямів розвитку сучасної фінансової науки та практики. Використання цих технологій сприятиме підвищенню точності прогнозів, оптимізації управління ліквідністю та формуванню стійкої цифрової економіки. У перспективі це дозволить зміцнити довіру до децентралізованих фінансів і забезпечить їх активніше впровадження у глобальну фінансову систему.

РОЗДІЛ 3 РОЗРОБКА АЛГОРИТМУ ДЛЯ ВИЯВЛЕННЯ ШАХРАЙСКИХ ТРАНЗАКЦІЙ.

3.1 Архітектура запропонованої системи

У сучасних умовах розвитку фінансових технологій зростає кількість електронних транзакцій, що, у свою чергу, призводить до підвищення ризиків шахрайських дій. За даними міжнародних фінансових аналітичних центрів, понад 1% усіх електронних платежів у світі мають ознаки шахрайства. Тому створення ефективних систем виявлення шахрайських транзакцій є критично важливим завданням як для банків, так і для платіжних сервісів.

Традиційні системи виявлення шахрайства часто базуються на статичних правилах (наприклад, ліміти суми, частота операцій чи географічне обмеження). Проте такі підходи мають обмеження: вони не враховують адаптивність шахраїв і не здатні виявляти нові типи атак. У зв’язку з цим актуальним є застосування методів машинного навчання, які дозволяють аналізувати великі обсяги даних та виявляти приховані закономірності.

Основна мета алгоритму полягає у визначенні аномальних транзакцій у потоці фінансових операцій у режимі, максимально наближеному до реального часу.

Алгоритм виявлення шахрайських транзакцій можна поділити на кілька ключових етапів:

Використовуються історичні дані про транзакції:

- сума;
- час;
- місце здійснення;
- тип операції;
- ідентифікатори клієнта та пристрою;

Необхідним кроком є анонімізація даних з метою дотримання норм GDPR та локального законодавства.

Дані очищуються від шумів, аномалій та дублікатів. Проводиться категоризація (наприклад, перетворення країни або типу картки в числові ознаки), а також масштабування числових параметрів.

Формування ознак (feature engineering):

- Кількість транзакцій за певний проміжок часу;
- Відстань між геолокаціями останніх платежів;
- Співвідношення середньої витрати клієнта;

історичні патерни поведінки (час доби, у який найчастіше здійснюються платежі);

Вибір моделі машинного навчання

Для задачі класифікації ефективними є алгоритми:

- Random Forest – добре працює з нерівноважними вибірками;
- Gradient Boosting (XGBoost, LightGBM) – забезпечує високу точність при великому обсязі даних;
- Нейронні мережі (LSTM, Autoencoder) – особливо корисні для аналізу послідовностей транзакцій;

Важливим є використання методів балансування класів (наприклад, SMOTE), оскільки кількість шахрайських транзакцій зазвичай значно менша за кількість легальних.

Валідація та налаштування моделі

Застосовується крос-валідація та метрики якості: *Precision, Recall, F1-score, ROC-AUC*. У контексті банківських систем особливу увагу приділяють метриці *Recall*, адже пропуск шахрайської транзакції є значно дорожчим, ніж хибно позитивне спрацювання.

Онлайн-обробка транзакцій

Після навчання модель інтегрується у систему реального часу. Кожна транзакція оцінюється, і якщо ризик шахрайства перевищує порогове значення, вона блокується або передається на додаткову перевірку.

Архітектура запропонованої системи

Система виявлення шахрайських транзакцій має багаторівневу архітектуру, що поєднує модулі збору даних, обробки та прийняття рішень.

Рівень збору даних:

- Підключення до платіжних шлюзів та банківських систем;
- Інтеграція з історичними базами даних;
- Реєстрація метаданих транзакцій у режимі реального часу;

Рівень попередньої обробки:

- Очищення та нормалізація даних;
- Формування ознак на основі транзакційної історії;
- Кешування останніх операцій клієнта для швидкої обробки;

Аналітичний рівень (модуль виявлення аномалій):

- Модель машинного навчання для класифікації транзакцій;
- Блок нейронних мереж для аналізу поведінкових патернів;
- Адаптивний механізм оновлення моделі на основі нових даних;

Рівень прийняття рішень:

- Класифікація транзакції як «безпечна», «підозріла» або «шахрайська»;
- Автоматичне блокування підозрілих операцій;

- Перенаправлення даних до аналітика для ручної перевірки у випадку сумнівів;

Інтерфейс моніторингу:

- Дашборди для відображення статистики;
- Система сповіщень для співробітників банку;
- Модуль аудиту та журналювання подій;

Переваги та перспективи розвитку

Запропонована система поєднує гнучкість алгоритмів машинного навчання з високою швидкістю обробки транзакцій.

Основними перевагами є:

- Зменшення кількості хибнопозитивних спрацьовувань;
- Здатність адаптуватися до нових типів шахрайських схем;
- Масштабованість для обробки мільйонів транзакцій на добу;

У перспективі можливе використання глибинного навчання для аналізу графових структур транзакцій (Graph Neural Networks), а також інтеграція з блокчейн-технологіями для забезпечення додаткової прозорості та безпеки.

Розробка алгоритму для виявлення шахрайських транзакцій є складним, але надзвичайно актуальним завданням у сфері фінансових технологій. Запропонована архітектура системи забезпечує ефективний баланс між точністю виявлення шахрайства та швидкістю обробки даних. Використання сучасних методів машинного навчання та побудова модульної архітектури дозволяють забезпечити надійну роботу системи та підвищити рівень безпеки електронних платежів.

3.2 Формування навчальної вибірки (балансування класів, попередня обробка даних)

У сфері кібербезпеки, що швидко розвивається, машинне навчання (ML) відіграє вирішальну роль у покращенні виявлення та пом'якшення загроз. Ефективні методи доповнення даних, такі як SMOTE та ADASYN, є важливими для усунення дисбалансу класів у наборах даних кібербезпеки. Однак простого балансування наборів даних та навчання моделей ML недостатньо, якщо ці моделі також не можуть призначати ваги ознакам набору даних. Розуміння важливості

ознак є ключовим для визначення того, які фактори найбільше впливають на реакції моделі, тим самим покращуючи інтерпретованість та ефективність моделі. У цьому розділі досліджуються різні методи доповнення даних, моделі навчання з учителем та значення зважування ознак у контексті кібербезпеки, пропонуючи розуміння оптимізації продуктивності моделі та прийняття рішень.

Для збагачення набору даних без втрати цінної інформації також можна використовувати передові методи, такі як генерація синтетичних даних за допомогою таких методів, як SMOTE (техніка синтетичної меншості надмірної вибірки [23]). Враховуючи вибірку xi з класу меншин, SMOTE визначає своїх k-найближчих сусідів у просторі ознак. Нехай xnn позначаємо одного з цих k-найближчих сусідів. Синтетичний зразок $x$new генерується шляхом інтерполяції між xi і xnn використовуючи рівняння:

$$Xnex=Xi+ \lambda (Xnm-Xi) \qquad (1.5)$$

де λ – випадкове число між 0 та 1. Цей крок інтерполяції створює нову вибірку, яка є лінійною комбінацією вихідної вибірки та її сусіда, таким чином зберігаючи загальний розподіл даних, розширюючи при цьому клас меншості. ADASYN (Адаптивна синтетична вибірка, [24]) розширює SMOTE, зосереджуючись більше на генерації синтетичних зразків поруч зі зразками класу меншин, які помилково класифіковані класифікатором. Математично ADASYN обчислює кількість синтетичних зразків для генерації для кожного зразка класу меншин xi використовуючи розподіл щільності:

$$ri= \gamma i/ (\sum N^* \ i =1) * \gamma i \qquad (1.6)$$

де γi – кількість вибірок мажорного класу серед k-найближчих сусідів xi Кількість синтетичних зразків Gi генерувати для кожного xi пропорційна ri. У випадках, коли класи суттєво перетинаються, SMOTE та ADASYN можуть вводити

синтетичні зразки в області, де класи недостатньо розділені. Це може призвести до збільшення кількості неправильних класифікацій, оскільки синтетичні зразки в областях, що перекриваються, можуть бути неправильно класифіковані, що знижує загальну продуктивність моделі. Крім того, введення синтетичних зразків в області, що перекриваються, може розмити межі прийняття рішень, що ускладнить для класифікатора розрізнення між класами. Коли синтетичні зразки вносять шум, як SMOTE, так і ADASYN можуть страждати від зниження продуктивності. Зокрема, синтетичні зразки, які не є репрезентативними для фактичного розподілу даних, можуть вносити шум, що призводить до поганого узагальнення моделі. Крім того, наявність шумних зразків може призвести до надмірного налаштування моделі на синтетичні дані, знижуючи її здатність добре працювати з невидимими даними. Крім того, шум може негативно вплинути на точність і повноту, оскільки модель може давати більше хибнопозитивних і хибнонегативних результатів. Однак, для зменшення проблем перекриття класів та шуму можна застосувати кілька стратегій. Наприклад, дані можна попередньо обробити для видалення шуму перед застосуванням SMOTE або ADASYN. По-друге, можна використовувати більш складні методи (такі як Borderline-SMOTE або SVM-SMOTE), які зосереджені на генерації вибірок поблизу межі прийняття рішення. Нарешті, можна постійно оцінювати продуктивність та налаштовувати параметри методів надмірної вибірки, щоб мінімізувати внесення шуму. Метод Borderline-SMOTE [25] спеціально орієнтований на вибірки класів меншин, які знаходяться близько до межі з класом більшості. Він використовує ту саму стратегію інтерполяції, що й SMOTE, але обмежує її тими вибірками меншин, найближчі сусіди яких включають вибірки класів більшості. Формула генерації синтетичних вибірок залишається такою ж, як і в методі SMOTE. Доповнення даних за допомогою зв'язків Томека [26] – це метод, який використовується переважно для підвищення продуктивності класифікаторів на незбалансованих наборах даних. Він включає ідентифікацію пар екземплярів, які є найближчими сусідами, але належать до різних класів, та їх видалення для збільшення роздільності класів. Між парою екземплярів існує зв'язок Томека xi і xj з різних класів, якщо немає екземпляра xk такий, що $d(xi,xk)<d(xi,xj)$, де d

представляє використовувану метрику відстані, часто евклідову відстань. Математично її можна охарактеризувати так .

Цей метод особливо ефективний для задач бінарної класифікації та часто використовується як метод очищення даних, а не як метод передискретизації. SMOTEENN [27] поєднує два підходи для вирішення проблеми дисбалансу класів у наборах даних машинного навчання: SMOTE (техніка синтетичної меншості надмірної вибірки) для надмірної вибірки класу меншості та ENN (відредагований найближчий сусід) для очищення даних шляхом недостатньої вибірки обох класів. ENN видаляє будь-який зразок, у якого більшість k-найближчих сусідів належать до іншого класу. Для заданого зразка xi, його видаляють, якщо де I – індикаторна функція, yj – мітка класу j-го найближчого сусіда, а yi є міткою класу xi. SMOTEENN застосовує SMOTE для генерації синтетичних зразків, а потім використовує ENN для видалення будь-яких згенерованих або оригінальних зразків, які неправильно класифіковані їхніми найближчими сусідами. Ця комбінація допомагає уточнити межі класів далі, ніж використання лише SMOTE. Нарешті, SMOTE-Tomek [28] – це гібридний метод, який поєднує підхід SMOTE для надмірної вибірки класу меншості з зв'язками Томека для очищення перекриваючихся вибірок між класами. Цей метод особливо ефективний для покращення класифікації незбалансованих наборів даних, як шляхом розширення класу меншості, так і шляхом підвищення роздільності класів. SMOTE-Tomek спочатку застосовує SMOTE для створення додаткових синтетичних вибірок для балансування розподілу класів. Згодом він застосовує метод зв'язків Томека для видалення будь-яких зв'язків Томека, виявлених між синтетичними та вихідними вибірками. Цей процес видалення допомагає зменшити шум і зробити класи більш чіткими, що корисно для подальшого процесу навчання. Цікаво обговорити обчислювальні витрати методів доповнення даних та їхній вплив на модель. Часова складність SMOTE становить $O(T\times k\times d)$, де T – кількість синтетичних зразків, k – кількість найближчих сусідів, а d – розмірність даних. Це призводить до помірного збільшення часу навчання через необхідність пошуку найближчих сусідів та

генерації синтетичних зразків. ADASYN має подібну часову складність до SMOTE, але включає додаткові обчислення для визначення складності екземплярів, що призводить до дещо вищих обчислювальних витрат та додаткового часу обробки порівняно зі SMOTE. Borderline SMOTE має таку ж часову складність, як і SMOTE, але з додатковими кроками для ідентифікації граничних зразків. Це призводить до вищих обчислювальних витрат, оскільки ідентифікація граничних зразків вимагає додаткових обчислень. Часова складність для Tomek Links становить $O(n2\times d)$, де n– кількість зразків, а d – розмірність. Це призводить до значного часу попередньої обробки через необхідність обчислення попарних відстаней між зразками, що впливає на загальний час навчання моделі. SMOTEENN поєднує витрати SMOTE та методу відредагованих найближчих сусідів (ENN), зазвичай $O(T\times k\times d)+O(n\times k)\times d)$. Високі обчислювальні витрати є результатом поєднання генерації синтетичних зразків та очищення за методом найближчого сусіда, що призводить до значного збільшення використання ресурсів. SMOTE Tomek поєднує витрати SMOTE та Tomek Links, зазвичай $O(T\times k\times d)+O(n2\times d$). Таке поєднання призводить до дуже високих обчислювальних витрат через значні попарні обчислення відстаней та генерацію синтетичних вибірок, що суттєво впливає на час навчання та споживання ресурсів.

Обчислювальні витрати цих методів безпосередньо впливають на час навчання моделі та використання ресурсів. Методи з вищою часовою складністю, такі як SMOTE Tomek та SMOTEENN, значно збільшують час попередньої обробки та подовжують загальний час навчання. Високі обчислювальні витрати призводять до збільшення використання процесора та пам'яті, що може бути обмежуючим фактором для великих наборів даних або складних моделей. Методи з квадратичною часовою складністю (наприклад, Tomek Links) можуть погано масштабуватися з великими наборами даних. Доповнення даних має вирішальне значення в кібербезпеці для створення більш повних наборів даних, які можуть допомогти в кращому навчанні моделей машинного та глибокого навчання. У

науковій літературі пропонуються різні дослідження, зосереджені навколо моделей машинного навчання, що застосовуються в кібербезпеці.

3.3 Вибір та обгрунтування алгоритмів машинного навчання

Для побудови системи рекомендацій було обрано Million Song Dataset з додатковими даними від Spotify та Last.fm [30]. Цей інтегрований набір даних містить інформацію про музичні треки (аудіо-характеристики, жанри, теги) та історію прослуховувань користувачів. Він сформований на основі кількох джерел: власне Million Song Dataset, підмножини Taste Profile від Echo Nest (дані про взаємодії "користувач-трек"), відкритого набору Last.fm 2020 (теги та додаткові метадані треків) та даних Spotify API (аудіо-фічі треків, посилання на прослуховування тощо) [30]. В результаті було отримано підмножину, що включає 50 683 унікальних треків та близько 9,7 млн записів історії прослуховувань користувачів [30]. Такий багатий набір даних забезпечує наявність як колаборативної інформації (взаємодії користувачів з треками), так і контентних характеристик треків, що є критично важливим для побудови персоналізованих рекомендацій та особливо корисним при реалізації гібридних моделей.

Структура даних. Кожен трек у датасеті характеризується рядом атрибутів, зокрема: назва, виконавець, рік випуску, тривалість, жанрові категорії та популярні мітки (теги) Last.fm, а також числові аудіо-фічі, отримані через API Spotify (темп, гучність, тональність, валентність, енергійність, танцювальність, акустичність тощо). Додатково для треків доступні жанрові анотації з набору tagtraum (позначення належності до одного або кількох з 15 верхньорівневих жанрів). Інша частина даних – це історія прослуховувань користувачів, яка представлена як множина пар (користувач, трек) з відповідними лічильниками прослуховувань. Ці взаємодії є неявними відгуками: факт прослуховування інтерпретується як позитивний сигнал про вподобання треку, тоді як відсутність прослуховувань розглядається як відсутність явного інтересу (неявний негативний сигнал). Усього у вибраній підмножині беруть участь сотні тисяч анонімних користувачів (ідентифікованих за ID), кожен з яких прослухав принаймні кілька треків із

наявного каталогу. Інструменти реалізації. Розробку системи здійснено в середовищі Jupyter Notebook мовою Python. Для обробки даних використовувалися бібліотеки pandas і NumPy, що забезпечують зручне завантаження та маніпулювання табличними даними. Для виконання алгоритмів машинного навчання залучено низку спеціалізованих бібліотек: зокрема, scikit-learn (реалізація алгоритму kближніх сусідів), бібліотека LightFM для побудови гібридної моделі матричної факторизації, та фреймворк PyTorch (разом з додатковими модулями) для реалізації графової нейронної мережі LightGCN. Вибір цих інструментів обумовлений їхньою ефективністю на розріджених даних та наявністю готових реалізацій потрібних алгоритмів, що прискорює процес експериментальної перевірки моделей.

3.4 Реалізація алгоритмів та інтеграція з банківською системою

Штучний інтелект є однією з найперспективніших технологій, що здатна кардинально трансформувати різні сфери, зокрема і банківський сектор. Завдяки можливості аналізувати великі масиви даних, виявляти закономірності та приймати обґрунтовані рішення, штучний інтелект змінює підхід до банківських послуг, підвищує рівень клієнтського обслуговування та сприяє автоматизації операцій (Біленко, Сивицька & Теленкова, 2018).

За даними Business Insider (Business Insider, 2024), близько 80% банків визнають значний потенціал штучного інтелекту у фінансовій сфері. Дослідження McKinsey (McKinsey & Company, 2024) свідчить, що генеративні моделі штучного інтелекту можуть підвищити продуктивність у банківському секторі на 5% і скоротити загальні витрати на 300 млрд дол США. Такі тенденції підтверджують, що фінансові установи активно впроваджують штучний інтелект, прагнучі підвищити ефективність, зменшити витрати та покращити клієнтський досвід.

Серед ключових напрямків використання штучного інтелекту в банківській сфері визнано боротьбу з шахрайством та персоналізацію послуг. Наприклад, Bank of America запустив чат-бот Erica (Bank of America, 2024), який з 2018 р. обробив понад 2 млрд запитів, надаючи клієнтам цілодобову підтримку та оптимізуючи фінансові операції, що суттєво скорочує час очікування та покращує рівень

задоволеності користувачів. Банк Великої Британії Barclays (Barclays, 2024), застосовує штучний інтелект для виявлення шахрайських дій, де аналізуються платіжні транзакції в реальному часі, що допомагає миттєво реагувати на підозрілі операції, забезпечуючи захист клієнтів і підвищуючи рівень довіри до банку.

В Україні штучний інтелект також кардинально змінює банківську діяльність, підвищуючи ефективність, зменшуючи витрати та покращуючи досвід клієнтів. Завдяки алгоритмам машинного навчання та аналізу великих даних українські банки можуть швидше ухвалювати рішення, виявляти шахрайство та персоналізувати фінансові послуги. Інтеграція штучного інтелекту у банківську діяльність здатна суттєво покращити ефективність роботи фінансових установ, підвищити рівень безпеки, персоналізувати обслуговування клієнтів та знизити операційні витрати (МФУ, 2024). Завдяки використанню алгоритмів машинного навчання банки можуть швидше та точніше аналізувати великі обсяги даних, що дозволяє оперативно виявляти шахрайські дії та підозрілі фінансові операції. Штучний інтелект також значно покращує взаємодію з клієнтами через чат-боти та віртуальні асистенти, які можуть надавати консультації в режимі 24/7, відповідати на запитання клієнтів, допомагати з вибором банківських послуг та супроводжувати користувачів у процесі оформлення кредитів або відкриття рахунків, що скорочує навантаження на працівників банку, даючи їм змогу зосередитися на більш складних завданнях.

Крім того, штучний інтелект використовується для персоналізації банківських послуг, аналізуючи фінансові звички клієнтів і пропонуючи їм найбільш релевантні продукти та послуги. Наприклад, система може запропонувати кредитну лінію з оптимальними умовами або нагадати про необхідність сплати рахунків, враховуючи індивідуальні особливості витрат. Безпека фінансових операцій також покращується завдяки використанню штучного інтелекту. Багато банків впроваджують біометричні методи автентифікації, такі як розпізнавання обличчя або відбитків пальців, що дозволяє захистити клієнтів від шахрайства. Крім того, аналіз поведінки користувачів у мобільному та онлайн-банкінгу дає змогу виявляти аномалії, які можуть свідчити про компрометацію

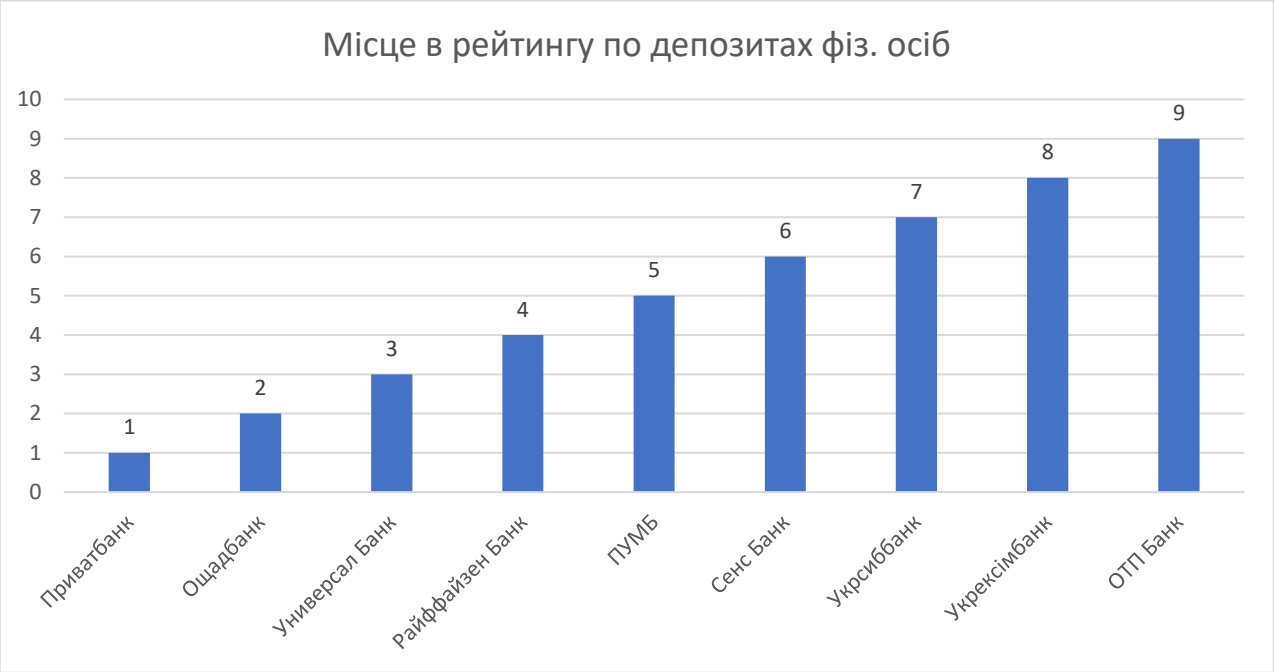
акаунта. Також алгоритми штучного інтелекту здатні аналізувати ринкові дані та передбачати потенційні фінансові ризики, що допомагає банкам ухвалювати більш обґрунтовані кредитні рішення, знижуючи ймовірність невиплат та дефолтів (Руда, 2024).

Таким чином, інтеграція штучного інтелекту у банківську діяльність забезпечує значне підвищення швидкості та точності операцій, покращує обслуговування клієнтів, зміцнює безпеку фінансових транзакцій і сприяє розвитку персоналізованих фінансових рішень, що робить банки більш конкурентоспроможними та технологічно розвиненими, відповідаючи сучасним викликам цифрової економіки.

Аналізуючи сучасний стан банківської системи України за даними Міністерства фінансів України (МФУ, 2024) можна визначити рейтинг стійкості банків (рис.) та місце банку за депозитами фізичних осіб (рис.) за III кв. 2024 р. До рейтингу увійшли банки з портфелями роздрібних депозитів від 1 млрд грн.[31]

Рис. 6 Загальний рейтинг стійкості банків *Джерело [31]*

Рис. 7 Місце в рейтингу по депозитах фізичних осіб *Джерело [31]*



Висновки до розділу 3

У результаті комплексного теоретико-аналітичного дослідження встановлено, що технології штучного інтелекту набувають системоутворювального значення у процесі трансформації сучасної банківської системи. Їх застосування перестає розглядатися виключно як інструмент оптимізації окремих операційних процесів і дедалі більше позиціонується як стратегічний чинник забезпечення фінансової стійкості, конкурентоспроможності та адаптивності банківських установ до динамічних змін зовнішнього середовища. У цьому контексті впровадження інтелектуальних алгоритмів є об'єктивною необхідністю, зумовленою зростанням обсягів даних, ускладненням фінансових продуктів та підвищенням вимог з боку споживачів фінансових послуг.

Одним із ключових напрямів використання штучного інтелекту в банківській діяльності є система протидії фінансовому шахрайству та зловживанням. Моделі машинного навчання, що ґрунтуються на аналізі великих масивів транзакційної інформації, забезпечують ідентифікацію аномальних фінансових операцій у режимі

реального часу. Це дозволяє не лише мінімізувати прямі фінансові втрати, а й підвищити загальний рівень довіри клієнтів до банківських інститутів. Для банківської системи України, яка функціонує в умовах підвищених кіберзагроз і активного розвитку дистанційних каналів обслуговування, зазначений аспект має особливо вагоме значення.

Вагомим напрямом застосування інтелектуальних технологій є індивідуалізація банківських продуктів і сервісів. Використання аналітики великих даних у поєднанні з поведінковими моделями клієнтів створює передумови для формування персоналізованих фінансових пропозицій, що відповідають платоспроможності, ризиковому профілю та фінансовим цілям конкретного користувача. Такий підхід забезпечує підвищення якості обслуговування, сприяє зростанню клієнтської лояльності та формує довгострокову цінність взаємовідносин між банком і споживачем фінансових послуг.

Окремої уваги заслуговує вплив штучного інтелекту на оптимізацію внутрішніх бізнес-процесів банківських установ. Запровадження автоматизованих систем обслуговування, зокрема віртуальних асистентів і чат-ботів, сприяє зниженню операційних витрат та підвищенню ефективності використання трудових ресурсів. У результаті рутинні операції делегуються автоматизованим платформам, тоді як людський капітал зосереджується на виконанні завдань, що потребують аналітичного мислення та професійного судження. Це позитивно позначається на швидкості обслуговування клієнтів і загальному рівні операційної результативності банку.

Застосування штучного інтелекту має також істотне значення у сфері управління фінансовими ризиками та прийняття управлінських рішень. Аналітичні моделі прогнозування дозволяють оцінювати кредитоспроможність позичальників, моделювати сценарії розвитку макроекономічної ситуації та ідентифікувати потенційні ризики на ранніх етапах. Це забезпечує більш обґрунтований підхід до формування кредитної політики та сприяє зниженню рівня проблемної заборгованості. Інтеграція таких моделей із сучасними методами біометричної ідентифікації формує комплексну систему фінансової безпеки.

Адаптація українських банків до глобальних тенденцій цифрової трансформації передбачає врахування досвіду провідних міжнародних фінансових інститутів, які вже досягли значних результатів у впровадженні інтелектуальних технологій. Водночас специфіка національного фінансового ринку, зокрема високий рівень цифровізації населення та активне використання мобільних банківських сервісів, створює можливості для формування власних інноваційних рішень, орієнтованих на локальні потреби.

У довгостроковій перспективі очікується подальше розширення використання генеративних моделей штучного інтелекту, здатних автоматизувати складні аналітичні процеси, підтримувати стратегічне планування та формувати сценарні прогнози розвитку фінансового сектору. Це потенційно дозволить підвищити ефективність функціонування банківської системи в цілому та зменшити витрати на управлінські процеси.

Водночас інтеграція штучного інтелекту в банківську діяльність супроводжується низкою викликів, серед яких ключовими є забезпечення кібербезпеки, захист персональних даних, дотримання принципів етичного використання алгоритмів і відповідність регуляторним вимогам. Для українських банків актуальним залишається завдання формування збалансованого підходу до впровадження інноваційних технологій із урахуванням чинної нормативно-правової бази.

Отже, результати дослідження дозволяють зробити висновок, що застосування штучного інтелекту в банківській системі є стратегічно важливим чинником її подальшого розвитку. Інтелектуальні технології формують якісно нову модель банківського обслуговування, орієнтовану на підвищення ефективності, безпеки та клієнтоцентричності. Для України реалізація потенціалу штучного інтелекту у фінансовій сфері може стати вагомим чинником зміцнення стійкості банківської системи та її інтеграції в глобальний цифровий економічний простір.

Висновки

Підсумкові результати роботи

Проведене дослідження дозволило комплексно розглянути проблему фінансового шахрайства, сучасні методи його виявлення, а також перспективи застосування інноваційних технологій у банківській сфері. Узагальнюючи висновки трьох розділів, можна сформувати цілісне уявлення про ключові тенденції та напрями подальшого розвитку фінансової безпеки.

Насамперед було з’ясовано, що фінансове шахрайство є багатовимірним явищем, яке охоплює як класичні схеми (підробка документів, кредитні маніпуляції, змови працівників), так і новітні форми, пов’язані з цифровими технологіями. Стрімка цифровізація суспільства створює нові можливості для шахраїв, які комбінують технічні й психологічні методи впливу. Це потребує від фінансових установ комплексного підходу до протидії, що включає правові, організаційні, освітні та технологічні механізми.

Ключовим об’єктом дослідження стали банківські транзакції, адже саме вони є основою фінансової системи. Їх класифікація за різними критеріями дозволяє ідентифікувати ризики, що супроводжують кожен вид операцій. Було встановлено, що кредитний, валютний, операційний та репутаційний ризики тісно пов’язані зі структурою транзакцій, а їхнє усвідомлення допомагає своєчасно впроваджувати механізми захисту. У цьому контексті важливим є поєднання організаційно-контрольних, аналітичних та технологічних методів, які у своїй сукупності формують систему багаторівневого захисту від шахрайських дій.

Дослідження продемонструвало, що найпоширенішими видами шахрайських схем залишаються документальні підробки, касові маніпуляції та зловживання у кредитуванні. Це підкреслює значення внутрішнього аудиту та контролю в банках, який має виконувати не формальну, а запобіжну функцію. Лише постійний моніторинг і своєчасне реагування на відхилення здатні забезпечити стійкість банківської діяльності.

Особливу увагу приділено аналізу сучасних технологій виявлення аномалій. Було встановлено, що методи машинного та глибинного навчання є ефективними

інструментами для ідентифікації підозрілих фінансових операцій. Автоенкодери, рекурентні (LSTM) та згорткові (CNN) нейронні мережі показують високу результативність у роботі з різними типами даних. Найбільш перспективними визнано гібридні моделі, які дозволяють досягти балансу між точністю й швидкістю обробки інформації.

У другому розділі дослідження наголошено на значенні Big Data та штучного інтелекту у сфері децентралізованих фінансів (DeFi). Застосування цих технологій відкриває нові можливості для аналітики, прогнозування та управління ризиками в умовах відсутності централізованого регулювання. Важливим результатом є підтвердження того, що обробка великих обсягів транзакційних даних у блокчейн-мережах забезпечує прозорість, швидкість реакції на зміни та підвищення точності фінансових прогнозів. Методи машинного навчання та NLP дозволяють враховувати інформаційні сигнали з ринкових та соціальних медіа, що робить фінансові моделі більш гнучкими й адаптивними.

Разом із тим, інтеграція Big Data та AI у DeFi супроводжується низкою викликів, серед яких — якість даних, їхня масштабованість і правова невизначеність. Проте перспективи їх розвитку залишаються надзвичайно широкими, зокрема завдяки впровадженню адаптивних та гібридних алгоритмів прогнозування. Це свідчить про формування нової парадигми фінансової аналітики, здатної враховувати як класичні економічні індикатори, так і поведінкові аспекти.

У третьому розділі було проаналізовано вплив штучного інтелекту на трансформацію банківської системи. Доведено, що ШІ є стратегічно важливим фактором для розвитку фінансової сфери, оскільки дозволяє одночасно підвищити ефективність, безпеку та клієнтоорієнтованість послуг. Алгоритми машинного навчання використовуються для виявлення шахрайських транзакцій, персоналізації банківських продуктів та оптимізації внутрішніх бізнес-процесів. Використання чат-ботів і віртуальних асистентів забезпечує цілодобову підтримку клієнтів, тоді як аналітичні системи дозволяють прогнозувати ризики та ухвалювати більш обґрунтовані рішення.

Значна увага приділена й викликам, що супроводжують інтеграцію ШІ. Питання кібербезпеки, захисту персональних даних та етичності алгоритмів залишаються пріоритетними для банківських установ. Саме баланс між інноваціями та регуляторними вимогами стане запорукою успішної цифрової трансформації українських банків.

Узагальнюючи результати роботи, можна зробити висновок, що протидія фінансовому шахрайству потребує комплексного підходу, який об’єднує класичні інструменти контролю та сучасні технології аналізу. Інтеграція штучного інтелекту й Big Data у банківську діяльність відкриває нові горизонти для управління ризиками, підвищення рівня довіри клієнтів та формування конкурентних переваг. У перспективі саме поєднання інновацій, регуляторної бази та міжнародної співпраці стане основою для побудови стійкої, безпечної та технологічно розвиненої фінансової системи України.

Подальші напрями досліджень

На основі проведеного дослідження можна визначити кілька перспективних напрямів для подальшого наукового пошуку та практичних розробок у сфері протидії фінансовому шахрайству та забезпечення стійкості банківської системи.

По-перше, актуальним є поглиблене дослідження гібридних моделей виявлення шахрайських транзакцій, що поєднують класичні статистичні методи, правила експертних систем і алгоритми глибинного навчання. Такий підхід дозволить підвищити точність виявлення аномалій і мінімізувати кількість хибнопозитивних результатів, які є однією з ключових проблем сучасних аналітичних систем.

По-друге, потребує розвитку напрям адаптивних моделей штучного інтелекту, здатних навчатися в режимі реального часу. Це особливо важливо в умовах швидкої еволюції шахрайських схем, коли традиційні алгоритми не встигають за динамікою змін. Використання технологій reinforcement learning може стати перспективним шляхом підвищення гнучкості та ефективності систем безпеки.

По-третє, перспективним напрямом досліджень є інтеграція блокчейн-технологій у систему фінансового моніторингу. Використання децентралізованих реєстрів може забезпечити прозорість операцій, зменшити ризик маніпуляцій з даними та створити єдину базу для відстеження підозрілих транзакцій як на національному, так і на міжнародному рівні.

По-четверте, важливим завданням стає дослідження поведінкових фінансів у контексті протидії шахрайству. Аналіз психологічних і соціальних факторів, що впливають на рішення клієнтів, у поєднанні з інструментами Big Data та Natural Language Processing дозволить будувати більш точні профілі користувачів і виявляти нетипові дії, характерні для шахрайських операцій.

По-п’яте, доцільним є розроблення національної платформи обміну даними про шахрайські транзакції між банками, фінтех-компаніями та державними органами. Така платформа сприятиме швидшому реагуванню на загрози, підвищенню рівня довіри у фінансовому секторі та зміцненню міжнародної співпраці в боротьбі з кіберзлочинністю.

По-шосте, особливої уваги заслуговують правові та етичні аспекти використання штучного інтелекту у фінансовій сфері. Подальші дослідження мають бути спрямовані на розроблення регуляторних механізмів, що гарантуватимуть прозорість алгоритмів, захист персональних даних і дотримання принципів справедливості при ухваленні автоматизованих рішень.

По-сьоме, важливим напрямом розвитку є дослідження кіберстійкості банківських систем, зокрема оцінка їхньої здатності протистояти комплексним атакам, що поєднують фінансові шахрайства з кібератаками на інфраструктуру. Інтеграція системи захисту від шахрайства з сучасними інструментами кібербезпеки дозволить створити багаторівневу оборону.

Таким чином, подальші дослідження у сфері протидії фінансовому шахрайству повинні бути міждисциплінарними та поєднувати економічний, технологічний, правовий і поведінковий підходи. Реалізація зазначених напрямів дозволить не лише підвищити ефективність виявлення та запобігання шахрайським

операціям, а й сприятиме формуванню стійкої фінансової екосистеми України в умовах глобальної цифрової економіки.

СПИСОК ВИКОРИСТАННИХ ДЖЕРЕЛ

1.Вплив цифрової трансформації на безпеку банківської системи України :ризики та шляхи їх мінімізації.

ТР Андрієць-2025-reposit.nupp.edu.ua.

2.Система виявлення шахрайських дій під час онлайн-платежів на основі технологій машинного навчання.

ВІ Рудик-2025-elartu.tntu.edu.ua.

3.Електронне шахрайство :поняття та основні види.

АВ Панов ,ЛО Панова ,АЮ Трачук-The 12th international…,2025-books google.com.

4.Протидія фінансовому шахрайству в Україні.

НР Жигайло ,ОВ Нагірна ,ГВ Миськів – dspace .Ivduvs .edu .ua.

Безділь НБ.,Данілюк ММ.

5.Ідентифікація та мінімізація ризиків фінансового шахрайства під час прийняття управлінських рішень.

https://economyandsociety.in.ua /index.php/journal/article/view/4289/4214.

6.Платіжна система України: аналіз та динаміка шахрайских операцій.

О Любкина,В Тараба-Економіка та суспільство,2025-economyandsociety.in.ua.

7.Новини платіжного ринку від ЄМА.Українська міжбанківська асоціація членів платіжних систем ЄМА.

URL:https://www.ema.com.ua/news/(дата звернення 05.02.2025)

8.Дайджест новин ЄМА-2023рік.Українська міжбанківська асоціація членів плаеіжних систем ЄМА.

URL:https://www.ema.com.ua/news/2762(дата звернення 13.02.2025)

9.Порівняльний аналіз методів виявлення аномалій у наборах даних.

О Левченко-biblio.umsf.dp.ua

10.Застосування моделей машинного навчання для раннього виявлення надзвичайних сітуацій на основі потокових.

О Шопський,Р Головатий-2025-sci.ldubgd.edu.ua

11.Malhotra , P .,Vig,L.,Shroff,G.,&Agarwal,P.(2015,April).

Long short-term memory networks for anomaly detection in time series.In Proceedings of the 23rd European Symposium on Artificial Neural Networks(ESAAN)(pp 89-94)https://arviv.org/ats/1502.06690

12.Big Data та штучний інтелект у процесі оцінки De Fi-проектів.

ПВ Івахно-2025-repository.kpi.kcharkov.ua

13.Метод підвищення ефективності мережевих комунікацій за допомогою Apache Kafka в розподілених системах.

ДГ Гиценко-2025-ela.kpi.ua

14.Jun Rao Jay Kreps,Neha Narkheda.Kafka:A distributed messaging system for log processing.In Proctssings of 6th International Workshop on Networking Meets Databases(NetDB),Athens,Greece,pages 1-7,2011

15.Kafka InsideKeystone Pipeline.

https://medium.com/netflix-techblog/Kafka-inside-keystone-pipeline-dd5aebaf6bb/

16.Anarticle an the analysis of factorsthat ehsure high productivity Apache Kafka.https://habr.com./eb/companres/slurm/articles/530498

17.Роспызнавання програм-вимагачів за допомогою машинного навчання.ВВ Ярощук-2025-ela.kpi.ua.

18.Varonis.The State of Ransomware 2024[Електронний ресурс]-2024-Режим доступу:https://www.sophos.com/en-content/state-of-ransom ware(дата звернення 16.04.2025)

19.Hoxhunt.Phishing Trends Report(Updated2024)[Електронний ресурс]-2024-Режим доступу:https://hoxhunt.com/phishing-trends-report(дата звернення 16.04.2025)

20.Порівняльний аналіз ефективності нейронних мереж та ансамблевих методів у прогнозуванні цін криптовалют.

Ю Клебан,Є Конощук-…March 04-07,2025 Hamburg,Germany;2025 books.google.com

21.Xue Y.,Y,M.,Liang L. Credit scoring model-aced.onensemble machine learning method//Expert Systems with Applications.2018.Vol.94.P.149-156.

22.Методи збільшення даних для покращення контрольованого навчання при виявленні кібератак.

ІС Баєв-2025-openarchive.nure.ua.

23.Fernandez,A.;Garcia,S.;Herrera,F.;Chawla,N.V. SMOTE for learning from imbalanced data:Progress and challenges,marking the 15-year anniversary.J.Artif.Intell.Res.2018,61,863-905.

24.He,H.,Bai,Y.;Garcia,E.A.;Li,S. ADASYN:Adaptive synthetic samling approach for imbalanced learning.In Proceedings of the 2008 IEEE International Congress Jont Conference on Neural Networks(IEEE World Congress on Computational Intelligence),Hong Kong,China,1-8June 2008

25.Han,H.;Wong,W.Y.;Mao,B.H. Berdeline-SMOTE new oversampling method in imbalanced.data sels learning.In Proceedings of the Internat ional Conference on Intelligent Computing,Hefei,China,23-26 August 2005; Springer: Berlin/Heidelberg, Germany,2005;pp 878-887

26.Swana,E.F. Doorsamy,W.; Bokoro,P.Tomek link and SMOTE approaches for, machine fault classification with an inbalansed dataset.Sensors 2022,22,3246

27.He,H.;Garcia,E.A. Learning from imbalanced data.IEEE. Trans.Knowl.Data Eng,2009.21,1263-1284.

28.Yang,H.;Li,M. Software Defect Prediction Based on SMOTE-Tomek and X6 Bost,In Bio-Inspired Computing.Theories and Applications;Pan,L;Cni,Z.,Cai,J;Li,L;Eds.; Springer: Singapore,2022;pp.12-31.

29.Система персоналізованих рекомендацій на основі алгоритмів машинного навчання.ПЄ Лобарєв-2025-ela.kpi.ua

30.Million Song Dataset+Spotify+Last. Fm// Kaggle.
URL:https://www.kaggle.com/datasets/and efinenull//million-song-dataset-spotifylastfm(дата звернення 02.06.2025)(Kaggle)
31.Особливості інтеграції штучного інтелекту у банківську систему України.П Пузирьова-Вчені записки Університету «КРОК»; 2025-snku,krok.edu.ua

ДОДАТКИ

Додаток А -Фрагмент програмного коду

```
#!/usr/bin/env python3
"""
async_image_downloader.py
Простий асинхронний завантажувач зображень з підтримкою:
- одночасних з'єднань (semaphore)
- повторних спроб (retries)
- таймаутів
- логування
- збереження з оригінальним/новим іменем

Використання:
    python async_image_downloader.py urls.txt output_dir --concurrency 8 --retries 3
"""
import asyncio
import aiohttp
import aiofiles
import logging
import sys
from pathlib import Path
```

```
from argparse import ArgumentParser
from typing import List

# --- Налаштування логування ---
logging.basicConfig(
    level=logging.INFO,
    format="[%(asctime)s] %(levelname)s: %(message)s",
    datefmt="%Y-%m-%d %H:%M:%S",
)
logger = logging.getLogger("imgdl")

# --- Константи ---
DEFAULT_CONCURRENCY = 5
DEFAULT_RETRIES = 2
TIMEOUT = aiohttp.ClientTimeout(total=20)  # загальний таймаут на сесію/запит


async def fetch(session: aiohttp.ClientSession, url: str, sem: asyncio.Semaphore, retries: int) -> bytes | None:
    """Скачує вміст URL з підтримкою семафору та retry-логіки."""
    attempt = 0
    while attempt <= retries:
        attempt += 1
        try:
            async with sem:  # обмеження одночасних запитів
                logger.debug(f"Fetching {url} (attempt {attempt}/{retries+1})")
                async with session.get(url) as resp:
                    if resp.status == 200:
                        data = await resp.read()
```

```
                logger.info(f"Downloaded {url} -> {len(data)} bytes")
                return data
            else:
                logger.warning(f"Non-200 status for {url}: {resp.status}")
                # для деяких статусів можна не пробувати повторно
    except asyncio.TimeoutError:
        logger.warning(f"Timeout fetching {url} (attempt {attempt})")
    except aiohttp.ClientError as e:
        logger.warning(f"Client error fetching {url}: {e} (attempt {attempt})")
    # exponential backoff перед наступною спробою
    backoff = 0.5 * (2 ** (attempt - 1))
    await asyncio.sleep(backoff)
logger.error(f"Failed to download {url} after {retries+1} attempts")
return None


def sanitize_filename(url: str) -> str:
    """Простіше перетворення URL в ім'я файлу (можна розширити)."""
    # беремо останню частину шляху або хеш при відсутності
    p = url.split("?")[0].rstrip("/")
    name = p.split("/")[-1]
    if not name:
        # fallback: заміна неалфавітних символів
        import hashlib
        h = hashlib.sha1(url.encode()).hexdigest()
        name = f"img_{h}.bin"
    # просте очищення
    name = "".join(ch for ch in name if ch.isalnum() or ch in "._-")
    if len(name) > 120:
        name = name[:120]
```

```
    return name


async def save_file(path: Path, data: bytes):
    """Асинхронне збереження файлу на диск."""
    async with aiofiles.open(path, "wb") as f:
        await f.write(data)
    logger.debug(f"Saved {path}")


async def worker(url: str, outdir: Path, session: aiohttp.ClientSession, sem: asyncio.Semaphore, retries: int):
    """Одна задача: скачує URL і зберігає результат."""
    data = await fetch(session, url, sem, retries)
    if data is None:
        return False
    filename = sanitize_filename(url)
    # при конфлікті імен додаємо постфікс
    target = outdir / filename
    counter = 1
    while target.exists():
        stem = target.stem
        suffix = target.suffix
        target = outdir / f"{stem}_{counter}{suffix}"
        counter += 1
    await save_file(target, data)
    return True


async def main_async(urls: List[str], outdir: Path, concurrency: int, retries: int):
```

```
    outdir.mkdir(parents=True, exist_ok=True)
    sem = asyncio.Semaphore(concurrency)
    connector = aiohttp.TCPConnector(limit_per_host=concurrency)
    async with aiohttp.ClientSession(timeout=TIMEOUT, connector=connector) as session:
        tasks = [worker(url.strip(), outdir, session, sem, retries) for url in urls if url.strip()]
        total = len(tasks)
        logger.info(f"Starting download of {total} items with concurrency={concurrency}, retries={retries}")
        results = []
        # запуск обмеженої кількості задач з прогресом
        for chunk_start in range(0, total, 50):  # контролюємо розмір одночасного створення тасків
            chunk = tasks[chunk_start:chunk_start+50]
            r = await asyncio.gather(*chunk)
            results.extend(r)
        ok = sum(1 for r in results if r)
        logger.info(f"Finished: {ok}/{total} succeeded")


def parse_args():
    p = ArgumentParser(description="Асинхронний завантажувач зображень")
    p.add_argument("urls_file", help="Файл зі списком URL (по одному в рядку)")
    p.add_argument("output_dir", help="Папка для збереження")
    p.add_argument("--concurrency", type=int, default=DEFAULT_CONCURRENCY, help="Кількість одночасних з'єднань")
    p.add_argument("--retries", type=int, default=DEFAULT_RETRIES, help="Кількість повторних спроб для кожного URL")
```

```
    p.add_argument("--verbose", action="store_true", help="Детальніше логування")
    return p.parse_args()


def main():
    args = parse_args()
    if args.verbose:
        logger.setLevel(logging.DEBUG)
    urls_path = Path(args.urls_file)
    if not urls_path.exists():
        logger.error("Файл зі списком URL не знайдено: %s", urls_path)
        sys.exit(2)
    urls = urls_path.read_text(encoding="utf-8").splitlines()
    outdir = Path(args.output_dir)
    try:
        asyncio.run(main_async(urls, outdir, args.concurrency, args.retries))
    except KeyboardInterrupt:
        logger.warning("Interrupted by user")
        raise


if __name__ == "__main__":
    main()
```

Пояснення фрагменту коду

1. Ідея та архітектура

Цей скрипт — класичний приклад асинхронного конвеєра завантаження. Основна мета — ефективно (паралельно) завантажити багато URL, не перевантажуючи сервер чи локальну мережу. Для цього використані:

asyncio — ядро асинхронності в Python;

aiohttp — асинхронний HTTP-клієнт;

aiofiles — асинхронне записування у файли (щоб не блокувати event loop при операціях I/O на диск).

Архітектурно: є окрема функція worker для кожного URL, яка викликає fetch (з апаратною логікою retry) та save_file. Контроль паралелізму — через asyncio.Semaphore і aiohttp.TCPConnector(limit_per_host=...). Логіка запуску — в main_async: створюються таски і виконуються пакетно.

2. Логування і конфігурація

На початку ми налаштовуємо просте логування через logging.basicConfig. Це важливо в реальних скриптах — допомагає розуміти, що відбувається під час виконання (успіхи/помилки/таймаути). Параметри, які варто змінювати під час використання:

DEFAULT_CONCURRENCY — обмежує кількість одночасних з'єднань, якщо з'єднань багато — збільшуємо, для збереження етикету до серверів — тримаємо помірне значення;

TIMEOUT — загальний таймаут запитів, 20 секунд — розумний дефолт, але для великих файлів можна збільшити;

3. Функція fetch — retry + semaphor

fetch реалізує:

Semaphore: async with sem — гарантує, що одночасно працюватиме не більше concurrency HTTP-запитів, це важливо, щоб не розірвати з'єднання внаслідок надміру одночасних з'єднань;

Retry: цикл while attempt <= retries з експоненційним backoff (await asyncio.sleep(backoff)), де backoff = 0.5 * (2 ** (attempt - 1)), такий підхід мінімізує навантаження при тимчасових помилках;

Обробка помилок:

asyncio.TimeoutError, aiohttp.ClientError — логуються і триває повторна спроба. Якщо статус HTTP != 200, ми логінуємо warn (для деяких сайтів потрібні 301/302 редиректи — aiohttp зазвичай їх обробляє автоматично).

Повертає bytes або None при невдачі.

4. Іменування файлів

Функція sanitize_filename перетворює URL у більш-менш придатне ім'я файлу:

прибирає параметри (?query=...);

бере останню частину шляху (.../image.jpg);

якщо кінцеве ім'я відсутнє — створює хеш SHA1 як fallback (img_<hash>.bin);

видаляє небажані символи та обрізає довгі імена;

Також в worker є логіка уникнення колізій імен:

якщо target.exists(), додається _1, _2 і т.д.

5. Пакетний запуск тасків

У main_async таски створюються як список tasks = [worker(...)]. Щоб уникнути одночасного створення сотень тисяч тасків у пам'яті (коли список дуже великий), реалізовано простий пакетний підхід з кроком 50:

```
for chunk_start in range(0, total, 50):
    chunk = tasks[chunk_start:chunk_start+50]
    r = await asyncio.gather(*chunk)
    results.extend(r)
```

Це пом’якшує навантаження на event loop та дозволяє бачити проміжні результати.

6. Асинхронне збереження

aiofiles дозволяє писати у файл без блокування. Це корисно, бо при великому потоці завдань синхронне писання в файл могло б зупиняти event loop і знижувати ефективність.

7. Параметри через CLI

argparse дозволяє запускати скрипт з аргументами:

urls_file — файл зі списком URL, по одному в рядку;

- output_dir — куди зберігати;

- concurrency — кількість одночасних з'єднань;

- retries — кількість повторних спроб;

- verbose — увімкнути DEBUG-логування;

Приклад запуску:

python async_image_downloader.py urls.txt ./images --concurrency 8 --retries 3 --verbose

8. Розширення й адаптації

Цей базовий шаблон можна розширити:

додати User-Agent, заголовки, cookies, або аутентифікацію в session.get(...);

підтримку rate limit (наприклад, asyncio.sleep між запитами до одного домену);

збереження метаданих (оригінальний URL, розмір, mime-type) у CSV/SQLite;

додати парсер для збирання URL з HTML-сторінок;

додати тести — unit для sanitize_filename, інтеграційні тести з локальним HTTP-сервером;

додати retry only on idempotent errors (не повторювати POST-запити тощо);

9. Безпека й етика

При масовому скачуванні важливо:

поважати robots.txt сайту (цей приклад його не перевіряє — додайте, якщо потрібно);

не перевантажувати сервери — налаштуйте прийнятну concurrency і backoff;

дотримуватися ліцензій і авторських прав;

10. Тестовий приклад urls.txt

Ось приклад файлу urls.txt:

https://example.com/image1.jpg

https://upload.wikimedia.org/wikipedia/commons/4/47/Example.png

https://example.org/somephoto.png

Запуск: python async_image_downloader.py urls.txt downloaded_images --concurrency 4 --retries 2.

Додаток В – Приклади шахрайських транзакцій

Приклад 1 — «Раптові великі перекази з нового каналу»

Сценарій: Клієнт, який зазвичай отримує зарплатні перекази та робить нечисленні платежі до 1000 грн, раптово отримує кілька великих вхідних переказів (по 50–100 тис. грн) від нових невідомих відправників. Після цього з рахунку починаються швидкі перекази на кілька третіх рахунків, відкритих того ж дня.

Ознаки підозри:

Різка зміна патерну транзакцій (обсяг/частота);

Наявність вхідних переказів від невідомих контрагентів перед швидким виведенням коштів;

Нова/порожня історія рахунку для одного або кількох пов’язаних рахунків;

Дії банку/аналітика:

Тимчасове блокування підозрілих вихідних переказів згідно внутрішніх процедур;

Перевірка KYC-профілів пов’язаних рахунків (документи, дата відкриття, IP/географія доступу);

Зв’язок із клієнтом (через заздалегідь відомі канали) для підтвердження операцій;

Якщо підозра підтверджується — підготовка та подача повідомлення про підозрілу операцію (SAR) відповідно до регуляторних вимог.

Урок: Раптові великі надходження, за якими слідує швидке розосередження коштів, — типова ознака «пральної» активності.

Приклад 2 — «Мікроплатежі для тестування картки»

Сценарій: Картка клієнта зазнає десятків дуже дрібних транзакцій (0.50–2.00 грн або еквівалент) упродовж години на різних інтернет-майданчиках, після чого йдуть кілька більших списань.

Ознаки підозри:

Велике число мікроплатежів від різних торговців за короткий час;

Невідповідність середньому профілю витрат клієнта;

Використання картки в Merchant Category Codes (MCC), які раніше не були характерні;

Дії:

Заблокувати подальші транзакції та повідомити клієнта;

Перевірити pattern різних MCC/географічних місць;

Запропонувати емітувати картку заново або провести заміну/чек-лист безпеки;

Урок:

Мікроплатежі часто використовуються для перевірки валідності картки перед великим шахрайським списанням;

Виявлення таких «skimming/test charges» дозволяє запобігти подальшим втратам;

Приклад 3 — «Соціальна інженерія — зміна реквізитів»

Сценарій: Корпоративний клієнт повідомляє (електронною поштою) про зміну реквізитів постачальника і просить перенаправити всі майбутні платежі на новий рахунок. Запит на зміну супроводжується документом із печаткою — але комбінація адреси IP, часу та телефону контактної особи викликає сумнів.

Ознаки підозри:

Запит на зміну реквізитів, що надходить через електронну пошту, а не через усталені канали/підпис;

Невідповідність контактних даних у документі з наявною інформацією;

Тиск на швидкість виконання (терміновість), щоб обійти стандартні верифікації;

Дії:

Дотримуватись внутрішнього регламенту:

Верифікувати зміну через попередньо зареєстрований телефон або фізичний підпис уповноваженої особи;

Якщо є сумнів — призупинити операцію та ініціювати додаткову верифікацію (наприклад, запитати довідку через офіційний контакт);

Підготувати інцидент-звіт і, за потреби, повідомити правоохоронні органи;

Урок:

Шахраї часто використовують підроблені документи та тиск у часі;

Стандартизовані процедури верифікації захищають від успішних атак соціальної інженерії;

Приклад 4 — «Аномальні операції з картки за кордоном»

Сценарій: Клієнт із типовими українськими витратами раптом демонструє використання картки в іншій країні або з іншого IP-регіону, з транзакціями в двох віддалених містах протягом короткого часу.

Ознаки підозри:

Географічна невідповідність (неможливість фізично бути в двох місцях одночасно);

Невідповідність звичному часу активності клієнта;

Підозріла спроба зняття готівки або high-risk MCC;

Дії:

Тимчасове блокування картки з повідомленням клієнта;

Запит підтвердження через відомий канал (дзвінок на зареєстрований номер);

Якщо клієнт не підтверджує — ініціювати процедуру чарджбек/розслідування;

Урок: Географічні та часові аномалії — простий і ефективний показник підозрілих операцій.

Загальні індикатори шахрайства (чек-ліст для аналітика)

Раптова зміна профілю транзакцій (сума, частота, MCC).

Нові або «порожні» рахунки, пов’язані один з одним.

Мікротестування картки (серія дрібних сум).

Терміновість або спроби тиску для обхідних дій.

Невідповідність між каналами підтвердження (електронна пошта vs. офіційні контакти).

Неодноразові відмови авторизації або повторні спроби.

Географічні аномалії або підміна IP/UA (відмінні від звичайних для клієнта).

Підозрілі ланцюжки переказів (швидке розсіяне виведення коштів).

Стандартні кроки :

- Ідентифікація та первинна оцінка ризику — зафіксувати час;
- ID транзакції;
- клієнта;
- суму;
- канал;

Тимчасове обмеження операції — згідно внутрішніх політик (наприклад, призупинення платежів до завершення верифікації).

Зв’язок з клієнтом — через зафіксовані канали для підтвердження операції.

Збирання доказів — журнали транзакцій, IP/лог-файли, копії документів (зберігати цілісність даних).

Подача внутрішнього інцидент-звіту та зовнішніх повідомлень (за потреби) — регулятор/правоохоронні органи.

Оновлення профілів ризику — блокування/позначка для подальшого моніторингу.

Навчання персоналу / оновлення процедур — якщо виявлено вразливість.

Шаблон повідомлення про підозрілу операцію

Тема: Підозріла транзакція — [ClientID] — [Дата/час]

Короткий опис: (1–2 речення)

Деталі транзакції:

- ID;
- сума;
- валюта;
- канал;
- MCC;
- відправник/одержувач (анонімізовані).

Ознаки підозри: (пункти з чек-ліста)

Дії виконані: блокування/верифікація/повідомлення клієнта.

Рекомендація: подати SAR / звернутися до правоохоронців / потреба в більш глибокому розслідуванні.

Контакт відповідальної особи:

- ім’я;
- відділ;
- телефон;
- e-mail;

Корисні поради для побудови навчального кейсу (без операційних інструкцій)

Використовуйте реальні, але анонімізовані кейси:

- прибрати ідентифікатори;
- зберегти загальний патерн;

Фокусуйтеся на індикаторах і рішеннях, а не на технічних деталях шахрайства.

Робіть вправи «table-top» для персоналу:

Обговорення сценарію;

- Визначення дій і процедур;

Інтегруйте автоматичні правила моніторингу + ручну перевірку для складних випадків.