ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Система прогнозування попиту на основі технологій великих даних»

на здобуття освітнього ступеня бакалавра
зі спеціальності 124 Системний аналіз
(код, найменування спеціальності)
освітньо-професійної програми Системний аналіз
(назва)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ *Олег Прач*

(підпис) *Ім'я, ПРІЗВИЩЕ здобувача*

Виконав: здобувач вищої освіти гр. САД-41
Олег ПРАЧ
Ім'я , ПРІЗВИЩЕ
Керівник: Михайло КУЗЬМІЧ
науковий ступінь, вчене звання Ім'я, ПРІЗВИЩЕ
Рецензент: _____
Ім'я , ПРІЗВИЩЕ

Київ 2026

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут інформаційних технологій

Кафедра Інформаційних систем та технологій

Ступінь вищої освіти Бакалавр

Спеціальність 124 Системний аналіз

Освітньо-професійна програма Системний аналіз

ЗАТВЕРДЖУЮ

Завідувач кафедрою _____

_____Ім'я, ПРІЗВИЩЕ

«____» _____2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Прач Олег Олександрович

(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи Система прогнозування попиту на основі технологій великих даних

керівник кваліфікаційної роботи Михайло КУЗМІЧ, доцент кафедри інформаційних систем та технологій, доктор філософії

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «__» _____ 2026 р. № ____

2. Строк подання кваліфікаційної роботи « 01 » червня 2026р.

3. Вихідні дані до кваліфікаційної роботи: масиви відкритих даних електронної системи ProZorro щодо публічних закупівель, науково-технічна література з питань аналізу Big Data та алгоритмів машинного навчання, технічна документація архітектури СУБД ClickHouse, нормативно-правова база у сфері державних фінансів та методичні матеріали з антикорупційного моніторингу.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Дослідити теоретико-методологічні засади прогнозування попиту та технології обробки великих даних у державному секторі.
2. Спроектувати архітектуру аналітичного сховища на базі СУБД ClickHouse та реалізувати ETL-конвеєр для підготовки даних системи ProZorro.
3. Розробити та навчити моделі машинного навчання для прогнозування цін пального та впровадити алгоритми детекції цінових аномалій.

4. Створити засоби візуалізації результатів у вигляді аналітичних дашбордів та обґрунтувати практичну ефективність впровадження системи.
5. Перелік ілюстративного матеріалу: *презентація*

6. Дата видачі завдання « 20 » лютого 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Збір даних	10.03.2026 – 20.03.2026	Виконано
2	Обробка матеріалу	21.03.2026 – 31.03.2026	Виконано
3	Написання першого розділу роботи	01.04.2026 – 08.04.2026	Виконано
4	Написання другого розділу роботи	09.04.2026 – 17.04.2026	Виконано
5	Написання третього розділу роботи	18.04.2026 – 30.04.2026	Виконано
6	Висновки за результатами роботи	01.05.2026 – 04.05.2026	Виконано
7	Розробка презентації	05.05.2026 – 12.05.2026	Виконано
8	Підготовка доповіді до захисту	13.05.2026 – 26.05.2026	Виконано

Здобувач вищої освіти _____ Олег ПРАЧ
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник
кваліфікаційної роботи _____ Михайло КУЗЬМІЧ
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина бакалаврської роботи: 52 сторінки, 8 таблиць, 18 рисунків, 21 джерела.

Об’єкт дослідження — процес управління та прогнозування попиту в системі публічних закупівель.

Предмет дослідження — моделі, методи та інструменти обробки великих даних для прогнозування обсягів закупівель та виявлення аномалій.

Мета роботи — розробка та обґрунтування моделі системи прогнозування попиту та виявлення аномалій у сфері публічних закупівель на основі технологій великих даних (Big Data).

Завдання роботи: дослідити теоретико-методологічні особливості прогнозування попиту у державному секторі; проаналізувати архітектуру системи зберігання великих даних та алгоритми машинного навчання для пошуку аномалій; здійснити збір, агрегацію та попередній розвідувальний аналіз масиву відкритих даних із системи ProZorro; спроєктувати та реалізувати сховище даних на базі СУБД ClickHouse; розробити програмні алгоритми прогнозування обсягів закупівель та виявлення цінових аномалій; розробити пропозиції щодо візуалізації та практичного впровадження отриманих результатів.

Методи дослідження – методи аналізу та синтезу, методи описової статистики та розвідувального аналізу даних, методи проєктування баз даних, методи машинного навчання (множинна лінійна регресія, випадковий ліс, градієнтний бустинг).

Короткий зміст роботи - у роботі досліджено проблему низької ефективності традиційних методів прогнозування попиту в державному секторі. Зібрано, очищено та збагачено новими ознаками масив транзакційних даних із системи ProZorro. Для забезпечення швидкої обробки великих обсягів інформації спроєктовано та розгорнуто аналітичне сховище на базі стовпчикової СУБД ClickHouse. Проведено порівняльний аналіз трьох ML-моделей, найкращою з яких за точністю визнано алгоритм XGBoost. На базі прогнозів моделі розроблено програмний інструмент виявлення корупційних ризиків, що ґрунтується на аналізі

математичних залишків. Алгоритм дозволив автоматично ідентифікувати аномальні транзакції з ознаками суттєвого завищення цін. Здійснено монетизацію виявлених ризиків та розроблено аналітичні дашборди, що підтверджує високу економічну ефективність впровадження системи.

Ключові слова: ВЕЛИКІ ДАНІ, BIG DATA, ПРОГНОЗУВАННЯ ПОПИТУ, МАШИННЕ НАВЧАННЯ, ПУБЛІЧНІ ЗАКУПІВЛІ, PROZORRO, ВИЯВЛЕННЯ АНОМАЛІЙ, CLICKHOUSE, XGBOOST, КОРУПЦІЙНІ РИЗИКИ.

ЗМІСТ

ВСТУП

Актуальність теми. Ефективне управління державними фінансовими ресурсами та якісне планування бюджету є критичними умовами стабільного розвитку держави, особливо в умовах обмежених ресурсів і в період економічного відновлення України. З моменту впровадження електронної системи публічних закупівель ProZorro та порталу відкритих фінансів Spending.gov.ua з'явилися широкі можливості для аналізу значних обсягів транзакційних даних, як приватного бізнесу, так і державного сектору. Проте величезний обсяг, розрізненість неоднорідність цих даних роблять використання традиційних статистичних підходів прогнозування попиту недостатньо ефективним. Відсутність точних прогнозів призводить до нераціонального використання державних коштів, а обмеженні можливості обробки великих масивів даних можуть створити передумови для виникнення корупційних ризиків та цінових аномалій.

Актуальним напрямом вирішення проблеми є застосування сучасних технологій обробки великих даних (Big Data) та методів машинного навчання. Вони дозволяють не лише підвищити точність прогнозування обсягів закупівель, але автоматизувати виявлення нетипових або потенційно ризикових операцій. Це підтверджує високу актуальність та своєчасність обраної теми дослідження.

Метою кваліфікаційної роботи є розробка та обґрунтування моделі системи прогнозування попиту та виявлення аномалій у сфері публічних закупівель на основі технологій великих даних (Big Data).

Для досягнення поставленої мети в межах дослідження поставлено наступні завдання:

- Дослідити теоретико-методологічні особливості прогнозування попиту у державному секторі;

- Проаналізувати архітектуру системи зберігання великих даних та алгоритми машинного навчання для пошуку аномалій;

- Здійснити збір, агрегацію та попередній розвідувальний аналіз масиву відкритих даних із системи ProZorro;

- Спроектувати та реалізувати сховище даних на базі СУБД ClickHouse для швидкої обробки транзакцій;

- Розробити програмні алгоритми прогнозування обсягів закупівель та виявлення цінових аномалій на основі зібраних даних;

- Розробити пропозиції щодо візуалізації та практичного впровадження отриманих результатів.

Об'єктом дослідження є процес управління та прогнозування попиту в системі публічних закупівель.

Предметом дослідження є моделі, методи та інструменти обробки великих даних для прогнозування обсягів закупівель та виявлення аномалій.

Методи дослідження, які було використано:

- Методи аналізу та синтезу – для дослідження предметної області та літературних джерел;
- Методи описової статистики та аналізу даних – для виявлення сезонних трендів;
- Методи проектування баз даних – для побудови архітектури сховища на базі ClickHouse;
- Методи машинного навчання – для розробки прогнозних моделей та ідентифікації статистичних аномалі

Наукова новизна полягає у розробці масштабованої архітектури аналітичної системи для обробки відкритих даних публічних закупівель із використанням Big Data-технологій та методів статистичного аналізу, орієнтованої на виявлення аномалій і підтримку антикорупційного моніторингу.

Практична значущість полягає у використанні побудованої архітектура бази даних та розробленої ML-моделі як MVP (мінімально життєздатний продукт) для впровадження в аналітичні процеси державних установ з метою оптимізації планування бюджету, а також аудиторськими та громадськими організаціями для

автоматизованого моніторингу ефективності закупівель.

Структура роботи. Кваліфікаційна робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

РОЗДІЛ 1. ОСНОВИ ПРОГНОЗУВАННЯ ПОПИТУ З ВИКОРИСТАННЯМ ВЕЛИКИХ ДАНИХ

1.1 Поняття та специфіка прогнозування попиту у державному секторі

У сучасному світі, та зокрема в Україні, поняття прогнозування попиту є досить обширним. Станом на зараз – це не просто "скільки потрібно купити", а результат впливу зовнішніх та внутрішніх факторів, особливо якщо це стосується державного сектору. На відміну від приватного бізнесу, де попит визначається ринковими механізмами – поведінкою споживачів, їхніми потребами, доходами та готовністю купувати товари чи послуги. Його основною метою є отримання прибутку, тому компанії постійно проводять додаткові аналізи, впроваджують сучасні інструменти аналітики та адаптуються до змін.

Натомість у державному секторі попит має адміністративний і плановий характер та передусім залежить від обсягу та структури державного бюджету, оскільки саме фінансування визначає можливість проведення закупівель незалежно від реальної проблеми. Основну роль відіграють нормативно-правові обмеження та процедури закупівель, які в Україні регламентуються Законом України «Про публічні закупівлі» та проводяться через систему ProZorro. «Метою цього Закону є забезпечення ефективного та прозорого здійснення закупівель, створення конкурентного середовища у сфері публічних закупівель, запобігання проявам корупції у цій сфері, розвиток добросовісної конкуренції»[1]. Оскільки будь-які зміни потребують часу, узгоджень, додаткового фінансування та зазвичай прив'язані до бюджетних періодів – це робить попит, у рамках державного сектору, менш гнучким.

Впровадження електронної системи публічних закупівель ProZorro та порталу Є-data (spending.gov.ua) дозволило перетворити процеси, які раніше були приховані, на великий масив відкритих та структурованих даних за стандартом Open Contracting Data Standard (OCDS). Він був створений для підтримки організацій у підвищенні прозорості укладання контрактів та забезпечення глибшого аналізу даних про укладання контрактів широким колом користувачів[2].

Що дало змогу застосувати математичні методи аналізу для перевірки та відслідковування обігу державних коштів.

Однак традиційні статистичні методи прогнозування у державному секторі зіштовхуються з деякими суттєвими труднощами, що ускладнюють отримання достовірних результатів.

По-перше, це *високий вплив макроекономічного середовища*. Обсяги та вартість державних закупівель прямо залежать від таких основних факторів, як рівень інфляції, індекс споживчих товарів чи зміни в податковій політиці, що робить прогнозування менш передбаченим.

По-друге, *фрагментованість попиту.* В Україні та світі закупівлі здійснюються досить великою кількістю різних установ – від невеликих місцевих організацій (сільські ради, об'єднанні територіальні громади, міські школи, лікарні, дитсадки) до центральних органів влади (Верховна Рада України, Кабінет Міністрів України, Офіс Президента України) – у різних регіонах та з різною періодичністю. У результаті це створює нерівномірні та "зашумлені" часові ряди, особливо на рівні окремих організацій чи регіонів.

По-третє, у даних часто можуть бути наявні *аномалії та ризики*, пов'язані з неефективністю та можливими зловживаннями. У випадку приватного бізнесу різкі зміни цін можуть пояснювати ринковими умовами, то у державному секторі відхилення, наприклад незвично висока ціна за одиницю товару чи нетипові обсяги, можуть свідчити про помилки планування або навіть потенційні корупційні ризики.

Таким чином, прогнозування попиту у державному секторі не може зводитися лише до побудови простих лінійних залежностей. Цей процес вимагає більш комплексної аналітики, яка дозволить не тільки оцінювати майбутні обсяги потреб для раціонального використання бюджету, а й виконувати функцію системи раннього попередження, зокрема для виявлення аномалій.

Зважаючи на великі обсяги даних, що щоденно генеруються на порталі ProZorro, застосування традиційних методів аналізу стає все менш ефективним. Це

зумовлює необхідність використання сучасних підходів, в тому числі технологій обробки великих даних (Big Data) та методів машинного навчання.

1.2 Еволюція та технології обробки Big Data для побудови аналітичних систем

Розвиток систем прогнозування попиту тісно пов'язаний із загальною еволюцією технологій обробки та аналізу даних. Зважаючи на збільшення обсягів інформації, що генерується системою ProZorro та іншими державними реєстрами, важливо якісно обробляти таку велику кількість даних.

Саме тоді на допомогу прийшло використання технологій великих даних (Big Data), які дозволяють просто опрацьовувати масштабні, різнорідні та швидкоплинні потоки інформації. Крім того, такі технології відкривають можливості для набагато глибшого аналізу, виявлення прихованих закономірностей і підвищення точності прогнозів, що особливо важливо в умовах складної та динамічної системи державних закупівель.

Big Data характеризуються принципом "7 V", через спільну першу літеру[3]. В рамках дослідження буде важливим детальніше розглянути кожну "V":

1. **Volume (Обсяг)** – базовий параметр, що відображає фізичний масштаб згенерованих і накопичених даних. Зберігання багаторічної історії транзакцій державного сектору потребує спеціалізованих сховищ, адже система ProZorro формує мільйони записів щодо договорів, учасників, тендерів і лотів.
2. **Velocity (Швидкість)** – характеризує темп створення, обробки та передачі нових даних. Сучасні аналітичні системи орієнтовані на роботу в режимі, наближеному до реального часу, тому для актуальності прогнозів необхідне оперативне завантаження нових даних.
3. **Variety (Різноманітність)** – відображає різні формати та структури даних, включаючи як структуровані, так і неструктуровані. Аналітична система повинна враховувати цю різноманітність під час обробки інформації.

4. **Veracity (Достовірність)** – визначає рівень точності, повноти та надійності даних. Великі масиви інформації часто містять шум, пропуски, помилки або навмисні викривлення, тому забезпечення якості даних є критично важливим для коректного аналізу та виявлення аномалій.
5. **Value (Цінність)** – підкреслює, що сам по собі великий обсяг даних не має значення без можливості отримання корисної інформації. Основна цінність системи полягає у здатності прогнозувати майбутній попит та підтримувати прийняття рішень.
6. **Variability (Мінливість)** – описує нестабільність потоків даних і зміну їх структури з часом. Для ефективного прогнозування важливо, щоб модель могла адаптуватися до таких змін.
7. **Visualization (Візуалізація)** – передбачає подання результатів аналізу великих обсягів даних у зрозумілому вигляді. Використання графіків, теплових карт і таблиць є необхідним для інтерпретації результатів і прийняття обґрунтованих управлінських рішень.

Однією із ключових технічних проблем при створенні аналітичних систем є вибір архітектури системи бази даних. У сучасній інженерії даних системи зберігання інформації зазвичай поділяють на дві ключові категорії: *транзакційні* (OLTP) та *аналітичні* (OLAP). *Транзакційні* системи орієнтовані на кінцевих користувачів і забезпечують обробку великої кількості коротких запитів, вони оптимізовані для швидкого внесення змін, високої надійності та цілісності даних. Натомість *аналітичні* системи призначені для задач глибокого аналізу та прогнозування. Мають меншу кількість запитів, але кожен із них є набагато складнішим і потребує значної обробки. Їх різницю чудово описано у роботі Мартіна Клеппманна "Designing Data-Intensive Applications", головні відмінності вказано у таблиці 1.1.

Таблиця 1.1

Головні відмінності OLTP та OLAP

Властивість	**Системи обробки транзакцій (OLTP)**	**Аналітичні системи (OLAP)**
Основний алгоритм читання	Невелика кількість записів на один запит, що витягуються за ключем	Звести дані за великою кількістю записів
Основний алгоритм запису	Запис з довільним доступом і низькою затримкою на основі введених користувачем даних	Масовий імпорт (ETL) або потік подій
В основному використовується	Кінцевий користувач/клієнт через веб-додаток	Внутрішній аналітик для підтримки прийняття рішень
Що відображають ці дані	Актуальний стан даних (на даний момент)	Хронологія подій, що відбувалися з плином часу
Розмір набору даних	Гігабайти в терабайти	З терабайтів у петабайти

Як випливає з наведеної характеристики, ключовою проблемою при створенні аналітичної системи є забезпечення швидкого сканування та обробки великих обсягів історичних даних. Традиційні реляційні бази даних використовують *рядкову* модель зберігання, за якої вся інформація про запис зберігається разом у вигляді одного рядка.

У результаті під час виконання аналітичних запитів — наприклад, при розрахунку середньої вартості закупівель для певної категорії товарів за

визначений період — система змушена зчитувати з диска повні записи з усіма їхніми полями, навіть якщо більшість із них не використовується в обчисленнях. Такий підхід є неефективним для аналітики, оскільки створює зайве навантаження на систему введення-виведення.

У контексті обробки великих масивів даних, зокрема тих, що формуються в системі ProZorro, це призводить до суттєвого зниження продуктивності. Надмірне навантаження на дискову підсистему стає вузьким місцем, що ускладнює виконання складних запитів і вимагає переходу до більш ефективних підходів зберігання та обробки даних.

Суть *стовпцевого* підходу до зберігання даних полягає в тому, що інформація організовується не за рядками, а за окремими стовпцями. Тобто всі значення певного атрибута зберігаються разом, незалежно від того, до якого запису вони належать. Завдяки цьому під час виконання запиту система звертається лише до тих стовпців, які необхідні для обчислень, і не витрачає ресурси на зчитування зайвих даних. Такий підхід дозволяє значно зменшити обсяг операцій введення-виведення та підвищити ефективність обробки інформації, особливо при роботі з великими наборами даних[4].

Окрім цього, зберігання однотипних даних у межах одного стовпця дає змогу використовувати значно ефективніші методи стиснення порівняно з рядковими СУБД. Це сприяє не лише зменшенню займаного дискового простору, але й суттєво підвищує швидкість виконання агрегаційних операцій, зокрема обчислення середніх цін чи сум закупівель за тривалі часові періоди.

З огляду на значні обсяги, високу швидкість оновлення та специфіку структури відкритих державних даних України, майбутня СУБД повинна забезпечити необхідний рівень продуктивності для обчислення аналітичних показників у режимі, близькому до реального часу. Одним із показових прикладів сучасних стовпчикових систем управління базами даних, архітектура яких спеціально розроблена для надшвидкого виконання OLAP-запитів, є ClickHouse. У таблиці 1.2 наведено порівняльну характеристику інших популярних СУБД та ClickHouse.

Таблиця 1.2

Порівняльна характеристика популярних СУБД

Характеристика	**PostgreSQL (OLTP)**	**Apache Hive (Hadoop)**	**DuckDB**	**ClickHouse**
Модель зберігання	Рядкова	Стовпчикова	Стовпчикова	Стовпчикова
Швидкість запитів	Низька	Середня	Дуже висока	Надзвичайно висока
Масштабованість	Обмежена	Висока	Відсутня	Дуже висока
Стиснення даних	Мінімальне	Високе	Високе	Дуже високе
Складність архітектури	Низька	Дуже висока	Дуже низька	Середня
Обробка в реальному часі	Так	Ні	Так	Так

На основі аналізу даних таблиці саме *ClickHouse* найкраще підходить для досягнення цілей дослідження, тому її було обрано як основний інфраструктурний елемент для побудови моделі прогнозування попиту в межах даного дослідження, оскільки воно найкраще відповідає вимогам до швидкості, масштабованості та ефективності обробки великих масивів даних.

1.3 Методи машинного навчання для аналізу часових рядів та пошуку аномалій

Наявність потужного та швидкодіючого сховища даних є важливим етапом, але сама по собі не гарантує створення повноцінної системи прогнозування. Класичні статистичні підходи до аналізу часових рядів, наприклад ковзнє середнє чи базові

моделі *ARIMA*, можуть демонструвати більшу ефективність в умовах комерційних ринків. Водночас у державному секторі вони часто працюють набагато гірше.

Це пояснюється складною природою даних, зокрема тих, що формуються в системі ProZorro: вони характеризуються високою нелінійністю, наявністю прихованих взаємозв’язків між численними факторами (такими як регіон, категорія товару за класифікатором CPV, макроекономічні показники), а також суттєвим впливом людського фактора.

Зважаючи на це, більш доцільно використовувати методи машинного навчання (ML), які здатні автоматично розпізнавати складні закономірності у великих обсягах даних. Зокрема, на відміну від традиційних методів статистичної екстраполяції, інтелектуальні алгоритми дають можливість створювати предиктивні моделі, які враховують не лише історичні зміни цін, а й складні контекстні взаємозв’язки між особливостями замовника та ринковою динамікою.

Методи машинного навчання чудово підходять у таких ситуаціях:

1. Коли для вирішення проблеми традиційне рішення потребує складного налаштування або великої кількості жорстко заданих правил. Один алгоритм машинного навчання часто може спростити код і працювати краще, ніж традиційний підхід.
2. Для складних проблем, у яких класичні підходи не дають задовільного результату, оскільки методи машинного навчання здатні знаходити оптимальне рішення.
3. В умовах мінливого середовища, адже можуть адаптуватися до нових даних і змін інформації.
4. Для отримання інформації про складні проблема та великі обсяги даних, що важко досягти за допомогою традиційних методів[5].

У рамках цього дослідження застосування ML виконує дві взаємопов'язані функції: *прогнозування* майбутнього попиту та *виявлення* аномалій, що можуть свідчити про потенційні корупційні ризики або неефективність роботи.

1. *Прогнозування попиту*

Прогнозування обсягів закупівель фактично зводиться до застосування регресійного аналізу часових рядів. Сучасним стандартом для роботи з табличними даними, зокрема тими, що отримуються з баз даних, є використання ансамблевих методів на основі дерев рішень, передусім алгоритмів градієнтного бустингу, такі як XGBoost, LightGBM та CatBoost. Вони здатні ефективно обробляти категоріальні ознаки (наприклад, ID замовника або код товару), а також враховувати вплив зовнішніх факторів, такі як індекс споживчих цін чи сезонні коливання бюджету. Завдяки цьому вони дозволяють будувати більш точні та надійні прогнози майбутніх обсягів закупівель.

2. Виявлення аномалій

Якщо прогнозування дозволяє визначити очікуванні, правильні параметри закупівель, то аналіз аномалій спрямований на виявлення ситуацій, коли реальні дані суттєво відрізняються від норми. Виявлення аномалій корисне в широкому спектрі застосувань, таких як виявлення шахрайства, виявлення дефектних продуктів у виробництві або видалення викидів з набору даних перед навчанням іншої моделі (що може значно покращити продуктивність результуючої моделі)[5]. В аналітиці державних закупівель аномалією вважається подія (окрема транзакція або тендер), яка статистично відрізняється від загального масиву даних. З практичної точки зору такі відхилення виступають початковими сигналами, які можуть свідчити про корупційні ризики, змову учасників або неефективне планування.

Методи *виявлення аномалій* поділяються на кілька основних категорій. З огляду на специфіку даних системи ProZorro, де переважає частка "нормальних" операцій, найбільш доцільним є використання алгоритмів навчання без учителя. Як правило, цей етап включає або автоматичне вилучення, або перетворення вибіркових ознак з нерозмічених масивів даних, кластеризацію даних, або ж ініціалізацію параметрів для наступних процедур навчання[6].

Серед таких методів можна виділити кілька ключових. Наприклад, алгоритм *Isolation Forest* базується на ансамблі дерева рішень і ґрунтується на припущенні, що аномальні спостереження є рідкісними та відрізняються від

інших[7], тому їх легше ізолювати на ранніх етапах побудови дерева. Це робить його надзвичайно швидким і ефективним для великих наборів даних. Графічне відображення принципу роботи алгоритму, де аномальні точки відсікаються за мінімальну кількість розгалужень, наведено на рисунку 1.1.

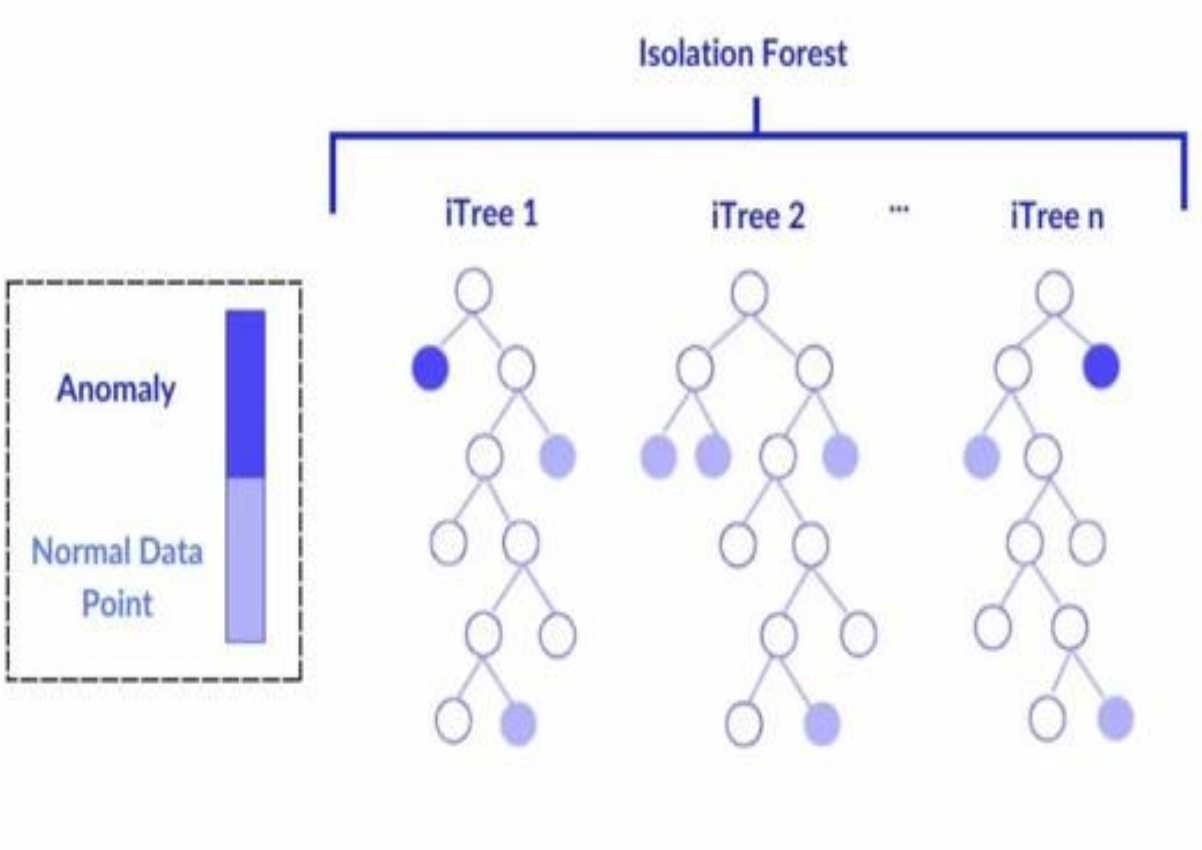


Рис. 1.1 – Метод Isolation Forest

Метод *DBSCAN* (Density-Based Spatial Clustering of Applications with Noise), виконує кластеризацію на основі щільності [8]. Він об'єднує закупівлі з подібними характеристиками в групи, тоді як інші об'єкти будуть розглядатися як аномалії. Основною перевагою цього підходу є відсутність потреби у попередньому заданні кількості кластерів, а також здатність ефективно працювати з даними складної нелінійної структури. У сфері публічних закупівель це дає змогу автоматично виділяти групи типових контрактів, наприклад за схожими обсягами постачання чи логістичними характеристиками, водночас коректно відокремлюючи нетипові цінові пропозиції, які розглядаються як статистичний «шум». Візуалізацію процесу кластеризації за допомогою алгоритму *DBSCAN*, що демонструє відокремлення

основних груп даних від поодиноких аномальних викидів, наведено на рисунку 1.2

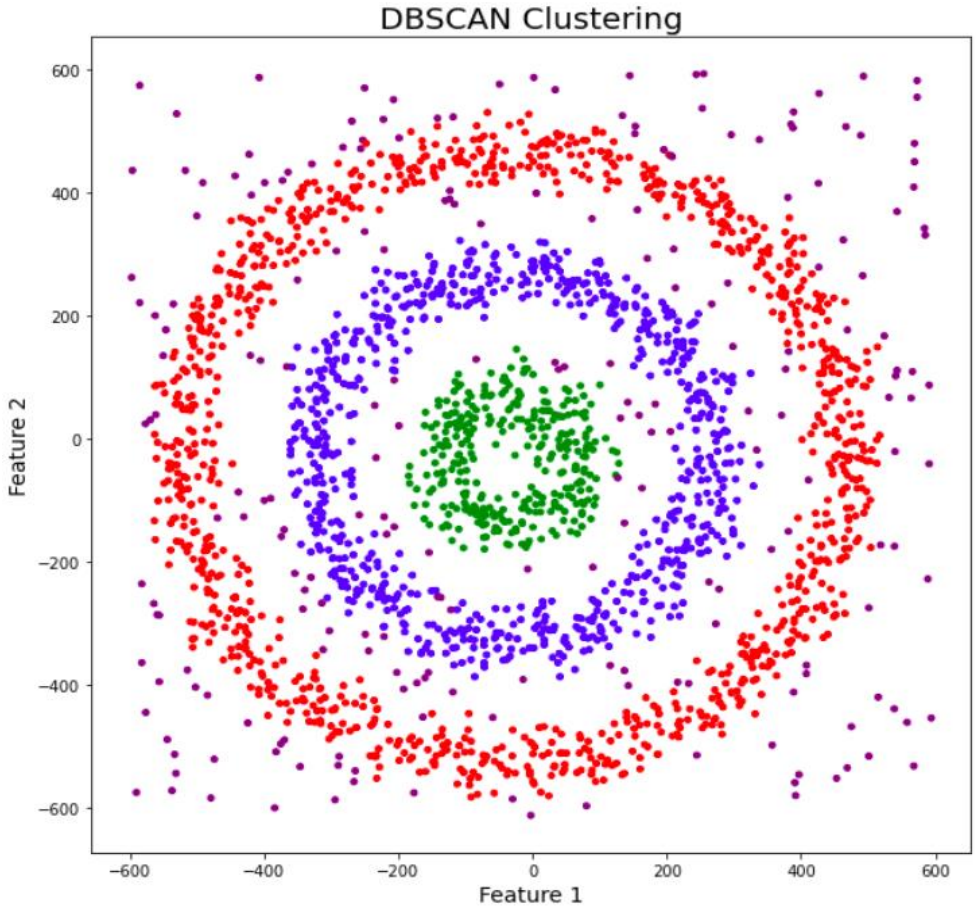


Рис. 1.2 – Приклад роботи DBSCAN

Ще один підхід - *Local Outlier Factor* (LOF) оцінює локальну щільність даних відносно найближчих сусідів, що дозволяє знаходити контекстуальні відхилення. Наприклад, ціна в межах міста може бути типовою, але водночас занадто великою для невеликого населеного пункту. Саме ця здатність алгоритму виявляти приховані маніпуляції на мікрорівні робить його особливо ефективним інструментом для моніторингу неоднорідних масивів державних закупівель, де корупційні націнки нерідко маскуються під специфічні умови окремих лотів. Графічну інтерпретацію роботи цього методу, де аномалії (викиди) ідентифікуються на основі суттєвої різниці між їхньою локальною щільністю та щільністю сусідніх об'єктів, наведено на рисунку 1.3

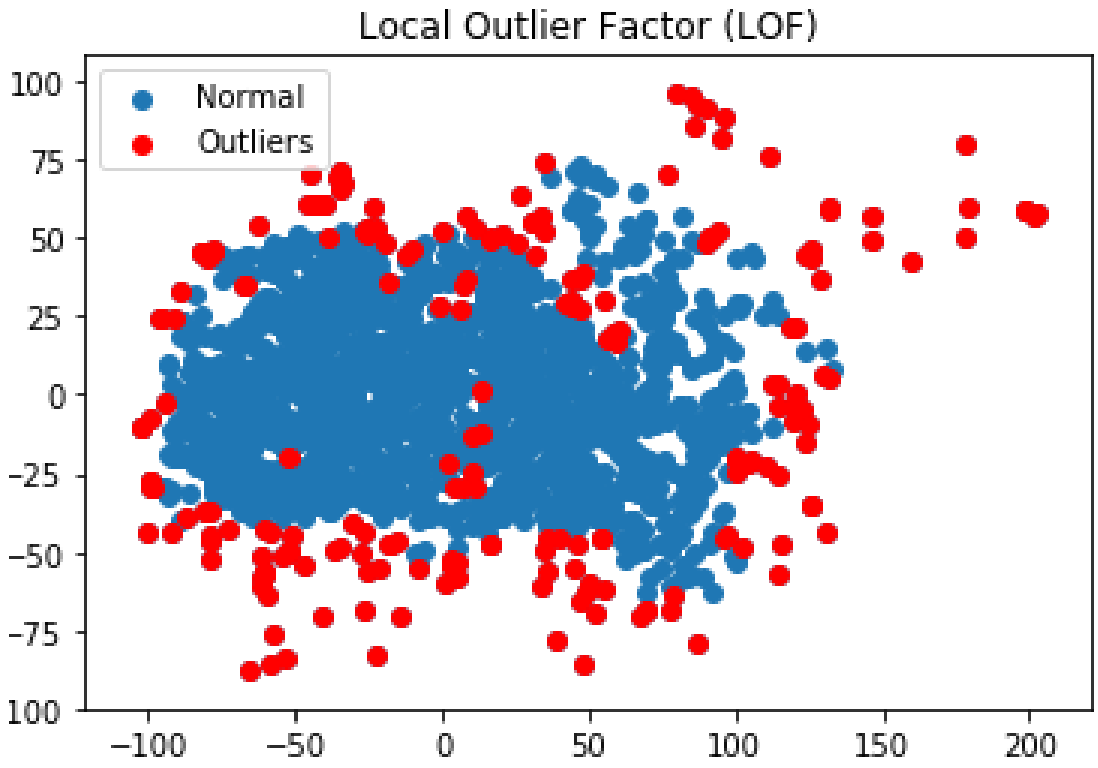


Рис. 1.3 – Метод Local Outlier Factor

Поєднання сучасних технологій зберігання та аналізу даних формує основу ефективної аналітичної системи. Зокрема, інтеграція стовпчикової СУБД ClickHouse, яка забезпечує швидку агрегацію історичних даних, із методами машинного навчання (такими як градієнтний бустинг для прогнозування і Isolation Forest для виявлення відхилень) дозволяє створити комплексне рішення. Така система не лише прогнозує майбутній попит, але й автоматично ідентифікує операції з підвищеним ризиком неефективного використання бюджетних ресурсів.

РОЗДІЛ 2. АНАЛІЗ ВІДКРИТИХ ДАНИХ ПУБЛІЧНИХ ЗАКУПІВЕЛЬ ТА ПРОЕКТУВАННЯ АРХІТЕКТУРИ АНАЛІТИЧНОЇ СИСТЕМИ

2.1 Аналіз джерел відкритих даних та методика формування датасету

Відповідно до базових принципів побудови інформаційних систем, передача даних від джерела до аналітичного сховища здійснюється за допомогою архітектурного підходу *ETL* (Extract, Transform, Load). Він складається з вилучення операційних даних з вихідної програми, їх перетворення, завантаження та індексування, забезпечення якості та публікації [9].

Як зазначає у своїх працях Ральф Кімбалл, процес *ETL* є ключовою складовою корпоративного сховища даних, адже саме на цьому етапі розрізнені, «сирі» та часто неточні дані перетворюються на впорядкований і якісний ресурс, придатний для подальшого аналізу. Принцип роботи коротко пояснено на рисунку 2.1.

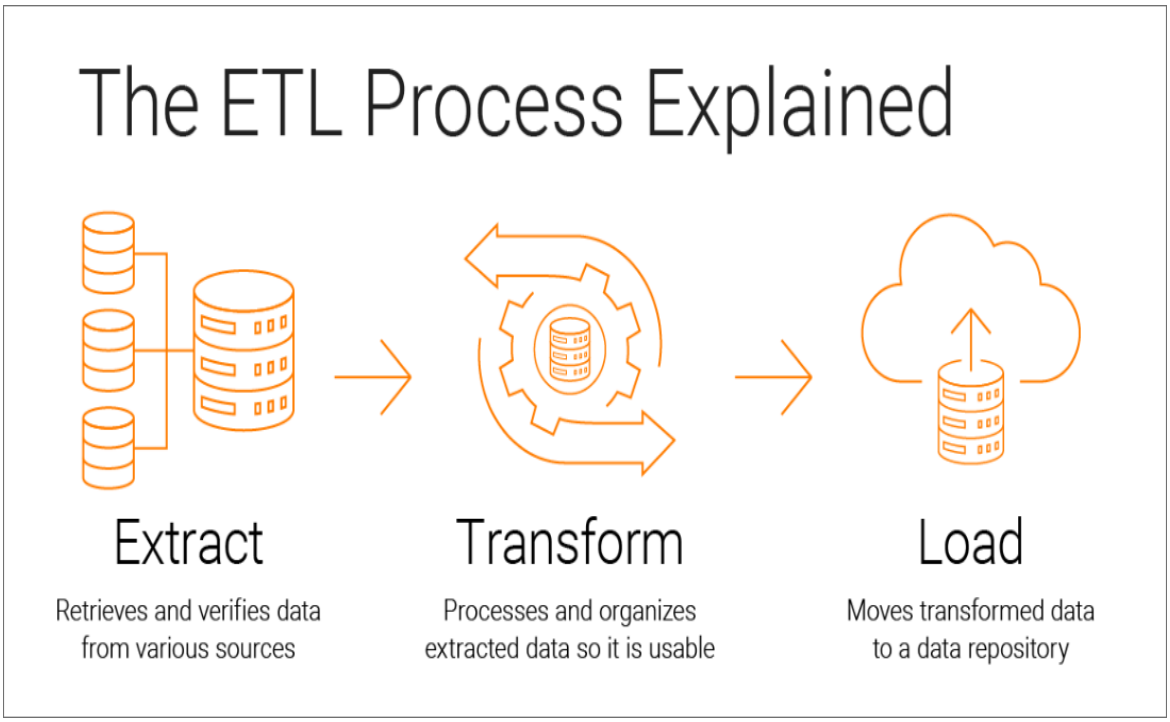


Рис. 2.1 – Пояснення процесу ETL

Етап витягування (*Extract*) передбачає отримання даних із зовнішніх джерел; етап трансформації (*Transform*) охоплює їх очищення, нормалізацію та

об’єднання; етап завантаження (*Load*) відповідає за збереження обробленої інформації у цільовому сховищі даних.

Побудова будь-якої аналітичної системи або прогнозної моделі починається з етапу збору та початкового аналізу даних (*Extract*). У сфері публічних закупівель України основним та найбільш повним джерелом інформації є електронна система ProZorro. Вона функціонує на базі стандарту відкритих даних Open Contracting Data Standard (OCDS), що забезпечує структурованість даних та можливість їх автоматизованого оброблення.

Оскільки повний обсяг даних ProZorro становить терабайти інформації, то для розробки мінімально життєздатного продукту (MVP) та перевірки гіпотез системи було прийнято рішення звузити генеральну сукупність.

Критерії вибірки для дослідження:

- *Часовий проміжок:* 01.01.2025-31.12.2025 рік.

- *Регіон*: Київська область.

- *Предметна область:* Було обрано категорію 09130000-9 — Нафта і дистиляти (пальне). Вибір обумовлений великою частотою закупівель у цій категорії, значною мінливістю цін, залежністю від макроекономічних показників (курси валют, ціни на нафту), а також наявністю випадків необґрунтованого завищення вартості.

- *Статус процедури:* Для аналізу будуть використовуватися лише тендери зі статусом «Завершена» (complete).

Архітектура отримання та обробки даних

Для ефективної обробки даних і вирішення поставлених задач було розроблено комплексну архітектуру аналітичної системи (рис. 2.2). Запропоноване рішення охоплює повний цикл роботи з даними: від їх первинного отримання з державних реєстрів до формування математичних прогнозів і візуалізації фінансових втрат.

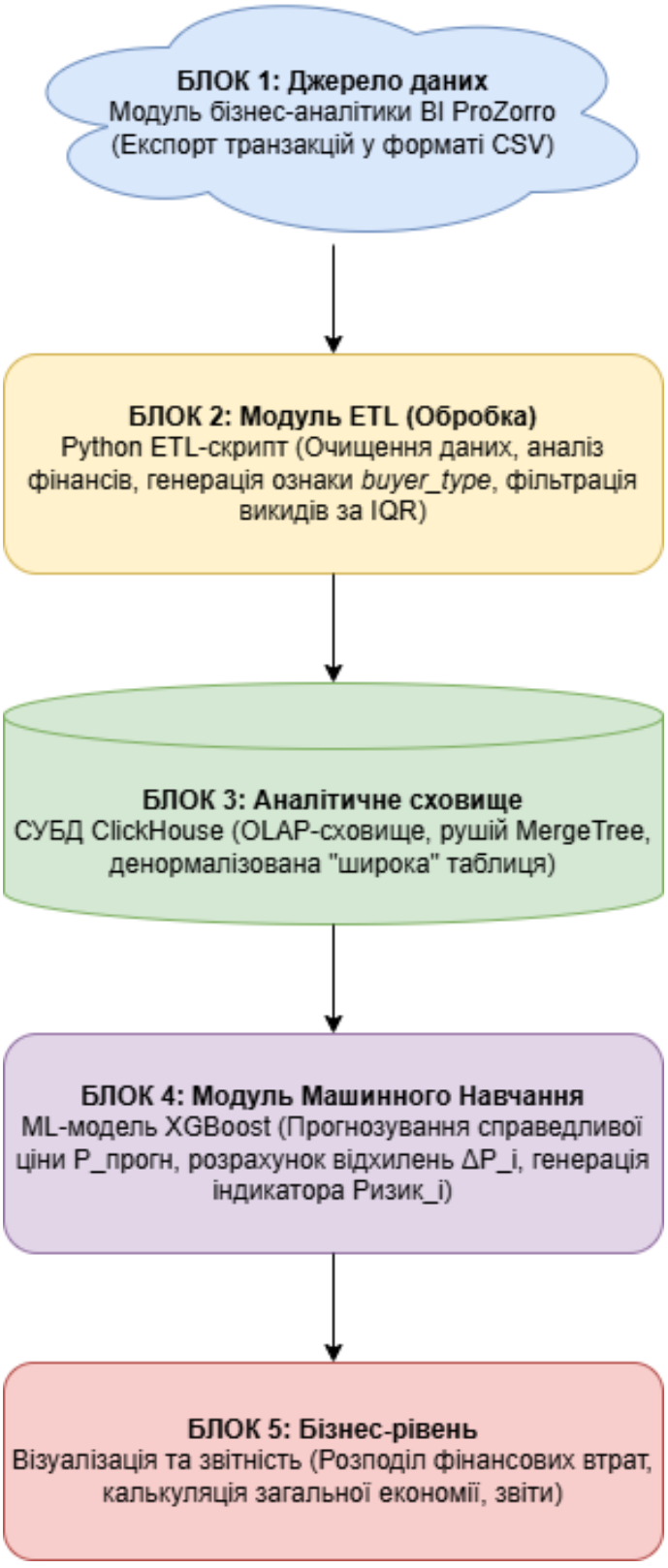


Рис. 2.2 – Загальна архітектура розробленої аналітичної системи

Як видно з наведеної схеми, першим рівнем системи є джерело даних. Зважаючи на те, що відкритий програмний інтерфейс (API) системи має суворі

обмеження пропускної здатності, для первинного наповнення аналітичного сховища було обрано архітектурний підхід пакетного завантаження. Вилучення історичних даних здійснено за допомогою офіційних статистичних масивів у форматі CSV, згенерованих через публічний модуль бізнес-аналітики BI ProZorro. Важливо відзначити, що модуль BI ProZorro є офіційною аналітичною підсистемою, яка безпосередньо синхронізується з центральною базою даних електронної системи закупівель. Відповідно, експортовані масиви даних є повністю актуальними, проходять державну перевірку та відповідають інформації, опублікованій на основному порталі ProZorro. Це гарантує збереження цілісності та достовірності джерела даних для дослідження.

Також слід зазначити, що обрана архітектура розроблена з урахуванням подальшого масштабування. У перспективі передбачено можливість інтеграції модуля інкрементального оновлення через REST API ProZorro, що дозволить автоматизувати щоденне підвантаження актуальних даних у реальному часі. Це забезпечить своєчасне оновлення прогнозованих моделей для майбутніх періодів 2026 року без необхідності повторного завантаження повних наборів даних.

Для автоматизації процесу трансформації даних (етап *Tranform*) розроблено програмний модуль мовою програмування Python з використанням бібліотеки *pandas*[10]. Аналіз реальних даних виявив ряд суттєвих проблем із їх якістю, що потребували комплексного підходу до обробки:

1. *Нормалізація фінансових форматів:* інформація про суми закупівель надходить у вигляді неструктурованого тексту з нерозривними пробілами та комами як десятковими роздільниками. Для їх обробки застосовано регулярні вирази, що дозволяють очистити текст, перетворити його у числовий формат *Float64* та обчислити фактичний обсяг закупівель у літрах.
2. *Формування ознак*: з огляду на суттєві відмінності у поведінці різних учасників ринку, у модулі реалізовано створення нової категоріальної ознаки *buyer_type*. Використовуючи бібліотеку *numpy*, система аналізує назву

замовника та автоматично відносить його до категорій «Комунальний», «Державний» або «Держ. компанія».

3. *Фільтрація аномалій*: для усунення аномальних значень застосовано метод на основі міжквартильного розмаху (IQR). Цей метод, вперше запропонований математиком Джоном Тьюкі, належить до методів робастної статистики. На відміну від класичних підходів, метод IQR не спирається на середнє арифметичне, яке саме спотворюється під впливом аномалій. Використання квартилів дозволяє ефективно та автоматизовано ідентифікувати статистичні викиди навіть у несиметричних розподілах, якими найчастіше є масиви цінових пропозицій на ринку[11].

Наявність зазначених проблем ускладнює безпосереднє завантаження «сирих» даних у базу ClickHouse для подальшого використання у ML-моделях. Тому й виникає критична потреба у проміжному етапі обробки даних (*Transform*).

Фрагмент розробленого програмного коду, що демонструє етапи зчитування даних та їх обробки, наведено на рисунку 2.3.

```
print("Початок завантаження")
file_name = 'b79ca308-9067-4f6e-8e73-409e2b77bc77.xlsx'
try:
    df_raw = pd.read_excel(file_name)
    print(f"Успішно зчитано {len(df_raw)} рядків з файлу.")
except FileNotFoundError:
    print(f"Помилка: Файл {file_name} не знайдено!")
    exit()
# Видалення нерозривних пробілів (\xa0) та звичайних пробілів, заміна коми на крапку
def clean_money_format(series):
    cleaned = series.astype(str).str.replace(r'[\s\xa0]', '', regex=True).str.replace(',', '.')
    return pd.to_numeric(cleaned, errors='coerce')
# Етап TRANSFORM
df_mapped = pd.DataFrame()
# Зіставлення колонок з файлу зі схемою бази даних
df_mapped['tender_id'] = df_raw['Ідентифікатор лота']
df_mapped['tender_date'] = pd.to_datetime(df_raw['Дата оголошення лота']).dt.date
df_mapped['cpv_code'] = df_raw['Класифікація CPV'].astype(str).str.split(' ').str[0]
df_mapped['item_name'] = df_raw['Лот']
df_mapped['buyer_region'] = df_raw['Організатор']
```

Рис. 2.3 – Фрагмент коду Python для обробки масивів

Повний код програмного модуля мовою Python, який реалізує весь цикл автоматизованого експорту, очищення, конструювання ознак та пакетного завантаження даних до СУБД ClickHouse, наведено у Додатку А.

З наведеного фрагмента коду видно, що розроблений модуль використовує механізми відмовостійкості для обробки великих масивів даних. Зчитана інформація перетворюється у структурований табличний формат за допомогою бібліотеки *pandas* для подальшого виконання.

Розроблений Python-скрипт виконує глибоке очищення: застосовує регулярні вирази для конвертації фінансових рядків у математичний тип, відокремлює ідентифікатори CPV від текстових описів, автоматично класифікує замовників та відсікає математичні викиди за допомогою робастного методу розрахунку міжквартильного розмаху (IQR). Тільки після цих комплексних операцій формується нормалізований чистий датасет, придатний для пакетного завантаження у високопродуктивне стовпчикове сховище

2.2. Проектування архітектури сховища даних на базі СУБД ClickHouse

Після завершення етапів отримання та очищення даних (*Extract* та *Transform*) підготовлений та нормалізований масив інформації підлягає завантаженню до цільового сховища (*Load*). Враховуючи обрану архітектурну модель OLAP, основою аналітичної інфраструктури розроблюваної системи є стовпчикова СУБД ClickHouse.

У традиційних реляційних базах даних (OLTP) структура зазвичай будується відповідно до нормальних форм, що дозволяє уникати дублювання інформації. Однак в аналітичних системах операції з’єднання таблиць (JOIN) є одними з найбільш ресурсоємних і значно уповільнюють виконання запитів при роботі з великими обсягами даних.

Саме тому при проєктуванні схеми для ClickHouse було використано підхід *денормалізації*. Він передбачає об’єднання всіх ключових вимірів (таких як дата,

характеристики замовників, параметри товарів) і фактів (ціни, обсяги) в одну «широку» таблицю. Така структура дозволяє значно пришвидшити агрегацію даних, що є критично важливим для розрахунку статистичних показників і підготовки ознак для моделей машинного навчання. На рисунку 2.4 наведено приклад об’єднаних даних в один довгий рядок.

tender_id	date	cpv_code	item_name	buyer_name	buyer_region	quantity	unit_price	total_value	status
UA-2024-01	2024-01-15	09134200-9	Дизельне пальне	ДП "Лікарня №1"	Київська	1000	51.50	51500.00	complete
UA-2024-02	2024-01-16	09132000-3	Бензин А-95	КП "Автодор"	Одеська	500	54.00	27000.00	complete

Рис. 2.4 – Приклад об’єднання даних

ClickHouse використовує концепцію так званих рушіїв таблиць, які визначають, як саме дані зберігаються на диску та обробляються. Для поставленої задачі найбільш доцільним є використання базового рушія сімейства **MergeTree**[12], який оптимізований для пакетного завантаження великих обсягів даних і швидкого виконання аналітичних запитів.

Ключовою характеристикою рушія **MergeTree** є ключ сортування (*ORDER BY*), який фактично виконує функцію первинного індексу. Саме за цим ключем дані сортуються та фізично зберігаються на диску. Оскільки система спрямована на аналіз динаміки попиту в часі та пошук цінових аномалій за категоріями, доцільно використовувати складений ключ, що включає дату оголошення тендеру та код класифікатора *cpv_code*. Це дозволяє значно підвищити ефективність виконання запитів, оскільки замість повного сканування таблиці система може швидко звертатися лише до необхідного діапазону даних, наприклад, усі закупівлі бензину за лютий 2025 року.

Така ефективність вибірки досягається завдяки архітектурі самого індексу, що регулюється параметром *index_granularity*. На відміну від традиційних баз даних, де індекс створюється окремо для кожного рядка, ClickHouse використовує розріджений індекс. Він формує мітки тільки для окремих блоків даних – за замовчуванням приблизно кожні *8192* рядки. Такий підхід зменшує

обсяг індексу і дозволяє повністю розміщувати його в оперативній пам’яті, забезпечуючи дуже швидкий пошук потрібних даних навіть при роботі з мільярдами записів.

Для забезпечення мінімального обсягу зберігання та максимальної швидкості обробки, при проектуванні фізичною схеми особливу увагу приділено вибору типів даних. Для текстових полів з обмеженою кількістю унікальних значень (наприклад регіон замовника чи статус тендеру) використано тип *LowCardinality(String)*. Це дає СУБД зберігати дані у вигляді словника з числовими ключами, що зменшує обсяг таблиці на диску та пришвидшує фільтрацію. На рисунку 2.5 представлено DDL-запит (Data Definition Language, він використовується у ClickHouse) для створення основної таблиці сховища, що враховує описані рішеня.

```
CREATE TABLE default.prozorro_tenders
(
    tender_id String,
    tender_date Date,
    cpv_code String,
    item_name String,
    buyer_region LowCardinality(String),
    buyer_type LowCardinality(String),
    expected_value Float64,
    item_quantity Float64,
    unit_price Float64,
    status LowCardinality(String)
)
ENGINE = MergeTree()
ORDER BY (tender_date, cpv_code)
SETTINGS index_granularity = 8192;
```

Рис. 2.5 – Запит створення таблиці у ClickHouse

Результат фізичного створення структури таблиці в середовищі DBeaver представлено на рис. 2.6. На ньому видно відповідність типів даних проектним рішенням, зокрема застосування оптимізованого зберігання для категоріальних ознак.

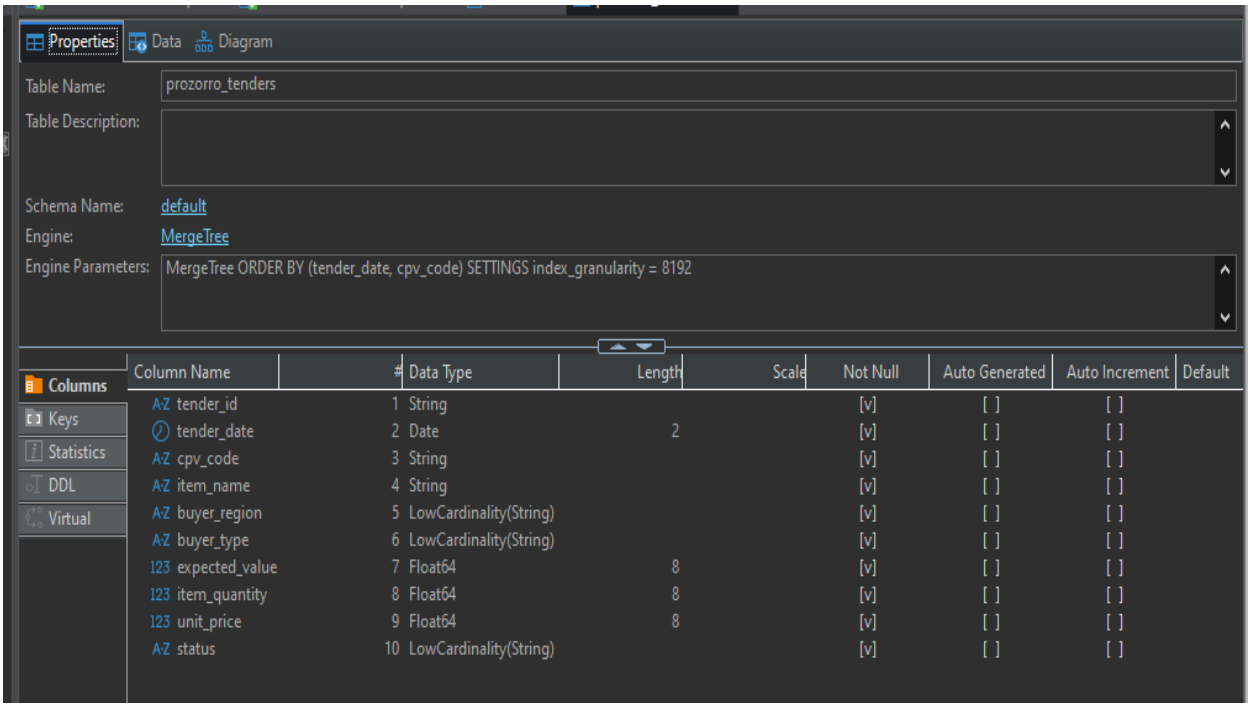


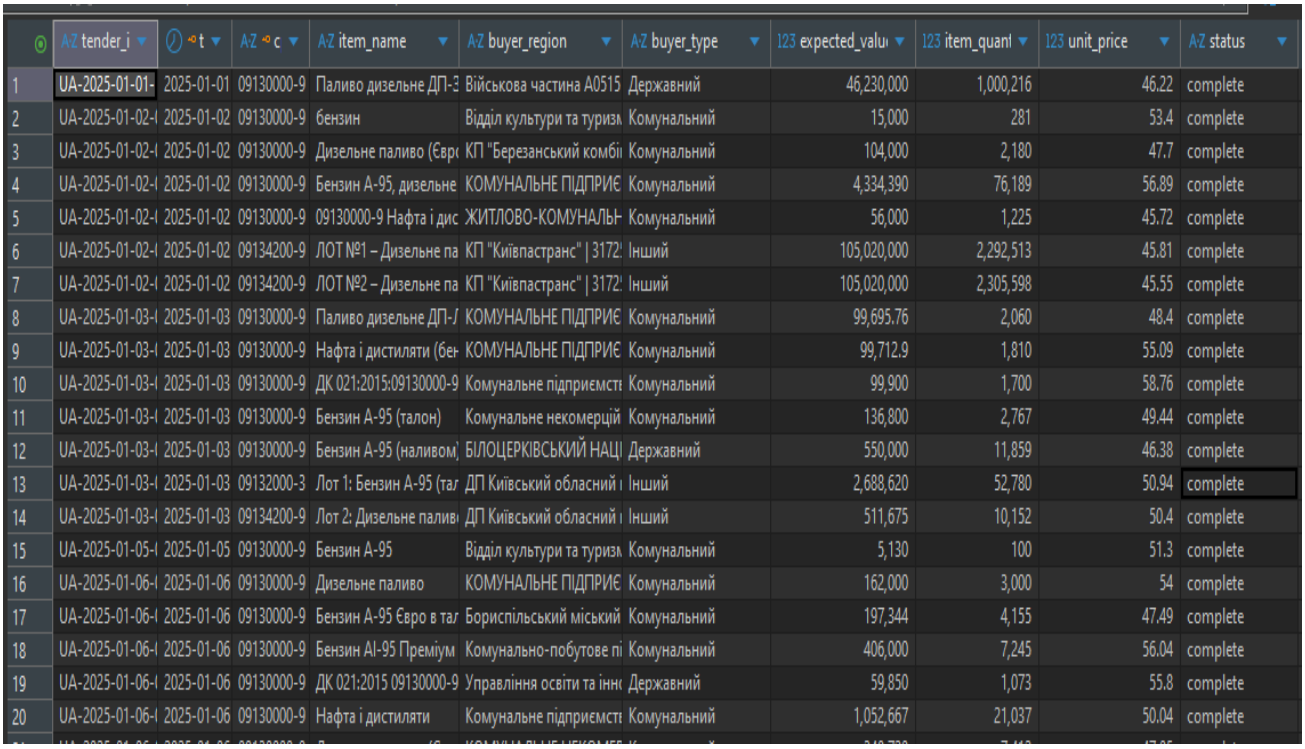
Рис. 2.6 –Візуальне представлення структури таблиці у DBeaver

Для забезпечення стабільного роботи та ізоляції ресурсів аналітичного сховища, СУБД ClickHouse було розгорнуто у віртуалізованому середовищі на базі операційної системи *Ubuntu Server*. Вибір цієї платформи зумовлений її високою продуктивністю при роботі з системними ресурсами та нативною підтримкою з боку розробників ClickHouse. Процес встановлення включав підключення офіційних репозиторіїв та налаштування мережевих інтерфейсів для забезпечення віддаленого доступу до сервера.

Для адміністрування бази даних, візуального проєктування структури та виконання аналітичних SQL-запитів використовується середовище DBeaver. Воно підтримує роботу з ClickHouse через спеціальний драйвер, що дозволяє не лише зручно переглядати вміст «широких» таблиць, а й здійснювати первинну візуалізацію даних та експорт результатів для подальшого аналізу. Використання такого інструменту дозволило суттєво пришвидшити етап розробки архітектури та перевірку коректності завантажених даних із системи ProZorro.

На рис. 2.7 продемонстровано фрагмент фінального датасету після успішного завершення ETL-процесу. Візуальна перевірка підтверджує цілісність даних:

відсутність пропусків, коректність обчислених цін та успішну роботу алгоритму автоматичної класифікації замовників.



	tender_i	t	c	item_name	buyer_region	buyer_type	expected_valu	item_quant	unit_price	status
1	UA-2025-01-01-	2025-01-01	09130000-9	Паливо дизельне ДП-З	Військова частина А0515	Державний	46,230,000	1,000,216	46.22	complete
2	UA-2025-01-02-(	2025-01-02	09130000-9	бензин	Відділ культури та туризм	Комунальний	15,000	281	53.4	complete
3	UA-2025-01-02-(	2025-01-02	09130000-9	Дизельне паливо (Євро	КП "Березанський комбі	Комунальний	104,000	2,180	47.7	complete
4	UA-2025-01-02-(	2025-01-02	09130000-9	Бензин А-95, дизельне	КОМУНАЛЬНЕ ПІДПРИЄ	Комунальний	4,334,390	76,189	56.89	complete
5	UA-2025-01-02-(	2025-01-02	09130000-9	09130000-9 Нафта і дис	ЖИТЛОВО-КОМУНАЛЬН	Комунальний	56,000	1,225	45.72	complete
6	UA-2025-01-02-(	2025-01-02	09134200-9	ЛОТ №1 – Дизельне па	КП "Київпастранс" | 3172	Інший	105,020,000	2,292,513	45.81	complete
7	UA-2025-01-02-(	2025-01-02	09134200-9	ЛОТ №2 – Дизельне па	КП "Київпастранс" | 3172	Інший	105,020,000	2,305,598	45.55	complete
8	UA-2025-01-03-(	2025-01-03	09130000-9	Паливо дизельне ДП-/	КОМУНАЛЬНЕ ПІДПРИЄ	Комунальний	99,695.76	2,060	48.4	complete
9	UA-2025-01-03-(	2025-01-03	09130000-9	Нафта і дистиляти (бен	КОМУНАЛЬНЕ ПІДПРИЄ	Комунальний	99,712.9	1,810	55.09	complete
10	UA-2025-01-03-(	2025-01-03	09130000-9	ДК 021:2015:09130000-9	Комунальне підприємст	Комунальний	99,900	1,700	58.76	complete
11	UA-2025-01-03-(	2025-01-03	09130000-9	Бензин А-95 (талон)	Комунальне некомерцій	Комунальний	136,800	2,767	49.44	complete
12	UA-2025-01-03-(	2025-01-03	09130000-9	Бензин А-95 (наливом,	БІЛОЦЕРКІВСЬКИЙ НАЦІ	Державний	550,000	11,859	46.38	complete
13	UA-2025-01-03-(	2025-01-03	09132000-3	Лот 1: Бензин А-95 (тал	ДП Київський обласний	Інший	2,688,620	52,780	50.94	complete
14	UA-2025-01-03-(	2025-01-03	09134200-9	Лот 2: Дизельне паливо	ДП Київський обласний	Інший	511,675	10,152	50.4	complete
15	UA-2025-01-05-(	2025-01-05	09130000-9	Бензин А-95	Відділ культури та туризм	Комунальний	5,130	100	51.3	complete
16	UA-2025-01-06-(	2025-01-06	09130000-9	Дизельне паливо	КОМУНАЛЬНЕ ПІДПРИЄ	Комунальний	162,000	3,000	54	complete
17	UA-2025-01-06-(	2025-01-06	09130000-9	Бензин А-95 Євро в тал	Бориспільський міський	Комунальний	197,344	4,155	47.49	complete
18	UA-2025-01-06-(	2025-01-06	09130000-9	Бензин АІ-95 Преміум	Комунально-побутове пі	Комунальний	406,000	7,245	56.04	complete
19	UA-2025-01-06-(	2025-01-06	09130000-9	ДК 021:2015 09130000-9	Управління освіти та інн	Державний	59,850	1,073	55.8	complete
20	UA-2025-01-06-(	2025-01-06	09130000-9	Нафта і дистиляти	Комунальне підприємст	Комунальний	1,052,667	21,037	50.04	complete

Рис. 2.7 – Фрагмент завантажених та нормалізованих даних

2.3 Розвідувальний аналіз даних та виявлення закономірностей

Етап розвідувального етапу (*EDA, Exploratory Data Analysis*) є критичним кроком перед побудовою прогнозних моделей машинного навчання. Як зазначав засновник цього напряму статистики Джон Тьюкі у своїй класичній праці «Exploratory Data Analysis» головна мета EDA полягає у тому, щоб дозволити даним «самостійно розкрити» свою структуру, перевірити гіпотези та ідентифікувати аномалії ще до застосування формалізованих математичних методів[13].

Оскільки сформований датасет публічних закупівель пального є масштабним і багатовимірним, розвідувальний аналіз було розділено на три основні етапи: розрахунок базових описових статистик, аналіз структури замовників та дослідження часових залежностей. Усі аналітичні операції виконувалися безпосередньо в СУБД ClickHouse із використанням агрегуючих SQL-запитів.

1. Базова описова статистика

Щоб зрозуміти загальний стан ринку публічних закупівель пального було розраховано ключові макропоказники генеральної сукупності за досліджуваний період. Отримані результати зведено у таблиці 2.1

Таблиця 2.1

Описова статистика датасету

Метрика	**Значення**	**Опис**
Загальна кількість успішних лотів	3 876	Кількість проаналізованих договорів
Середня зважена ціна (грн/л)	53,4	Глобальний індикатор ринку
Мінімальна ціна (грн/л)	37,98	Нижня межа
Максимальна ціна (грн/л)	64,86	Верхня межа
Сукупний обсяг закупівель (літрів)	151 898 628	Загальний об’єм споживання

2. Структурний аналіз замовників

Однією з головних гіпотез дослідження є припущення, що тип замовника суттєво впливає на ціноутворення. Для перевірки цієї гіпотези було використано згенеровану на етапі ETL ознаку *buyer_type.* Аналіз показав (Рис. 2.8), що ринок є глибоко сегментованим. Хоча комунальні підприємства генерують найбільшу кількість дрібних тендерів, їхня середня закупівельна ціна є найвищою. Це пояснюється роздрібним характером закупівель та відсутністю власних нафтобаз. Натомість державні монополії (наприклад, АТ «Укрзалізниця» або АТ «Укргазвидобування»), незважаючи на меншу кількість процедур, закуповують

пальне гігантськими обсягами. Це дозволяє їм отримувати ціни, які є значно нижчими за середньоринковий показник, і наближаються до гуртових цін заводів-виробників.

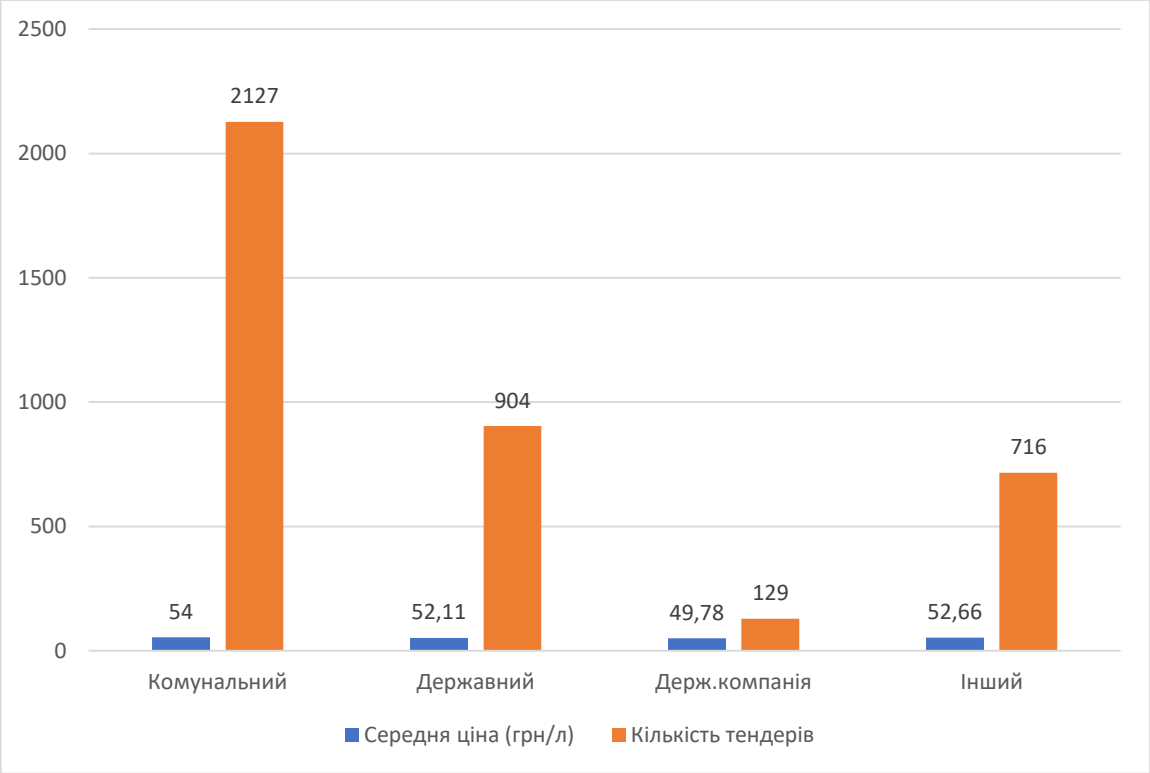


Рис. 2.8 – Вплив типу замовника на середню ціну та кількість тендерів

Розвідувальний аналіз часових рядів середніх цін на паливо за категоріями замовників дозволяє виявити ключові закономірності динаміки показників протягом 2025 року та сформувати обґрунтовані передумови для подальшого моделювання. Досліджувані дані охоплюють чотири групи замовників — комунальні, державні, державні компанії та інші — і характеризуються щомісячною частотою спостережень, що дає змогу простежити як короткострокові коливання, так і загальні тенденції. Результати цього аналізу наведено на рисунку 2.9.

Передусім варто відзначити, що починаючи з середини року, спостерігається поступове зростання середніх цін, яке досягає максимальних значень у жовтні–листопаді. Така динаміка може бути зумовлена як макроекономічними чинниками,

так і специфікою ринку пального, зокрема сезонним підвищенням попиту в осінньо-зимовий період.

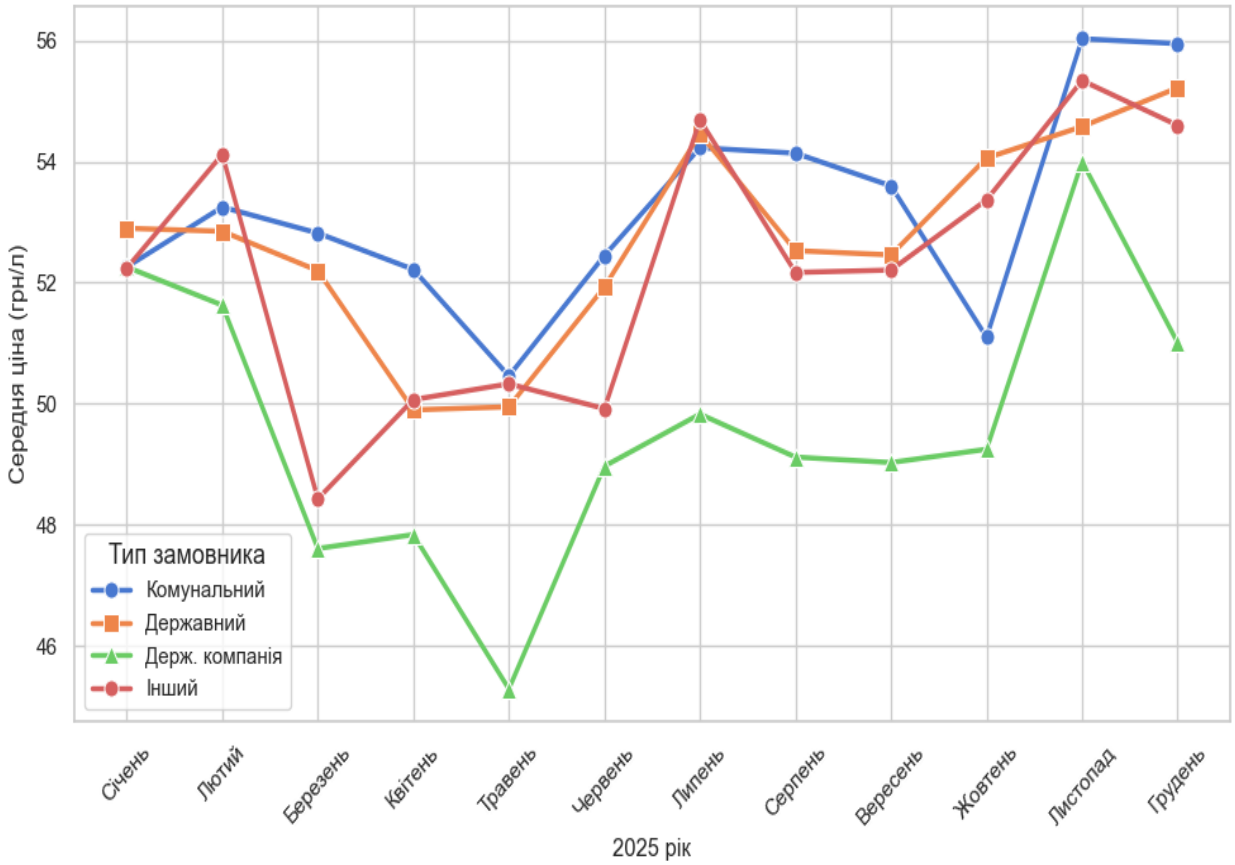


Рис. 2.9 – Динаміка середньої ціни на пальне у розрізі категорій замовників

Аналіз сезонної компоненти дозволяє виявити характерну закономірність: у весняні місяці (березень–травень) для всіх категорій замовників фіксується зниження цінових показників, причому найбільш виражене падіння спостерігається у сегменті державних компаній. Надалі, починаючи з літнього періоду, відбувається поступове відновлення цін із подальшим зростанням восени, така сезонність є типовою для енергетичних ринків.

Окремої уваги заслуговує порівняльний аналіз поведінки різних категорій замовників. Найбільш стабільну динаміку демонструють комунальні та державні установи: їхні часові ряди характеризуються відносно низькою мінливістю та плавними змінами без різких стрибків. Натомість категорії «державні компанії» та «інші» відзначаються значно вищим рівнем варіативності. У цих групах спостерігаються як різкі зниження (зокрема у весняний період), так і швидкі

зростання показників, що може бути пов’язано з більшим впливом ринкових факторів, специфікою окремих контрактів або неоднорідністю самих вибірок.

Також важливо зазначити, що у часових рядах виявлено окремі аномальні спостереження, які суттєво відхиляються від загальної тенденції. Подібні викиди можуть бути наслідком як технічних помилок при введенні даних, так і реальних, але нетипових ринкових ситуацій. Їх ідентифікація та обробка є важливим етапом підготовки даних перед побудовою моделей машинного навчання.

Таким чином, розроблений на етапі ETL архітектурний конвеєр дозволив сформувати якісний, збагачений новими ознаками масив даних. Виявлена складна природа взаємозв'язків обґрунтовує доцільність застосування методів інтелектуального аналізу даних.

РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ ПРОГНОЗУВАННЯ ПОПИТУ ТА АЛГОРИТМІВ ВИЯВЛЕННЯ АНОМАЛІЙ

3.1 Вибір та обґрунтування алгоритмів прогнозування

Як було доведено в ході розвідувального аналізу даних (Розділ 2), механізм формування цін на ринку публічних закупівель пального має яскраво виражений нелінійний характер. На кінцеву ціну одночасно впливають як часові фактори (наприклад, сезонність, макроекономічні тенденції), так і категоріальні змінні (тип замовника, обсяг закупівлі). Враховуючи ці особливості, використання традиційних підходів екстраполяції часових рядів, зокрема моделей *ARIMA*, є недоцільним, оскільки вони не здатні ефективно обробляти багатовимірні дані, що містять категоріальні ознаки.

Для вирішення поставленої задачі було розроблено архітектуру на базі методів навчання з учителем (*Supervised Learning*), де задача зводиться до регресії – прогнозування безперервної числової величини (ціни за літр пального). Відповідно до сучасних підходів у Data Science, для побудови моделі обрано три алгоритми різного рівня складності, що дає змогу провести порівняльний аналіз їхньої ефективності прогнозування:

1. Множинна лінійна регресія

Даний алгоритм застосовується як базова модель. Лінійна регресія спрямована на знаходження оптимальної гіперплощини, яка мінімізує суму квадратів відхилень між фактичними та прогнозованими значеннями. Її ключовою перевагою є висока інтерпретованість, оскільки значення коефіцієнтів безпосередньо відображають ступінь впливу окремих факторів. Водночас до недоліків належать чутливість до викидів і обмежена здатність моделювати складні нелінійні залежності між змінними.

2. Випадковий ліс

Цей підхід належить до потужних ансамблевих методів і ґрунтується на побудові великої кількості незалежних дерев рішень з подальшим усередненням їхніх прогнозів. З урахуванням результатів розвідувального аналізу, ринок пального характеризується наявністю локальних аномалій, до яких даний алгоритм є стійким. Окрім цього, він ефективно працює з нелінійними залежностями та не потребує складних процедур нормалізації ознак, що робить його практичним для задач такого типу. Як зазначає фундатор цього методу Лео Брейман, випадковий ліс являє собою комбінацію дерев рішень, кожне з яких будується на основі незалежної випадкової вибірки[14]. Серед них важливими є такі математичні властивості: *стійкість до шуму* – алгоритм значно стійкішим до статистичних викидів, ніж класична регресія; *вбудована оцінка* – внутрішні механізми алгоритму дозволяють кількісно виміряти важливість кожної змінної; *адаптивність до регресії* – алгоритм природно підтримує прогнозування неперервних числових значень, зокрема очікуваної ціни за літр пального.

3. Градієнтний бустинг

Даний алгоритм вважається сучасним галузевим стандартом для аналізу табличних даних. Як зазначає засновник цього підходу Джером Фрідман, алгоритм інтерпретує задачу апроксимації функції як оптимізаційну проблему, що розв’язується шляхом градієнтного спуску у функціональному просторі. При цьому кожне наступне дерево рішень навчається не на вихідних даних, а на залишках (помилках), отриманих попередньою моделлю, тобто фактично відбувається оцінка антиградієнта функції втрат. Такий послідовний «жадібний» підхід забезпечує досягнення максимально можливої точності прогнозування [15].

З метою узагальнення викладених теоретичних положень і остаточного обґрунтування вибору алгоритмів було здійснено їх порівняльний аналіз. Оцінювання проводилося за низкою ключових критеріїв, що є визначальними для задачі прогнозування цін на пальне: здатністю працювати з *нелінійними залежностями*, стійкістю до *викидів*, складністю налаштування *гіперпараметрів*, а

також рівнем *інтерпретованості* результатів для бізнес-користувачів. Узагальнені характеристики обраних методів машинного навчання наведено в таблиці 3.1.

Таблиця 3.1

Порівняльна характеристика обраних алгоритмів прогнозування

Критерій	**Лінійна регресія**	**Випадковий ліс**	**Градієнтний бустинг**
Здатність до узагальнення	Висока	Дуже висока	Екстремально висока
Стійкість до викидів	Низька	Висока	Середня
Складність налаштування	Мінімальна	Середня	Висока
Інтерпретованість	Пряма	Через важливість ознак	Складна

Підсумовуючи результати проведеного порівняння, варто зазначити, що в межах даного дослідження передбачено практичне навчання та тестування всіх трьох розглянутих алгоритмів. Множинна лінійна регресія використовуватиметься як базова модель, відносно якої буде оцінюватися приріст ефективності ансамблевих підходів — випадкового лісу та градієнтного бустингу. Остаточний вибір оптимальної моделі для побудови фінального прогнозу здійснюватиметься на основі порівняння значень метрик похибки, отриманих на тестовій вибірці.

Проте, незалежно від математичного апарату обраних методів, жоден із них не здатний працювати з неструктурованими текстовими даними безпосередньо. Для коректного навчання моделей необхідно перетворити категоріальні змінні у числовий формат та розділити генеральну сукупність на незалежні вибірки. Ці процеси інженерії ознак та підготовки даних виступають першим практичним етапом побудови системи прогнозування.

Інженерія ознак та формування вибірок

Ефективність моделей машинного навчання значною мірою визначається форматом і якістю вхідних даних. Оскільки обрані алгоритми працюють виключно з числовими матрицями, попередньо очищений датасет потребував додаткових перетворень.

Зокрема, текстову ознаку типу замовника *buyer_type* було закодовано за допомогою методу One-Hot Encoding. В свою чергу використання порядкового кодування для номінальних категорій призводить до штучного формування математичної ієрархії, що може викривляти результати регресійного аналізу[16]. У межах цього підходу для кожної унікальної категорії формується окрема бінарна змінна (0 або 1), що дозволяє уникнути штучного встановлення порядку або ієрархії між категоріями. Крім того, дату проведення тендеру було декомпозовано: з неї виділено тільки *числовий індекс місяця* для врахування сезонних коливань.

З метою усунення дисбалансу між ознаками різного масштабу (наприклад, обсягами закупівель у тисячах літрів) до числових змінних застосовано стандартизацію за методом Z-score. Оскільки змінні з різними масштабами можуть непропорційно впливати на функцію втрат та швидкість збіжності, особливо в алгоритмах, що базуються на оптимізації градієнтного спуску[17].

Для розв’язання цієї проблеми було використано клас *StandardScaler* із бібліотеки *scikit-learn*. Даний інструмент виконує стандартизацію даних, перетворюючи їх таким чином, щоб розподіл мав нульове середнє значення та одиничну дисперсію. Це реалізується шляхом віднімання середнього значення вибірки від кожного спостереження з подальшим діленням на стандартне відхилення[18]. У результаті всі числові ознаки приводяться до єдиного масштабу, що забезпечує їх рівнозначний вплив на початкових етапах навчання лінійних моделей і зменшує ризик перенавчання.

Фінальним етапом підготовки став поділ генеральної сукупності на тренувальну (80%) та тестову (20%) вибірки. Таке співвідношення гарантує, що алгоритми будуть перевірятися на незалежних даних, які не використовувалися у

процесі оптимізації їхніх внутрішніх ваг, що забезпечує об'єктивну оцінку прогнозної здатності системи.

Процедура навчання та порівняльний аналіз моделей

Після підготовки даних було проведено навчання трьох обраних алгоритмів. Для кожної моделі підбиралися найкращі параметри за допомогою крос-валідації, що дозволило отримати більш точні результати та уникнути перенавчання.

Процес створення і навчання моделей у середовищі Python показано у вигляді фрагмента коду (рис. 3.1). Повний програмний код, який включає завантаження даних із ClickHouse, їх обробку (*ETL*) та побудову візуалізацій, наведено у Додатку Б.

```
# Навчання та порівняння моделей
models = {
    "Лінійна регресія": LinearRegression(),
    "Випадковий ліс": RandomForestRegressor(n_estimators=100, random_state=42),
    "XGBoost": XGBRegressor(n_estimators=100, learning_rate=0.1, max_depth=5, random_state=42)
}
results = []
for name, model in models.items():
    if name == "Лінійна регресія":
        model.fit(X_train_scaled, y_train)
        preds = model.predict(X_test_scaled)
```

Рис. 3.1 – Фрагмент коду навчання моделей

Для об’єктивної оцінки ефективності розроблених моделей на тестовій вибірці було використано систему статистичних метрик:

- *Середня абсолютна похибка* (MAE) – показує на скільки гривень у середньому помиляється модель при прогнозуванні ціни одного літра пального.

- *Корінь із середньоквадратичної похибки* (RMSE) – цей показник значно знижує оцінку моделі при наявності аномально великих помилок.

- *Коефіцієнт детермінації* (R^2) - відображає, яку частину варіації цін на пальне може пояснити модель (чим ближче значення до 1.0, тим точніший прогноз).

Результати порівняльного аналізу зведено у таблиці 3.2.

Таблиця 3.2

Порівняльна характеристика точності прогнозних моделей

Алгоритм	**MAE (грн/л)**	**RMSE (грн/л)**	**Коефіцієнт R^2**
Лінійна регресія	3,82	4,74	0,14
Випадковий ліс	3,11	4,22	0,32
Градієнтний бустинг (XGBoost)	2,93	3,88	0,42

Аналіз результатів, наведених у таблиці 3.2, показує, що ансамблеві підходи забезпечують суттєво вищу точність порівняно з базовою лінійною моделлю, яка продемонструвала найгірші результати (R^2=0,14). Найефективнішим серед досліджуваних методів виявився алгоритм градієнтного бустингу (XGBoost)[19], що дозволяє пояснити близько 42% варіації ціни (R^2=0,42) при мінімальній середній абсолютній помилці на рівні 2,93 грн за літр.

Слід зауважити, що для реальних економічних даних у сфері публічних закупівель подібне значення коефіцієнта детермінації є цілком очікуваним. Воно свідчить про те, що формалізовані фактори, такі як тип замовника та часовий період, визначають значну частину ціноутворення. Водночас частина варіації, яку модель не може пояснити, відображає вплив прихованих чинників, зокрема

індивідуальних умов постачання, рівня конкуренції в межах конкретного тендера, а також можливих непрямих факторів, включаючи потенційні переплати.

Отримані результати підтверджують доцільність використання обраного підходу. Досягнута точність моделі XGBoost, із середньою похибкою менш ніж 3 грн/л, дозволяє розглядати її як надійний аналітичний інструмент. На основі її прогнозів можна формувати обґрунтований діапазон «ринкової ціни», що, у свою чергу, дає змогу виявляти контракти з аномальними відхиленнями. Практичні аспекти застосування цього підходу для ідентифікації цінових аномалій і потенційних ризиків розглядаються у наступному підрозділі.

3.2 Алгоритмічне виявлення корупційних ризиків та цінових аномалій у сфері публічних закупівель

Після проведення порівняльного аналізу різних алгоритмів і обрання моделі градієнтного бустингу (XGBoost) як найточнішої (з коефіцієнтом детермінації R^2=0,42 та середньою похибкою 2,93 грн/л), запропоновану архітектуру було використано для розв’язання ключового прикладного завдання дослідження — автоматизованого виявлення аномалій у сфері публічних закупівель. Застосування прогнозних значень моделі як динамічного цінового орієнтиру дає змогу перетворити складні математичні обчислення на прикладний інструмент раннього виявлення фінансових ризиків.

Теоретичні засади підходу до виявлення аномалій ґрунтуються на використанні методів машинного навчання, які є класичним інструментом попередньої аналітики у задачах виявлення економічного шахрайства. Інтелектуальні алгоритми здатні ефективно виявляти приховані закономірності маніпуляцій у великих обсягах фінансових даних, аналізуючи історичні приклади та відхилення від типових значень. У сфері публічних закупівель одним із ключових індикаторів потенційної корупції або штучного обмеження конкуренції є статистично значуще перевищення ціни контракту відносно ринкового рівня.

Основна концепція алгоритмічного виявлення ризиків у даній роботі ґрунтується на аналізі математичних залишків — різниці між фактичною ціною укладеного договору та «еталонною» ціною, спрогнозованою ML-моделлю. Для кожного тендеру *i* розраховується абсолютне та відносне відхилення за формулами:

$$\Delta P_i = P_{\text{факт}\_i} - P_{\text{прогн}\_i}$$

$$\delta_i = \left(\frac{\Delta P_i}{P_{\text{прогн}_i}} \right) \times 100\%$$

де $P_{\text{факт}\_i}$ – фактична ціна за літр пального, зафіксована у системі ProZorro; $P_{\text{прогн}\_i}$ – справедлива ціна, спрогнозована моделлю XGBoost на основі обсягів, типу замовника та фактора сезонності.

Встановлення порогу аномалій

Оскільки ринок пального є динамічним і схильним до природних коливань, навіть невелике позитивне відхилення між фактичною та прогнозованою ціною не можна автоматично вважати ознакою ризику. Для того щоб відокремити звичайний ринковий шум від справжніх аномалій і забезпечити автоматизований аудит, необхідно задати статистично обґрунтоване порогове значення *T*.

У задачах регресійного аналізу доцільно використовувати правило двох стандартних похибок для побудови довірчого інтервалу, який за умови наближеності розподілу відхилень до нормального охоплює близько 95% спостережень[17]. З огляду на те, що середня абсолютна похибка (MAE) розробленої моделі становить 2,93 грн/л, природні коливання ціни можна вважати такими, що перебувають у межах інтервалу $2 \times MAE$. Відповідно, будь-яке перевищення цієї розрахункової межі класифікується системою як потенційна цінова аномалія та потребує подальшого детального аналізу.

На основі цього статистичного правила було розроблено шкалу оцінки ризику закупівель, яку наведено у таблиці 3.3

Таблиця 3.3

Критерії класифікації корупційних ризиків за величиною залишку

Рівень ризику	**Діапазон відхилення**	**Інтерпретація для *MAE*=2.93**	**Дія системи**
Норма	$\Delta P_i \leq 1.5 \times \text{MAE}$	Переплата до 4.40 грн/л	Ігнорування
Середній	$1.5 \times \text{MAE} < \Delta P_i \leq 2.5 \times \text{MAE}$	Переплата від 4.41 до 7.33 грн/л	Додаткова перевірка
Аномалія	$\Delta P_i > 2.5 \times \text{MAE}$	Переплата понад 7.33 грн/л	Високий рівень ризику

Система маркує тендер як критично аномальний, якщо фактична ціна перевищує прогнозовану більше ніж на 7,33 грн/л (в межах дослідження для жорсткої фільтрації встановлено робочий поріг T = 8 грн/л).

Для математичної формалізації цього процесу, яка в подальшому застосовується для розрахунку загального економічного ефекту, запроваджено бінарний індикатор аномальності — Ризик_i. Алгоритмічне правило його формування має такий вигляд:

$$\text{Ризик}_i = \begin{cases} 1, \text{якщо } \Delta P_i > 8 \\ 0, \text{якщо } \Delta P_i \leq 8 \end{cases}$$

Відповідно, значення Ризик_i сигналізує алгоритму про необхідність включення конкретної транзакції до фінального реєстру фінансових втрат.

Практична реалізація та результати

Програмна реалізація аналітичного модуля для виявлення аномалій виконана мовою Python із застосуванням бібліотеки *pandas*, що забезпечує ефективну обробку векторизованих операцій. Алгоритм роботи скрипта базується на зіставленні тестового набору даних із прогнозними значеннями моделі XGBoost,

обчисленні залишків (відхилень) та автоматичному відсіві транзакцій за допомогою розрахованого індикатора Ризик$_i$

На рисунку 3.3 наведено основний фрагмент коду, який реалізує процедури фільтрації та ранжування потенційно ризикових транзакцій. Повний програмний код аналітичного модуля виявлення аномалій (що є логічним продовженням процедури навчання моделей, наведеної у Додатку Б) представлено в Додатку В.

```
# Розрахунок математичного залишку
results_df['Різниця'] = results_df['Актуальна ціна'] - results_df['Прогнозована ціна']
results_df['Відсоток різниці'] = (results_df['Різниця'] / results_df['Прогнозована ціна']) * 100
# Поріг аномальності
THRESHOLD = 8.0
# Виявлення Ризик_і (1 - якщо аномалія, 0 - якщо норма)
results_df['Ризик_і'] = (results_df['Різниця'] > THRESHOLD).astype(int)
# Тільки ті тендери де Ризик_і дорівнює 1
anomalies = results_df[results_df['Ризик_і'] == 1].copy()
anomalies = anomalies.sort_values(by='Різниця', ascending=False)
```

Рис. 3.3 – Фрагмент коду фільтрації аномальних закупівель

Налаштована система була запущена на відкладеній тестовій вибірці. Алгоритм в автоматичному режимі проаналізував кожну транзакцію та виявив 21 критичну аномалію (тендери з високим рівнем ризику).

У таблиці 3.4 наведено фрагмент результатів роботи алгоритму з переліком найбільш підозрілих тендерів, виявлених у досліджуваному періоді.

Таблиця 3.4

Фрагмент реєстру виявлених цінових аномалій (Топ-10 відхилень)

ID тендеру	Тип замовника	Обсяг (л)	Актуальна ціна	Прогнозована ціна	Різниця, ΔP_i
UA-2025-06-19-008104-a-L1	Комунальний	1	63.00	49.31	13.70

Продовження таблиці 3.4

UA-2025-08-12-006869-a-L1	Інший	2514	60.84	47.27	13.58
UA-2025-05-08-011532-a-L1	Державний	41193	57.00	43.79	13.22
UA-2025-10-14-013010-a-L1	Державний	1238	63.00	51.08	11.93
UA-2025-01-13-009955-a-L1	Комунальний	16119	62.04	50.39	11.66
UA-2025-12-17-013481-a-L1	Державний	7501	63.48	52.01	11.48
UA-2025-03-20-002581-a-L1	Інший	4606	57.96	46.79	11.18
UA-2025-01-23-001973-a-L1	Комунальний	1129	62.00	51.81	10.20
UA-2025-08-07-000820-a-L1	Комунальний	5518	61.80	51.62	10.19
UA-2025-05-01-004895-a-L1	Інший	8067	54.00	44.10	9.91

3.3. Візуалізація результатів та оцінка практичної ефективності впровадження системи

Виявлення аномальних відхилень у сфері публічних закупівель є лише проміжним етапом аналітичного процесу. Щоб результати, отримані за допомогою машинного навчання, могли бути застосовані для ухвалення управлінських рішень та ініціювання перевірок, їх необхідно перетворити у зрозумілий бізнес-формат. Відповідно, фінальним етапом розробки системи стало впровадження інструментів візуалізації у вигляді дашбордів, а також оцінювання потенційного економічного ефекту від її використання.

Концептуальні засади візуалізації даних

Ефективний аналітичний дашборд повинен забезпечувати подання ключової інформації в межах одного екрану, що дає змогу швидко виявляти проблемні області без надмірного когнітивного навантаження[20]. Враховуючи це, архітектура розробленої системи передбачає узагальнення результатів моделі XGBoost у форматі інтерактивних графічних представлень.

Базовий аналітичний дашборд включає три основні візуалізації:

1. *Розподіл фінансових втрат за категоріями замовників:* дає змогу визначити типи державних установ, які формують найбільші обсяги переплат.
2. *Динаміка корупційних ризиків у часі:* відображає зміну середнього значення математичного залишку за місяцями та слугує індикатором можливих системних порушень у певні періоди.
3. *Гістограма відхилень:* показує частотний розподіл різних рівнів переплат, що дозволяє відокремити звичайні ринкові коливання від суттєвих аномальних значень.

Практичну реалізацію описаних аналітичних інструментів, побудовану на основі виявлених 21 аномальних транзакцій із тестової вибірки, наведено на рисунках 3.4–3.6.

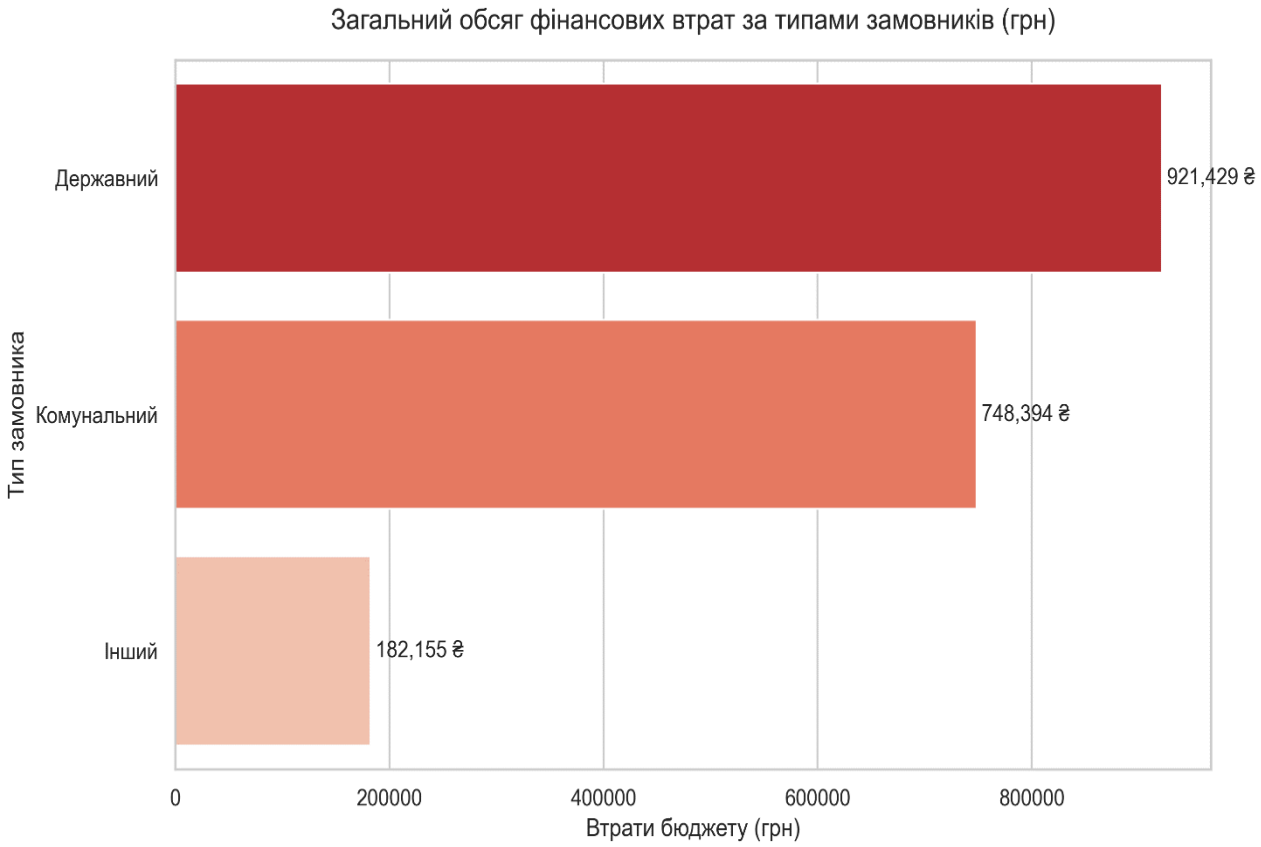


Рис. 3.4 – Розподіл загального обсягу фінансових втрат за типами замовників

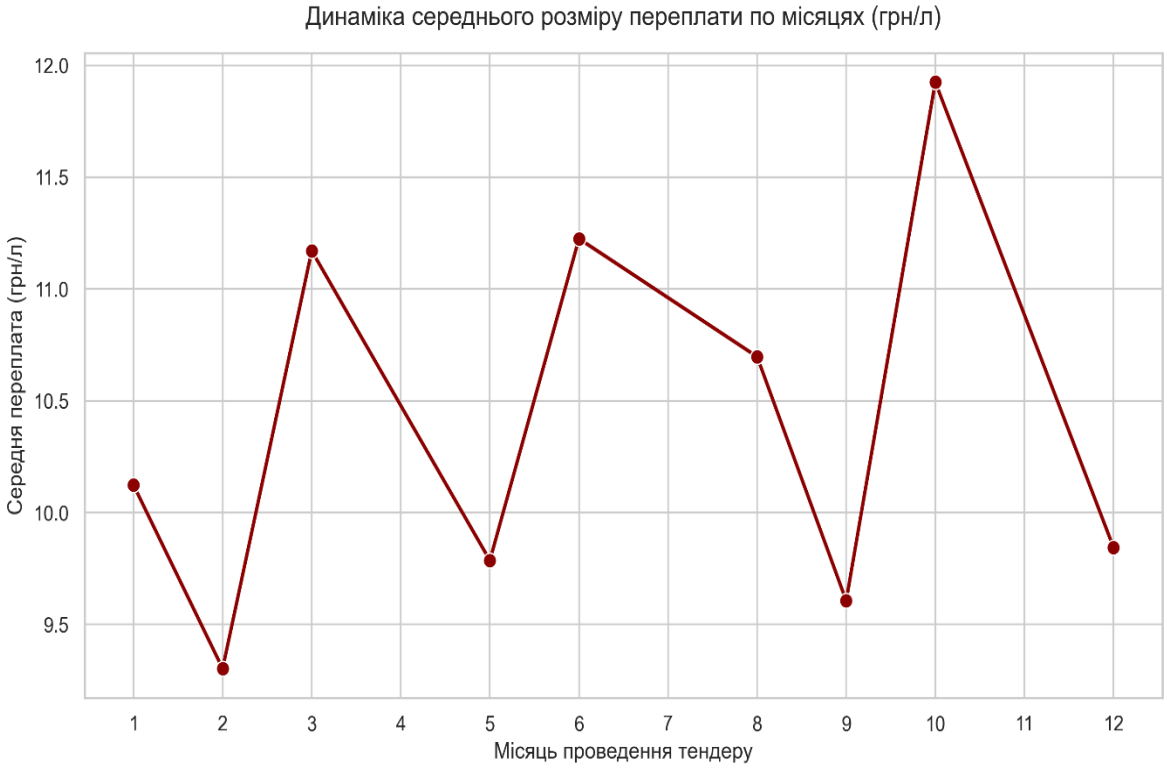


Рис. 3.5 – Динаміка середнього розміру переплати (грн/л) у часовому розрізі

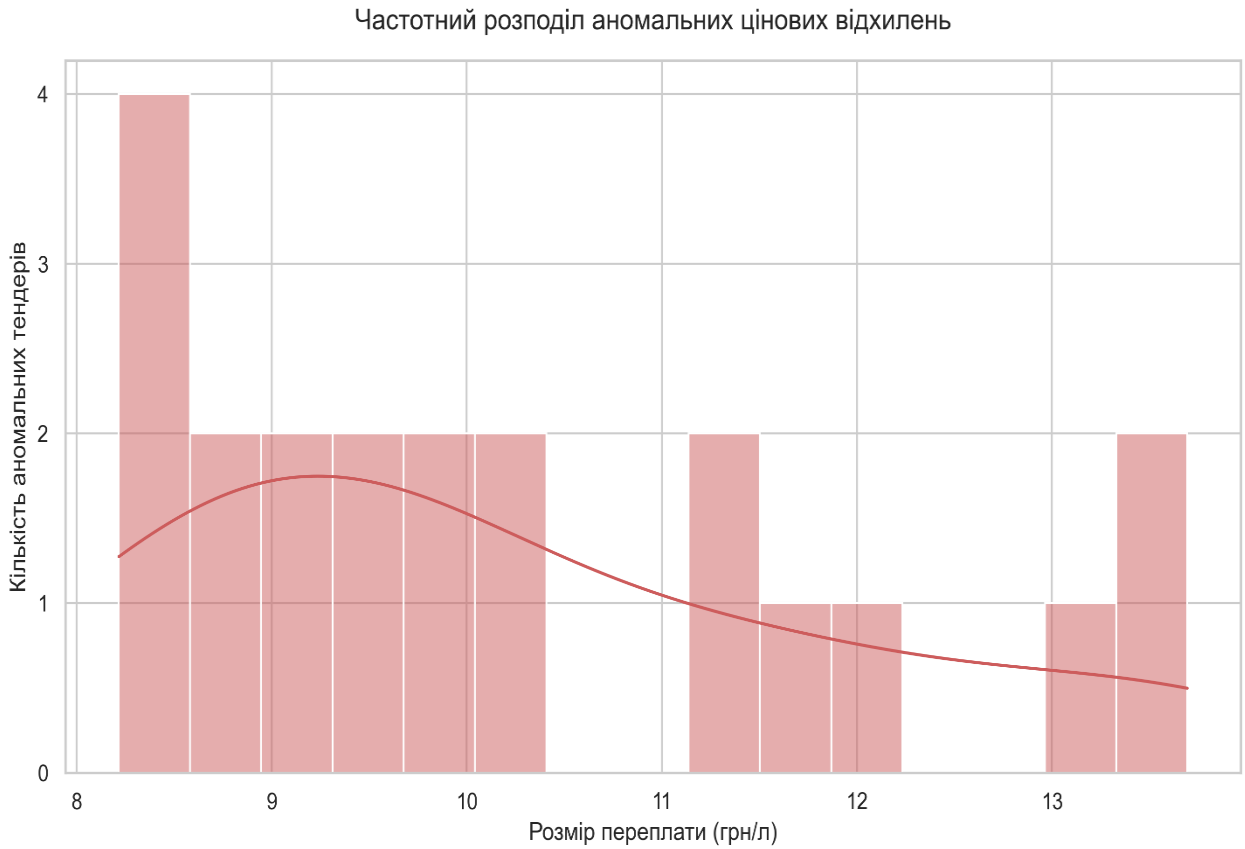


Рис. 3.6 – Частотний розподіл аномальних цінових відхилень

Ключовим критерієм доцільності впровадження будь-якої аналітичної IT-системи є її здатність створювати фінансову цінність. У контексті машинного навчання застосовується концепція «очікуваної цінності», яка дозволяє трансформувати технічні показники точності моделі у грошові еквіваленти [21].

Оскільки розроблена модель визначає величину переплати (відхилення ΔP_i) для кожного виявленого аномального тендеру, з'являється можливість кількісно оцінити фінансові втрати держави. Формула розрахунку втраченої суми для окремої закупівлі має такий вигляд:

$$\text{Втрати}_i = \Delta P_i \times V_i$$

де ΔP_i — різниця між фактичною та спрогнозованою ціною за літр (грн/л) для i-го тендеру; V_i — загальний обсяг закупленого пального (у літрах) у межах цього контракту.

Загальний економічний ефект (потенційно врятовані кошти бюджету) обчислюється як сума втрат за всіма тендерами, що подолали поріг аномальності ($T \geq 8$ грн/л):

$$\text{Загальня економія} = \sum_{i=1}^{N} (\text{Втрати}_i \times \text{Ризик}_i)$$

де $\text{Ризик}_i \in \{0, 1\}$ — бінарний індикатор аномальності тендеру.

Програмна реалізація обчислення економічного ефекту повністю відповідає описаній математичній моделі. Засобами мови Python умова перевищення порогового значення аномальності перетворюється на числовий індикатор Ризик_i, після чого виконується підсумовування втрат лише для транзакцій, віднесених до ризикових. Основний алгоритм розрахунку загального обсягу зекономлених коштів подано на рисунку 3.7.

```
# Розрахунок Втрати_і для всіх тендерів
results_df['Втрати_і'] = results_df['Різниця'] * results_df['Обсяг (л)']
# Вирахування Загальної економії за формулою
results_df['Втрати_з_урахуванням_ризику'] = results_df['Втрати_і'] * results_df['Ризик_і']
total_savings = results_df['Втрати_з_урахуванням_ризику'].sum()
print(f"Загальний потенціал економії: {total_savings:,.2f} грн")
# Формування таблиці тільки з аномаліями для виведення Топ-10 (Ризик == 1)
anomalies_final = results_df[results_df['Ризик_і'] == 1].sort_values(by='Втрати_і', ascending=False)
print("\nВтрати по Топ-10 аномальних тендерів:")
print(anomalies_final[['ID тендеру', 'Тип замовника', 'Обсяг (л)', 'Різниця', 'Втрати_і']].head(10))
```

Рис. 3.7 – Алгоритм калькуляції загальної суми збережених коштів

Практичні результати на тестовій вибірці

Застосування розробленого підходу до тестового набору даних дало змогу перевести виявлені ризики у грошовий вимір. Система автоматично ранжувала аномалії за величиною фінансових втрат, формуючи пріоритетний перелік для

подальшого аналізу аудиторами. У таблиці 3.5 подано результати оцінки збитків для десяти найбільш критичних транзакцій, ідентифікованих алгоритмом XGBoost.

Таблиця 3.5

Економічна оцінка виявлених цінових аномалій (Топ-10)

ID тендеру	**Тип замовника**	**Обсяг (л)**	**Різниця, ΔP_i (грн/л)**	**Втрати бюджету (грн)**
UA-2025-05-08-011532-a-L1	Державний	41193	13.22	544 337.22
UA-2025-01-13-008948-a-L1	Комунальний	36082	8.52	307 117.62
UA-2025-01-13-009955-a-L1	Комунальний	16119	11.66	187 939.84
UA-2025-02-05-014507-a-L1	Комунальний	12002	9.31	111 652.99
UA-2025-12-25-010549-a-L1	Державний	12200	8.22	100 242.30
UA-2025-12-17-013481-a-L1	Державний	7501	11.48	86 065.30
UA-2025-05-01-004895-a-L1	Інший	8067	9.91	79 889.47
UA-2025-05-20-010932-a-L1	Державний	8996	8.35	75 068.26
UA-2025-08-07-000820-a-L1	Комунальний	5518	10.19	56 197.90
UA-2025-06-19-008834-a-L1	Державний	6091	8.76	53 337.99

Сумарні втрати державного бюджету лише за десятьма наведеними закупівлями перевищили 1 601 908 гривень. Загальний обсяг потенційних переплат (Загальня економія) за всіма виявленими аномальними тендерами в межах тестової вибірки сягнув понад 1 851 978 млн гривень.

Узагальнюючи, результати дослідження підтверджують, що застосування методів машинного навчання, зокрема градієнтного бустингу, здатне суттєво підвищити ефективність моніторингу публічних закупівель. Перехід від реактивного ручного аналізу до предиктивного автоматизованого контролю зменшує вплив людського фактору та забезпечує своєчасне виявлення потенційних порушень, що, у свою чергу, створює значний економічний ефект для держави.

ВИСНОВКИ

У кваліфікаційній роботі вирішено актуальне науково-прикладне завдання — розроблено та обґрунтовано модель аналітичної системи прогнозування попиту та автоматизованого виявлення цінових аномалій у сфері публічних закупівель на основі технологій великих даних (Big Data).

У результаті проведеного дослідження, відповідно до поставлених у Вступі завдань, сформовано такі основні теоретичні та практичні висновки:

1. *Досліджено теоретико-методологічні особливості прогнозування попиту у державному секторі.* Встановлено, що на відміну від комерційного ринку, державні закупівлі мають жорсткий адміністративно-плановий характер, високу фрагментованість та суттєву залежність від бюджетних періодів. Доведено, що традиційні статистичні методи аналізу часових рядів є недостатньо ефективними для роботи з масивами системи ProZorro через високий рівень «шуму», нелінійність даних та наявність прихованих корупційних ризиків.
2. *Проаналізовано та обґрунтовано вибір архітектури системи зберігання та алгоритмів машинного навчання.* Визначено, що для обробки транзакцій державного сектору оптимальним є використання стовпчикових аналітичних систем (OLAP). Для задач регресії та предиктивної аналітики найбільш доцільним є застосування ансамблевих методів на основі дерев рішень, які здатні працювати з багатовимірними даними та категоріальними ознаками.
3. *Здійснено збір, обробку (ETL) та розвідувальний аналіз (EDA) відкритих даних.* Розроблено програмний модуль мовою Python для нормалізації сирих даних, фільтрації викидів за методом міжквартильного розмаху (IQR) та автоматичної класифікації замовників. Розвідувальний аналіз виявив глибоку сегментацію ринку пального (найвищі ціни фіксуються у комунальних підприємств, найнижчі – у державних монополій) та стійку сезонність із ціновими піками в осінньо-зимовий період.

4. *Спроєктовано та реалізовано аналітичне сховище на базі СУБД ClickHouse.* Створено фізичну модель даних із застосуванням підходу денормалізації (формування єдиної «широкої» таблиці). Використання рушія MergeTree із розрідженими індексами та складеним ключем сортування дозволило забезпечити надшвидку агрегацію історичних транзакцій, підготувавши якісний фундамент для ML-моделювання.
5. *Розроблено та оптимізовано програмні алгоритми прогнозування попиту і виявлення аномалій.* Проведено порівняльний аналіз трьох моделей (Лінійна регресія, Випадковий ліс, Градієнтний бустинг). Найвищу точність продемонстрував алгоритм XGBoost (коефіцієнт детермінації R^2 = 0,42, середня похибка 2,93 грн/л). На базі розрахованих математичних залишків створено алгоритм виявлення ризиків із порогом спрацювання T = 8 грн/л. У результаті автоматизованого сканування тестової вибірки система успішно ідентифікувала 21 критичну транзакцію з ознаками аномального завищення ціни.
6. Розроблено інструменти візуалізації та проведено оцінку економічної ефективності. Результати роботи алгоритмів інтегровано в аналітичні дашборди для зручного бізнес-аналізу. Практична монетизація виявлених ризиків засвідчила, що сумарні фінансові втрати держави лише за топ-10 аномальними тендерами перевищили 1,6 млн грн, а загальний обсяг потенційної економії в межах досліджуваної вибірки сягнув понад 1,85 млн гривень.

Практичні рекомендації щодо подальшого використання та впровадження:

Розроблена комплексна програмно-аналітична інфраструктура, яка поєднує високопродуктивне сховище даних ClickHouse та предиктивні моделі машинного навчання, являє собою готовий мінімально життєздатний продукт (MVP). З огляду на підтверджену високу економічну ефективність, рекомендується:

- *Для Державної аудиторської служби України та НАБУ*: впровадити розроблений алгоритм як систему раннього попередження. Перехід від

ручного ретроспективного аудиту до машинного контролю дозволить автоматично генерувати пріоритетні списки для перевірок ще на етапі укладання договорів.

- *Для громадських антикорупційних організацій (наприклад, Dozorro):* використовувати запропоновані дашборди для моніторингу динаміки переплат у розрізі окремих регіонів та типів замовників.
- *Перспективи розвитку:* у подальшому розроблену систему доцільно масштабувати шляхом інтеграції модуля інкрементального оновлення через REST API ProZorro для обробки даних у режимі реального часу та розширення предиктивної моделі на інші високоризикові категорії товарів (електроенергія, продукти харчування, медикаменти).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Про публічні закупівлі : Закон України від 25.12.2015 № 922-VIII. URL: https://zakon.rada.gov.ua/laws/show/922-19
2. Data Standard – Open Contracting Partnership. Open Contracting Partnership. URL: https://www.open-contracting.org/data-standard/
3. Gandomi A., Haider M. Beyond the hype: Big data concepts, methods, and analytics. International Journal of Information Management. 2015. Vol. 35, Iss. 2. P. 137–144.
4. Kleppmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. O'Reilly Media, 2017. P. 96.
5. Géron A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. 2nd ed. O'Reilly Media, 2019. P. 5-266.
6. Синєглазов В. М., Чумаченко О. І. Методи та технології напівкерованого навчання: курс лекцій. Київ : КПІ ім. Ігоря Сікорського, 2022. С. 31.
7. Liu F. T., Ting K. M., Zhou Z. H. Isolation Forest. 2008 Eighth IEEE International Conference on Data Mining. 2008. P. 413–422.
8. Ester M., Kriegel H. P., Sander J., Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96). 1996. P. 226–231.
9. Kimball R., Ross M. The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling. 2nd ed. Wiley, 2002. P. 401.
10. McKinney W. Data Structures for Statistical Computing in Python. Proceedings of the 9th Python in Science Conference. 2010. P. 51–56.
11. Han J., Kamber M., Pei J. Data Mining: Concepts and Techniques. 3rd ed. Morgan Kaufmann, 2011. P. 49.
12. ClickHouse Documentation: MergeTree Family. ClickHouse. URL: https://clickhouse.com/docs/en/engines/table-engines/mergetree-family/mergetree
13. Tukey J. W. Exploratory Data Analysis. Addison-Wesley, 1977. P. 39–44.

14. Breiman L. Random Forests. Machine Learning. 2001. Vol. 45, Iss. 1. P. 5–32.
15. Friedman J. H. Greedy function approximation: a gradient boosting machine. Annals of Statistics. 2001. Vol. 29, № 5. P. 1195.
16. Kuhn M., Johnson K. Applied Predictive Modeling. Springer, 2013. P. 47–48.
17. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed. Springer, 2009. P. 52.
18. sklearn.preprocessing.StandardScaler. Scikit-learn. URL: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html
19. Chen T., Guestrin C. XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016. P. 785–794.
20. Few S. Information Dashboard Design: The Effective Visual Communication of Data. O'Reilly Media, 2006. 223 p.
21. Provost F., Fawcett T. Data Science for Business: What you need to know about data mining and data-analytic thinking. O'Reilly Media, 2013. P. 187–195.

ДОДАТКИ

Додаток А

```
import pandas as pd
import clickhouse_connect
import numpy as np

print("Початок завантаження")
file_name = 'b79ca308-9067-4f6e-8e73-409e2b77bc77.xlsx'
try:
    df_raw = pd.read_excel(file_name)
    print(f"Успішно зчитано {len(df_raw)} рядків з файлу.")
except FileNotFoundError:
    print(f"Помилка: Файл {file_name} не знайдено!")
    exit()
# Видалення нерозривних пробілів (\xa0) та звичайних пробілів, заміна коми на крапку
def clean_money_format(series):
    cleaned = series.astype(str).str.replace(r'[\s\xa0]', '', regex=True).str.replace(',', '.')
    return pd.to_numeric(cleaned, errors='coerce')
# Етап TRANSFORM
df_mapped = pd.DataFrame()
# Зіставлення колонок з файлу зі схемою бази даних
df_mapped['tender_id'] = df_raw['Ідентифікатор лота']
df_mapped['tender_date'] = pd.to_datetime(df_raw['Дата оголошення лота']).dt.date
df_mapped['cpv_code'] = df_raw['Класифікація CPV'].astype(str).str.split(' ').str[0]
df_mapped['item_name'] = df_raw['Лот']
df_mapped['buyer_region'] = df_raw['Організатор']
#Створення категорій замовників
conditions = [
    # Комунальні
    df_mapped['buyer_region'].str.contains('комунал|міськ|селищ|сільськ|лікарня|школа|водоканал', case=False, na=False),
    # Державні монополії та великі АТ
    df_mapped['buyer_region'].str.contains(
        'укрзалізниця|укргазвидобування|нафтогаз|енергоатом|укренерго|укрпошта|укрнафта', case=False, na=False),
    # Державні
    df_mapped['buyer_region'].str.contains('міністерство|національн|державн|військов|агентство|служба|зсу', case=False,
                                           na=False)
]
# Відповідні назви категорій для цих правил
choices = [
    'Комунальний',
    'Держ. компанія',
    'Державний'
]
```

```python
# Нова колонка
df_mapped['buyer_type'] = np.select(conditions, choices, default='Інший')

df_mapped['expected_value'] = clean_money_format(df_raw['Очікувана вартість'])
df_mapped['unit_price'] = clean_money_format(df_raw['Ціна за одиницю'])
# Обрахунок кількості
df_mapped['item_quantity'] = np.where(
    df_mapped['unit_price'] > 0,
    df_mapped['expected_value'] / df_mapped['unit_price'],
    y: 0
)
df_mapped['item_quantity'] = df_mapped['item_quantity'].round(0)
df_mapped['status'] = 'complete'
df_mapped = df_mapped.dropna(subset=['unit_price'])
df_mapped = df_mapped[df_mapped['unit_price'] > 0]
# Фільтрація викидів (Метод IQR)
initial_count = len(df_mapped)
Q1 = df_mapped['unit_price'].quantile(0.25)
Q3 = df_mapped['unit_price'].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR
df_cleaned = df_mapped[(df_mapped['unit_price'] >= lower_bound) & (df_mapped['unit_price'] <= upper_bound)]
print(f"Відфільтровано математичних викидів: {initial_count - len(df_cleaned)}")
print(f"Готово до завантаження: {len(df_cleaned)} чистих тендерів.")
# Завантаження в ClickHouse
CH_HOST = '192.168.0.103'
CH_PORT = 8123
CH_USER = 'default'
CH_PASSWORD = '12345678'
try:
    client = clickhouse_connect.get_client(
        host=CH_HOST, port=CH_PORT, username=CH_USER, password=CH_PASSWORD
    )
    client.insert_df( table: 'prozorro_tenders', df_cleaned)
    print("Дані з файлу успішно завантажено у сховище ClickHouse!")
except Exception as e:
    print(f"Помилка підключення або запису в БД: {e}")
```

Додаток Б

```python
import clickhouse_connect
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from xgboost import XGBRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
# Підключення до БД ClickHouse
client = clickhouse_connect.get_client(
    host='192.168.0.108',
    port=8123,
    username='default',
    password='12345678',
    database='default'
)
# Витягування даних через SQL-запит, беремо тільки потрібні дані
query = """
        SELECT tender_id, \
               tender_date, \
               buyer_type, \
               item_quantity, \
               unit_price
        FROM prozorro_tenders
        WHERE unit_price > 0 \
          AND item_quantity > 0 \
        """
# Отримуємо дані одразу у форматі DataFrame
df = client.query_df(query)
print(f"Дані успішно завантажено. Кількість записів: {len(df)}")
# Інженерія ознак
# Обробка дат
df['tender_date'] = pd.to_datetime(df['tender_date'])
df['month'] = df['tender_date'].dt.month
```

```
# Трансформація категоріальних ознак
df_encoded = pd.get_dummies(df, columns=['buyer_type'], drop_first=True)
# Визначаємо цільову змінну (y) та предиктори (X)
y = df_encoded['unit_price']
X = df_encoded.drop( labels: ['unit_price', 'tender_date', 'tender_id'], axis=1)

# Формування вибірок та масштабування
X_train, X_test, y_train, y_test = train_test_split( *arrays: X, y, test_size=0.2, random_state=42)
# Стандартизація
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
# Навчання та порівняння моделей
models = {
    "Лінійна регресія": LinearRegression(),
    "Випадковий ліс": RandomForestRegressor(n_estimators=100, random_state=42),
    "XGBoost": XGBRegressor(n_estimators=100, learning_rate=0.1, max_depth=5, random_state=42)
}
results = []
for name, model in models.items():
    if name == "Лінійна регресія":
        model.fit(X_train_scaled, y_train)
        preds = model.predict(X_test_scaled)
    else:
        model.fit(X_train, y_train)
        preds = model.predict(X_test)
    # Розрахунок метрик
    mae = mean_absolute_error(y_test, preds)
    rmse = np.sqrt(mean_squared_error(y_test, preds))
    r2 = r2_score(y_test, preds)
    results.append({
        "Алгоритм": name,
        "MAE (грн/л)": round(mae, 2),
        "RMSE (грн/л)": round(rmse, 2),
        "R2": round(r2, 2)
    })
# Підсумки
comparison_df = pd.DataFrame(results)
print("\nРезультати порівняння моделей:")
print(comparison_df)
final_model = models["XGBoost"]
```

Додаток В

```python
# Отримання прогнозів від найкращої моделі на тестовій вибірці
y_pred_test = final_model.predict(X_test)

# Формування DataFrame
results_df = pd.DataFrame({
    'ID тендеру': df.loc[X_test.index, 'tender_id'],
    'Тип замовника': df.loc[X_test.index, 'buyer_type'],
    'Обсяг (л)': df.loc[X_test.index, 'item_quantity'],
    'Актуальна ціна': y_test,
    'Прогнозована ціна': y_pred_test
})
# Розрахунок математичного залишку
results_df['Різниця'] = results_df['Актуальна ціна'] - results_df['Прогнозована ціна']
results_df['Відсоток різниці'] = (results_df['Різниця'] / results_df['Прогнозована ціна']) * 100
# Поріг аномальності
THRESHOLD = 8.0
# Виявлення Ризик_і (1 - якщо аномалія, 0 - якщо норма)
results_df['Ризик_і'] = (results_df['Різниця'] > THRESHOLD).astype(int)
# Тільки ті тендери де Ризик_і дорівнює 1
anomalies = results_df[results_df['Ризик_і'] == 1].copy()
anomalies = anomalies.sort_values(by='Різниця', ascending=False)
print(f"Просканово тендерів у тестовій вибірці: {len(results_df)}")
print(f"Знайдено критичних аномалій: {len(anomalies)}")
pd.set_option( *args: 'display.max_columns', None)
pd.set_option( *args: 'display.width', 1000)
# Виведення Топ-10 найбільш підозрілих тендерів
print("\nТоп-10 тендерів з найбільшим ризиком переплати:")
print(anomalies[['ID тендеру', 'Тип замовника', 'Обсяг (л)', 'Актуальна ціна', 'Прогнозована ціна', 'Різниця']].head(10))
```

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

СИСТЕМА ПРОГНОЗУВАННЯ ПОПИТУ НА ОСНОВІ ТЕХНОЛОГІЙ ВЕЛИКИХ ДАНИХ

ВИКОНАВ:
Олег ПРАЧ
гр. САД-41

НАУКОВИЙ КЕРІВНИК:
Михайло КУЗЬМІЧ
PhD, доцент кафедри ІСТ

Державний університет інформаційно-комунікаційних технологій
Київ, 2026 рік

Актуальність дослідження

Проблема Big Data

Впровадження системи ProZorro згенерувало терабайти інформації, яка щодня оновлюється. **Ручний ретроспективний контроль** такої кількості транзакцій є фізично неможливим.

Математичний фактор

Класичні статистичні підходи та лінійні методи не здатні якісно обробляти **розрізнену, нелінійну та зашумлену** базу даних державних закупівель.

Економічний ефект

Необхідність створення автоматизованих систем **превентивного моніторингу** для запобігання ціновим аномаліям та корупційним ризикам до моменту оплати за договорами.

Мета та завдання дослідження

МЕТА РОБОТИ

Розробка масштабованої інтелектуальної системи для предиктивного моделювання цінових параметрів попиту та автоматичного виявлення цінових аномалій у сфері публічних закупівель.

Дослідити методологію Big Data та ML

Спроектувати сховище даних на базі ClickHouse.

Побудувати наскрізний ETL-скрипт очищення даних.

Навчити предиктивні ML-моделі та порівняти їх.

Розробити алгоритм детекції ризиків та оцінити ефект.

Об'єкт та предмет дослідження

Об'єкт дослідження

Процес управління, планування фінансового попиту та формування очікуваної вартості лотів у системі державних публічних закупівель.

Предмет дослідження

Моделі машинного навчання, архітектурні рішення стовпчикових баз даних (ClickHouse) та статистичні методи аналізу залишків для виявлення аномалій.

Методологія обробки великих даних (Big Data)

Концепція та виклики Big Data (Парадигма 3V)

Volume

Обсяг даних

Пакетна та потокова обробка терабайтних історичних масивів транзакцій ProZorro без втрати швидкодії.

Velocity

Швидкість обробки

Необхідність миттєвого виконання аналітичних OLAP-запитів до бази даних у режимі реального часу.

Variety

Різноманітність

Робота зі слабоструктурованими даними (текстові поля тендерів, суміш числових і категоріальних ознак).

Класифікація досліджених методів

Методи зберігання та агрегації

Досліджено концепцію OLAP (Online Analytical Processing) та переваги стовпчикових СУБД над традиційними реляційними моделями при роботі з великими аналітичними вибірками.

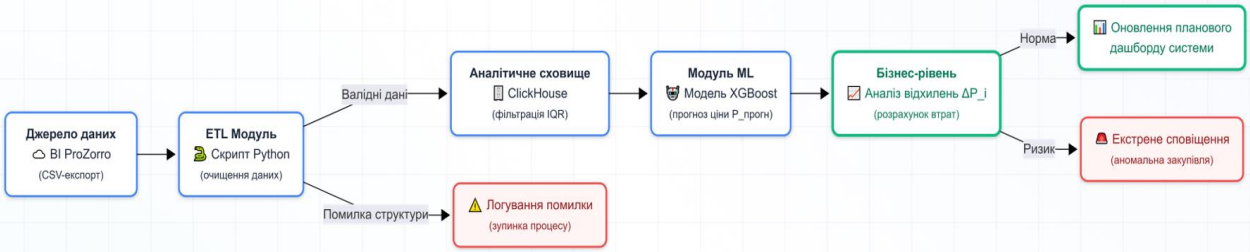
Методи інтеграції даних (ETL)

Досліджено конвеєрні процеси видобування (Extract), трансформації (Transform) та завантаження (Load) великих масивів з використанням векторизованих обчислень.

Предиктивні методи (Data Mining & ML)

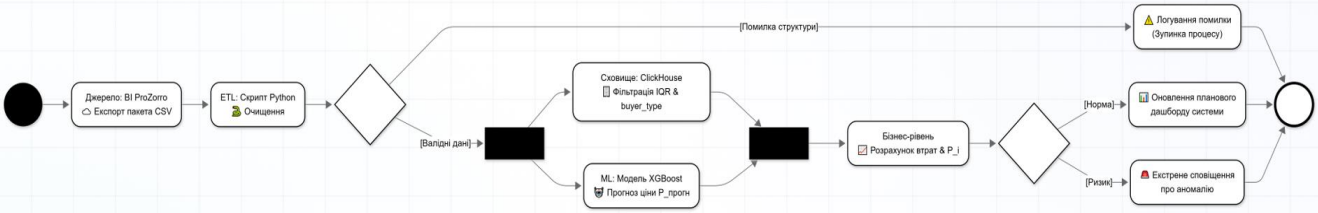
Досліджено алгоритми ансамбльованого машинного навчання як інструмент інтелектуального аналізу нелінійних ринкових процесів.

Архітектура системи



Наскрізна архітектура забезпечує автоматизований рух від сирих транзакційних даних до готових бізнес-рішень.

Об'єктно-орієнтована модель (UML)



Принцип модульності та автоматизація

Система спроєктована за принципом розділення відповідальності. Кожен блок (ETL,ClickHouse, ML) працює незалежно.

UML-модель передбачає повну автоматизацію переходу від очищення даних (IQR-метод) до фінальної візуалізації.

Логічні рівні

- **Data рівень:** Забезпечує інтеграцію з BI ProZorro та первинну обробку через регулярні вирази (RegExp).
- **Storage рівень:** Використання СУБД ClickHouse забезпечує цілісність даних та миттєвий доступ до мільйонів записів.
- **Intelligence рівень:** Предиктивне ядро XGBoost та логічний блок RiskAnalyzer для обчислення математичних залишків.

Формування датасету та математичне очищення

Параметри вибірки

- **Категорія:** 09130000-9 (Нафта і дистиляти / Пальне)
- **Регіон:** Київська область
- **Період:** 2025 рік
- **Обсяг:** 3 876 успішних лотів

Математичний апарат очищення від помилок

Формула міжквартильного розмаху:

$$IQR = Q_3 - Q_1$$

Довірчий інтервал норми:

$$X \in Q_1 - 1.5 \times IQR \; ; \; Q_3 + 1.5 \times IQR$$

Метод IQR належить до робастної статистики. Він не викривляється під впливом екстремальних значень і дозволяє **автоматично видалити технічні помилки вводу** (наприклад, ціну 5000 грн/л замість 50 грн/л).

Інфраструктура Big Data: СУБД ClickHouse

Стовпчасте зберігання

Стовпчикова модель зберігання даних (OLAP). СУБД зчитує з диска виключно потрібні стовпці, що **знижує дискове навантаження на 90%** порівняно з рядковими базами.

MergeTree Індикація

Складений первинний ключ сортування. Через розріджену структуру індексу (index_granularity = 8192) пошук по мільйонах записів займає **менше 50 мілісекунд**.

Оптимізація

Застосування **повної денормалізації** (створення єдиної "широкої" таблиці) дозволило ліквідувати важкі та повільні реляційні операції.

Відкритий код

Відкритий вихідний код гарантує можливість **безкоштовного розгортання** інфраструктури в державних установах без регулярних ліцензійних витрат.

Навчання та математична оцінка моделей

Модель	Коефіцієнт детермінації (R^2)	Середня похибка (MAE)
Лінійна регресія	0,14	3,82 грн/л
Випадковий ліс (Random Forest)	0,32	3,11 грн/л
XGBoost (Градієнтний бустинг)	**0,42**	**2,93 грн/л**

Середня абсолютна похибка:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} y_i - \hat{y}_i$$

Коефіцієнт детермінації:

$$R^2 = 1 - \frac{\sum y_i - \hat{y}_i^2}{\sum y_i - \bar{y}^2}$$

Висновок: Алгоритм XGBoost показав найкращу предиктивну здатність, послідовно мінімізуючи залишки попередніх дерев, що дозволило ефективно врахувати нелінійність ринку.

Алгоритмічне виявлення аномалій

Фактична ціна

-

Прогнозована ціна

=

Залишок ΔP

↓

Поріг 8,0 грн/л

↓

RiskFlag

Математичний залишок:

$$\Delta P_i = P_{факт_i} - P_{прогн_i}$$

Статистичне обґрунтування порогу:

Похибка кращої моделі MAE = 2,93 грн/л.

Природні коливання в межах довірчого інтервалу: 2 × MAE ≈ 5,86 грн/л.

З метою виключення хибнопозитивних спрацювань, робочий поріг детекції встановлено **T = 8,0 грн/л**.

Логічне правило **Ризику**:

$$RiskFlag_i = \begin{cases} 1, & якщо \Delta P_i > 8.0 (Аномалія) \\ 0, & якщо \Delta P_i \leq 8.0 (Норма) \end{cases}$$

Ефективність впровадження системи

1,85 млн ₴

сумарний обсяг переплат, виявлений алгоритмом

86%

усіх втрат (1,6 млн ₴) сконцентровано всього у 10 критичних тендерах (Закон Парето)

Калькуляція фінансового ефекту:

$$Loss_i = \Delta P_i \times V_i \times RiskFlag_i$$

Превентивність

Виявлення штучного завищення цін на етапі планування, тобто **до того, як кошти будуть списані** з бюджету.

Автоматизація

Повна заміна ручного перегляду документів суцільним алгоритмічним ML-контролем **100% транзакцій**.

Масштабованість

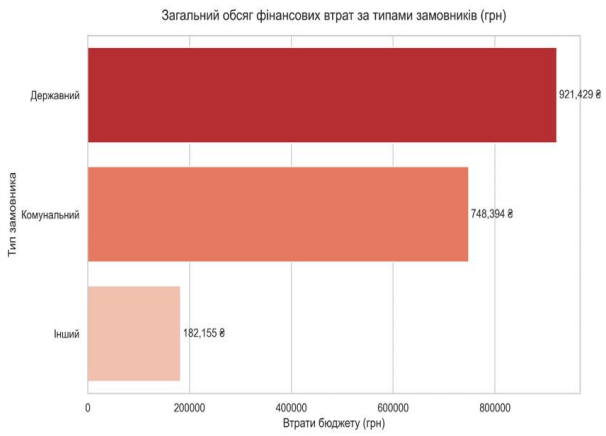
Архітектура ClickHouse готова до обробки мільярдів записів без деградації швидкості при **додаванні нових категорій** (ліки, енергія).

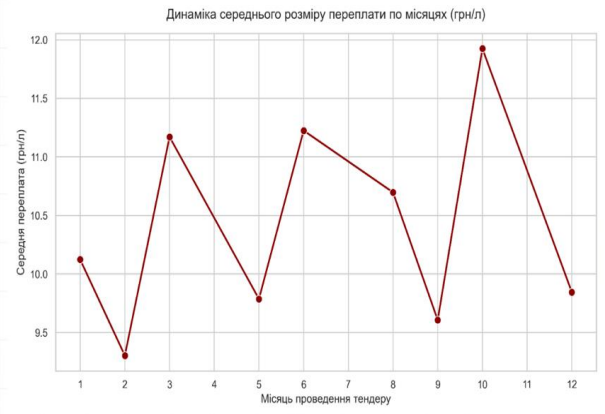
Реєстр критичних аномалій (Топ-випадки)

ID Тендеру ProZorro	Тип замовника	Переплата (ΔP_i)	Втрати бюджету
UA-2025-05-08-011532	Державний	+13,22 грн/л	544 337 ₴
UA-2025-01-13-008948	Комунальний	+8,52 грн/л	307 118 ₴
UA-2025-01-13-009955	Комунальний	+11,66 грн/л	187 940 ₴
UA-2025-02-05-014507	Комунальний	+9,31 грн/л	111 653 ₴
UA-2025-12-25-010549	Державний	+8,22 грн/л	100 242 ₴

Всього в автоматичному режимі система виявила **21 критичну аномалію**.

Візуалізація результатів аналізу





Аналітичний інсайт: Державний сектор проводить меншу кількість тендерних процедур, але за рахунок великих обсягів лотів він генерує абсолютну більшість прямих фінансових втрат (921 тис. грн). Комунальні підприємства натомість мають вищу частоту дрібних порушень (748 тис. грн сумарно).

Висновки

01 Успішно спроєктовано та впроваджено предиктивну Big Data інфраструктуру на базі стовпчикової СУБД ClickHouse та моделей машинного навчання.

02 Доведено ефективність алгоритму XGBoost (MAE = 2,93 грн/л), що дозволило статистично обґрунтувати поріг детекції аномалій (8 грн/л).

03 Практичне тестування системи на реальних даних ProZorro успішно локалізувало 21 аномалію із сумарним обсягом втрат понад 1,85 млн гривень.

04 Створений мінімально життєздатний продукт (MVP) є повністю масштабованим і готовим до інтеграції в процеси антикорупційного моніторингу.

ДЯКУЮ ЗА УВАГУ!

Готовий відповісти на ваші запитання