Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

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

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Система аналізу та прогнозування ринкових показників із застосуванням інтелектуальних підходів
(тема)

Виконав:
здобувач четвертого року навчання,
групи ІТШІ-22-1

Георгій КАРПОВ
(власне ім'я, прізвище)

Спеціальність 122 Комп'ютерні науки

(код і повна назва спеціальності)
Тип програми освітньо-професійна
Освітня програма Штучний інтелект

(повна назва освітньої програми)

Керівник ст. викл. Олександр СТЬОПІН
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ _____ Лариса ЧАЛА
(підпис) (власне ім'я, прізвище)

2026 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Штучного інтелекту

Рівень вищої освіти перший (бакалаврський)

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

Тип програми освітньо-професійна

Освітня програма Штучний інтелект
(повна назва)

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

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Карпову Георгію Дмитровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Система аналізу та прогнозування ринкових показників із застосуванням інтелектуальних підходів

затверджена наказом університету від 5 червня 20 26 р. № 488Ст
2. Термін подання здобувачем роботи до екзаменаційної комісії 10 липня 20 26 р.
3. Вихідні дані до роботи науково-технічні публікації з аналізу ринкових даних, відкритий датасет Craigslist Vehicles, документація з методів машинного навчання, матеріали щодо CatBoost, Gradient Boosting, часових рядів, очищення даних, інтерактивної візуалізації та тестування програмних систем.

4. Перелік питань, що потрібно опрацювати в роботі
1) Аналіз предметної галузі та постановка задачі
2) Обґрунтування методів та технологій розробки системи
3) Розробка системи аналізу та прогнозування ринкових показників

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	08.06.2026	виконано
2	Аналіз предметної галузі та теоретичних джерел	12.06.2026	виконано
3	Характеристика вхідних даних та методів їх	16.06.2026	виконано
	підготовки		
4	Огляд інтелектуальних підходів до	19.06.2026	виконано
	прогнозування ринкових показників		
5	Обґрунтування вибору моделей машинного	22.06.2026	виконано
	навчання та метрик оцінювання		
6	Розробка програмного конвеєра обробки даних,	27.06.2026	виконано
	навчання моделей та формування прогнозів		
7	Оформлення пояснювальної записки	01.07.2026	виконано
8	Перевірка на академічний плагіат	03.07.2026	виконано
9	Нормоконтроль	05.07.2026	виконано
10	Підготовка презентації та доповіді	06.07.2026	виконано
11	Попередній захист	07.07.2026	виконано
12	Захист перед ЕК	10.07.2026	

Дата видачі завдання 8 червня 20 26 р.

Здобувач _____
(підпис)

Керівник роботи _____ ст. викл. Олександр СТЬОПІН
(підпис) (посада, власне ім’я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 59 с., 6 рис., 7 табл., 1 дод., 18 джерел.

МАШИННЕ НАВЧАННЯ, РИНКОВІ ПОКАЗНИКИ, РИНОК, ПРОГНОЗУВАННЯ, ЧАСОВІ РЯДИ, CATBOOST, TRUST-SCORE.

Об'єкт дослідження – процес аналізу та прогнозування ринкових показників на основі відкритого датасету електронного майданчика Craigslist Vehicles, що містить оголошеня про продаж вживаних автомобілів.

Предмет дослідження – методи порогової фільтрації та виявлення фінансової лексики для очищення табличних даних; формування ознак із частотним кодуванням; прогнозування ціни засобами CatBoostRegressor; розрахунок ринкових індикаторів; оцінювання довіри до оголошень на основі out-of-fold прогнозів; інтерактивна візуалізація результатів.

Мета роботи – розробити модульну програмну систему аналізу та прогнозування ринкових показників, що реалізує повний конвеєр від завантаження вихідних даних до інтерактивного аналітичного інтерфейсу із застосуванням методів машинного навчання.

Методи дослідження – аналіз предметної галузі та огляд методів машинного навчання; програмна реалізація конвеєру обробки даних; навчання CatBoostRegressor та GradientBoostingRegressor; модульне тестування засобами pytest. Методи розробки базуються на Python з бібліотеками CatBoost, scikit-learn, pandas та фреймворком Streamlit.

У роботі реалізовано сім програмних модулів, що утворюють послідовний конвеєр: від очищення даних і формування ознак до прогнозування ціни, розрахунку ринкових індикаторів, оцінювання довіри та інтерактивного дашборду.

ABSTRACT

Bachelor’s thesis contains: 59 pp., 6 fig., 7 tabl., 1 ann., 18 references.

CATBOOST, FORECASTING, MACHINE LEARNING, MARKET, MARKET INDICATORS, TIME SERIES, TRUST-SCORE.

The object of research is the process of analyzing and forecasting market indicators based on the open dataset of the Craigslist Vehicles online marketplace containing used car listings.

The subject of research encompasses methods of threshold filtering and financial lexicon detection for tabular data cleaning; feature engineering with frequency encoding; price prediction using CatBoostRegressor; aggregated market indicator calculation; listing trust assessment based on out-of-fold predictions; interactive result visualization using Streamlit.

The purpose of the work is to develop a modular software system for analyzing and forecasting market indicators, implementing a complete pipeline from raw data ingestion to an interactive analytical interface using machine learning methods.

The research methods include domain analysis and review of machine learning methods; software implementation of the data processing pipeline; training of CatBoostRegressor and GradientBoostingRegressor; modular testing using pytest. The development methods are based on Python with CatBoost, scikit-learn, pandas libraries and the Streamlit framework.

The thesis implements seven software modules forming a sequential pipeline: from data cleaning and feature engineering to price prediction, market indicator calculation, trust assessment and an interactive dashboard. System functionality is confirmed by successful execution of 26 automated pytest tests.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів 8
Вступ 10
1 Аналіз предметної галузі та постановка задачі 12
1.1 Актуальність аналізу та прогнозування ринкових показників 12
1.2 Характеристика ринку вживаних автомобілів як прикладної області дослідження 14
1.3 Особливості табличних ринкових даних та проблеми їх підготовки 15
1.4 Огляд інтелектуальних підходів до прогнозування ціни та ринкової динаміки 17
1.5 Огляд існуючих програмних рішень для аналізу ринкових даних 19
1.5.1 Маркетплейси та агрегатори оголошень 20
1.5.2 Спеціалізовані сервіси оцінювання вартості автомобілів 21
1.5.3 Методи машинного навчання для прогнозування ціни 22
1.6 Постановка задачі дослідження 24
2 Обґрунтування методів та технологій розробки системи 26
2.1 Загальний підхід до побудови системи аналізу та прогнозування 26
2.2 Методи очищення, фільтрації та контролю якості даних 27
2.3 Формування ознак для прогнозування ринкової ціни 29
2.4 Обґрунтування використання CatBoostRegressor для прогнозування ціни автомобіля 31
2.5 Розрахунок ринкових індикаторів та показників сегментів ринку 32
2.6 Прогнозування часових рядів ринкових показників 33
2.7 Методика оцінювання довіри до оголошень на основі trust-score 35
2.8 Вибір програмних засобів та бібліотек для реалізації системи 37
2.9 Метрики оцінювання якості моделей та результатів прогнозування 38
3 Розробка системи аналізу та прогнозування ринкових показників 41
3.1 Загальна структура програмної реалізації та конвеєр обробки даних 41

3.2 Реалізація модуля очищення даних 42
3.3 Реалізація модуля формування ознак 44
3.4 Реалізація модуля навчання моделі прогнозування ціни 46
3.5 Реалізація модуля розрахунку ринкових індикаторів 47
3.6 Реалізація модуля оцінювання довіри до оголошень 48
3.7 Реалізація модуля прогнозування ринкових показників 50
3.8 Реалізація інтерактивного дашборду користувача 51
3.9 Тестування програмної системи та перевірка працездатності 52
3.10 Аналіз отриманих результатів та обмеження реалізації 53
Висновки 55
Перелік джерел посилання 57
Додаток А Відомість кваліфікаційної роботи 59

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Градієнтний бустинг – ансамблевий метод машинного навчання, що послідовно навчає слабкі моделі, кожна з яких виправляє помилки попередньої;

Дедублікація – процес виявлення та видалення повторюваних записів у наборі даних;

Конвеєр обробки даних – послідовний набір програмних модулів, де результат кожного кроку є вхідними даними для наступного;

Лагова ознака – значення часового ряду, зміщене на один або кілька попередніх періодів, що використовується як предиктор;

Медіанна ціна – значення ціни, що ділить впорядкований набір на дві рівні частини; менш чутливе до викидів порівняно із середнім;

Ринковий індикатор – числова або категоріальна характеристика, що описує стан певного сегмента ринку;

Частотне кодування – метод перетворення категоріальних ознак, за якого кожне значення замінюється відносною частотою його появи у наборі даних;

ШІ – штучний інтелект;

AI – Artificial Intelligence – штучний інтелект;

API – Application Programming Interface – інтерфейс програмування застосунків;

CatBoost – Categorical Boosting – алгоритм градієнтного бустингу з нативною підтримкою категоріальних ознак;

CSV – Comma-Separated Values – текстовий формат зберігання табличних даних;

CRISP-DM – Cross-Industry Standard Process for Data Mining – міжгалузевий стандартний процес інтелектуального аналізу даних;

JSON – JavaScript Object Notation – текстовий формат обміну даними;

MAE – Mean Absolute Error – середня абсолютна похибка;

MAPE – Mean Absolute Percentage Error – середня абсолютна відсоткова похибка;

ML – Machine Learning – машинне навчання;

NLP – Natural Language Processing – обробка природної мови;

OOF – Out-of-Fold – метод прогнозування, за якого кожен рядок оцінюється моделлю, що не навчалась на ньому;

Overfitting (перенавчання) – стан моделі, за якого вона надмірно підлаштовується під навчальні дані та втрачає здатність до узагальнення;

R^2 – коефіцієнт детермінації – метрика, що показує частку поясненої варіації цільової змінної;

RMSE – Root Mean Squared Error – корінь середньоквадратичної похибки;

SDLC – Software Development Life Cycle – життєвий цикл розробки програмного забезпечення;

SHAP – SHapley Additive exPlanations – метод інтерпретації прогнозів моделей машинного навчання на основі значень Шеплі;

Trust-score – числова оцінка довіри до оголошення, що формується на основі відхилення фактичної ціни від прогнозованої та наявності ризикових ознак.

ВСТУП

Цифрові ринки генерують значні обсяги структурованих даних: оголошення про продаж, цінові показники, характеристики товарів, географічні ознаки, часові мітки публікацій та текстові описи. Обробка таких масивів потребує автоматизованих програмних засобів, оскільки ручний аналіз не забезпечує необхідної повноти та системності. Для порівняння ринкових сегментів, виявлення цінової динаміки або ідентифікації аномальних пропозицій необхідна спеціалізована програмна система.

У межах цієї роботи аналіз ринкових показників здійснюється на прикладі ринку вживаних автомобілів. Вибір обумовлений багатофакторністю ціноутворення у цій предметній області: вартість транспортного засобу формується під впливом року випуску, пробігу, марки, моделі, типу кузова, технічного стану та регіону реалізації. Додатково, у відкритому доступі наявний датасет Craigslist Vehicles, що містить сотні тисяч реальних оголошень і дозволяє перевірити запропоновані підходи на репрезентативних даних [1].

Актуальність теми обумовлена тим, що ціна в оголошенні не завжди відображає об'єктивну ринкову вартість об'єкта. Частина записів містить технічні помилки, дублікати, рекламні фінансові пропозиції або аномально низькі значення. Для ринку вживаних товарів ця проблема є особливо значущою через інформаційну асиметрію між продавцем і покупцем, яка проявляється не лише у прихованості технічного стану, а й у характері подання оголошення. Табличне представлення цінових даних не усуває цієї проблеми, оскільки не дозволяє визначити типовість ціни для сегмента або простежити динаміку ринку в часі. У зв'язку з цим доцільним є поєднання методів підготовки даних, машинного навчання та аналізу часових рядів в єдиному конвеєрі відповідно до методології CRISP-DM [4].

Методологічною основою роботи є підхід CRISP-DM, який передбачає послідовне виконання етапів: розуміння задачі та даних, підготовку набору, побудову моделі, оцінювання її якості та інтерпретацію результатів.

Розроблена система реалізує повний цикл обробки даних – від завантаження вихідного CSV-файлу до інтерактивного аналітичного інтерфейсу. Конвеєрна архітектура забезпечує можливість повторної обробки при надходженні нових даних із збереженням єдиної логіки розрахунків. Об'єктом дослідження є процес аналізу та прогнозування ринкових показників на основі відкритих табличних даних про пропозиції товарів на електронних майданчиках. Предметом дослідження є методи очищення ринкових даних, формування ознак, прогнозування ціни, розрахунку ринкових індикаторів, оцінювання довіри до оголошень та візуалізації результатів.

Метою роботи є розробка модульної програмної системи аналізу та прогнозування ринкових показників із застосуванням інтелектуальних підходів на прикладі ринку вживаних автомобілів. Досягнення поставленої мети передбачає виконання таких завдань: аналіз предметної області та характеристики вхідних даних; підготовку та очищення набору даних; реалізацію моделей прогнозування ціни та ринкових індикаторів; формування системи оцінювання довіри до оголошень; розробку інтерактивного інтерфейсу перегляду результатів.

Практичне значення роботи полягає в тому, що розроблений прототип може слугувати основою для побудови систем ринкової аналітики: оцінювання справедливої вартості транспортного засобу, виявлення підозрілих оголошень та аналізу динаміки медіанної ціни. Науково-практична новизна полягає в інтеграції очищення даних, прогнозування ціни засобами CatBoostRegressor, аналізу часових рядів та оцінювання довіри до оголошень в єдиному конвеєрі обробки даних.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Актуальність аналізу та прогнозування ринкових показників

У межах даної роботи під ринковим показником розуміється числова або категоріальна характеристика, що відображає стан певного сегмента ринку. Прикладами таких показників є медіанна ціна, кількість оголошень, розкид цін, частка підозрілих записів, середній пробіг або прогнозоване значення на наступний період. Окреме оголошення набуває аналітичної цінності лише у контексті порівняння з іншими об'єктами відповідного сегмента.

Практична цінність системи визначається її здатністю відповідати на конкретні запити: чи відповідає ціна ринковому рівню, чи є ознаки маніпулятивного оголошення, як змінюються ціни в певному сегменті. Реалізація таких функцій вимагає поєднання методів підготовки даних, агрегованих ринкових показників, модельного оцінювання та інтерпретації результатів в єдиній програмній системі.

Для ринку вживаних автомобілів ця проблематика є особливо актуальною, оскільки вартість транспортного засобу формується під впливом багатьох факторів одночасно: пробігу, року випуску, технічного стану, регіону та типу продавця. Неоднорідність даних – наявність пропусків, шумових значень і некоректно заповнених полів – унеможливлює отримання достовірної оцінки на основі простих агрегатів.

Додатковим чинником актуальності є динамічність ринкового середовища: оголошення публікуються та видаляються безперервно, що зумовлює швидке старіння статичної аналітики. Конвеєрна архітектура програмної системи забезпечує можливість повторної обробки при надходженні нових даних із збереженням єдиної логіки розрахунків.

У контексті цієї роботи під інтелектуальним підходом розуміється здатність системи виявляти приховані залежності в даних, узагальнювати

інформацію про ринкові сегменти та оцінювати відхилення від очікуваних значень. Модель прогнозування розглядається як складова частина ширшої аналітичної логіки, а не як самостійний кінцевий результат.

Необхідно також розмежовувати поняття аналізу та прогнозування. Аналіз встановлює поточний стан даних: типові цінові рівні, представленість брендів, наявність аномалій. Прогнозування оцінює невідомі або майбутні значення: очікувану ціну конкретного транспортного засобу чи майбутній рівень медіанної ціни. У роботі обидва підходи застосовуються у взаємодоповнювальній формі.

Зокрема, модель прогнозування ціни формує оцінку для конкретного запису, однак сам прогноз не є достатнім для висновку щодо достовірності оголошення. Для цього необхідне зіставлення фактичної ціни з прогнозованою, аналіз текстових сигналів ризику та застосування прозорої системи штрафів. Результатом такого підходу є показник trust-score, що забезпечує інтерпретовану оцінку рівня довіри до оголошення.

З таблиці видно, що система повинна працювати з різними видами інформації. Частина показників обчислюється без моделі, частина потребує прогнозування, а частина виникає вже на стику прогнозу і правил інтерпретації. Саме тому архітектура має бути модульною.

У роботі використовуються п'ять груп ринкових показників:

– цінові показники – медіанна ціна, прогнозована ціна та відхилення від прогнозу – дозволяють оцінити рівень ринку та справедливість окремої пропозиції;

– об'ємні показники – кількість оголошень, частка сегмента та кількість підозрілих записів – відображають активність ринку та наповненість сегментів;

– сегментні показники формуються в розрізі марки, року випуску, типу кузова та регіону і використовуються для порівняння груп товарів між собою;

– часові показники – динаміка медіанної ціни та тренд кількості оголошень – забезпечують оцінку змін ринку та короткострокове прогнозування;

– довірчі показники – trust-score, клас угоди та ознаки фінансової реклами – слугують для попередження про нетипові або ризикові оголошення.

1.2 Характеристика ринку вживаних автомобілів як прикладної області дослідження

Ринок вживаних автомобілів є зручною прикладною областю для задачі прогнозування ціни, оскільки вартість об'єкта формується під впливом багатьох факторів одночасно. На відміну від товарів з однорідними характеристиками, транспортний засіб не може бути описаний єдиним параметром: марка та модель визначають базовий ціновий рівень, рік випуску відображає вік, пробіг – інтенсивність експлуатації, а технічний стан і наявність документів можуть суттєво змінювати вартість.

Водночас цей ринок не є повністю формалізованим. У відкритих оголошеннях поширені як надмірно короткі, так і рекламно орієнтовані описи. Частина продавців зазначає повну вартість транспортного засобу, тоді як інші вказують щомісячний платіж або мінімальний перший внесок замість реальної ціни. Включення таких записів до навчальної вибірки без попередньої фільтрації призводить до навчання моделі на некоректних прикладах.

Датасет Craigslist Vehicles охоплює оголошення з різних регіонів США, тобто в одному наборі поєднані локальні ринки з різною купівельною спроможністю, попитом і складом пропозицій. Це обумовлює необхідність включення географічних ознак до складу предикторів, хоча їх застосування потребує обережності з огляду на ризик надмірної специфікації моделі.

Окремою характеристикою є висока кардинальність ряду полів: кількість унікальних значень для моделей автомобілів, регіонів і населених пунктів є значною. Застосування one-hot encoding для таких полів суттєво збільшує розмірність простору ознак. У програмній реалізації для високо-кардинальних полів застосовано частотне кодування, тоді як низько-кардинальні категорії передаються в CatBoost з нативною обробкою.

Важливою вимогою до системи є не лише точність прогнозу, а й відповідність результатів обмеженням предметної галузі. Транспортний засіб, характеристики якого відповідають вартості в десятки тисяч доларів, не може правдоподібно пропонуватися за кілька сотень доларів без додаткового пояснення. Подібні випадки можуть свідчити про помилку, шахрайську схему, фінансову рекламу або неповне оголошення, і кожен з них потребує окремої позначки для користувача.

Таким чином, ринок вживаних автомобілів використовується у роботі як складна прикладна область, що дозволяє перевірити повний цикл системи: очищення даних, формування ознак, прогнозування ціни, аналіз сегментів, оцінювання довіри та візуалізацію результатів. Запропонований підхід може бути адаптований до інших ринків з аналогічною структурою табличних даних. Особливості ринку систематизовано за п'ятьма групами: багатофакторність ціноутворення, значний рівень шуму в даних, висока кардинальність ряду полів, локальний характер попиту та наявність маніпулятивних оголошень. Сукупність цих особливостей визначає вимоги до архітектури системи та унеможливлює навчання моделі без попереднього етапу очищення даних.

1.3 Особливості табличних ринкових даних та проблеми їх підготовки

Табличні ринкові дані містять рядки, колонки та числові значення, однак зміст цих значень часто залежить від контексту. Нуль у полі ціни може означати технічну помилку, рекламну заглушку або відсутність

інформації. Відсутнє значення у полі стану не обов'язково робить запис непридатним, тоді як відсутність моделі або опису може бути критичною для подальшої обробки.

Підготовка даних у даній задачі охоплює кілька рівнів. На першому усуваються технічні помилки: дублікати, майже порожні колонки та некоректні типи даних. На другому фільтруються економічно неправдоподібні значення – занадто низькі або завищені ціни. На третьому виконується обробка пропусків, оскільки видалення всіх записів з неповними категоріями призводить до надмірної втрати даних.

У програмній реалізації підозрілі записи не видаляються, а зберігаються у файлі flagged_listings.csv. Такі записи виключаються з навчальної вибірки моделі, однак залишаються доступними для аудиту та формування правил оцінювання довіри. Система не приховує проблемні дані, а переводить їх у контрольовану зону.

Якість підготовки даних безпосередньо впливає на якість прогнозування. Якщо навчальна вибірка містить фінансові оголошення з цінами у 100–200 доларів, модель отримує хибні приклади, що призводить до систематичного заниження прогнозів для реальних пропозицій. З цієї причини очищення даних розглядається у роботі не як допоміжний етап, а як складова інтелектуального підходу.

Окремої уваги потребує обробка часових полів. Дати публікації та видалення оголошень використовуються для побудови місячних і тижневих агрегатів, оцінювання кількості активних записів та формування лагових ознак. Некоректна обробка часових полів унеможливлює отримання змістовної ринкової динаміки та перетворює часовий ряд на випадкову послідовність значень.

Найбільш поширені проблеми вхідних даних та методи їх усунення:

– дублікати (повторні оголошення одного автомобіля) – видалення точних копій і репостів за content-key;

– некоректна ціна (значення 0, 1, 100 доларів або аномально високі) – фільтрація за межами PRICE_MIN та PRICE_MAX;

– фінансова реклама («monthly payment», «bad credit», «down payment») позначення finance_signal і винесення у flagged_listings;

– пропуски категорій (порожні поля fuel, drive, paint_color) – заповнення значенням unknown;

– аномальний пробіг (старий автомобіль із майже нульовим пробігом) – контроль порогів odometer та low mileage;

– надмірно порожні колонки (службові поля з майже всіма null) – видалення колонок з часткою пропусків понад заданий поріг.

Після такого етапу дані стають придатними для формування ознак і навчання моделі. Водночас очищення не повинно бути надто агресивним. Якщо видалити всі нетипові випадки, модель втратить здатність працювати з реальними даними, які майже завжди трохи брудні.

1.4 Огляд інтелектуальних підходів до прогнозування ціни та ринкової динаміки

Для прогнозування ціни може бути застосовано різні групи методів: лінійні моделі, дерева рішень, ансамблі, градієнтний бустинг, нейронні мережі. Вибір методу залежить не лише від бажаної точності, а й від типу даних. У даній роботі дані є табличними, містять числові й категоріальні ознаки та характеризуються нелінійними залежностями, що робить ансамблеві методи природним вибором для цієї задачі [3].

Лінійна регресія може використовуватися як базова модель для порівняння. Вона є зрозумілою та обчислювально ефективною, однак її можливості обмежені при наявності складних взаємодій між ознаками. Наприклад, вплив віку автомобіля на ціну суттєво відрізняється для бюджетного седана і преміального електромобіля, що потребує явного задання взаємодій при використанні лінійної моделі.

Дерева рішень ефективніше описують нелінійні залежності, однак є нестабільними: незначна зміна навчальної вибірки може суттєво змінити структуру дерева. Ансамблеві методи усувають цей недолік шляхом усереднення або послідовного виправлення помилок, що зумовило широке застосування Random Forest [5], XGBoost [6], LightGBM і CatBoost у задачах на табличних даних.

CatBoost є особливо придатним для задач з великою кількістю категоріальних ознак, оскільки підтримує їх нативну обробку на основі ordered target statistics [7]. Це дозволяє уникнути ручного розширення простору ознак через one-hot encoding та знижує ризик витоку цільової змінної під час кодування категорій. Додатково, CatBoost надає показники важливості ознак, що дозволяє перевірити відповідність моделі логіці предметної області. Якщо модель спирається переважно на вік автомобіля, пробіг та марку, це підтверджує коректність сформованих ознак.

Прогнозування ринкових індикаторів у часі є задачею іншої природи. Тут об'єктом аналізу є не окреме оголошення, а агреговані значення за місяцями або тижнями. Для таких задач застосовуються лагові ознаки, ковзні середні, сезонні індикатори і трендові компоненти [8], які забезпечують передачу моделі інформації про попередні періоди.

У роботі для прогнозування часових рядів застосовано GradientBoostingRegressor [9]. Ця модель працює з табличними ознаками, не потребує складного налаштування та дозволяє порівнювати результат з простим baseline. Алгоритм градієнтного бустингу ґрунтується на послідовному навчанні слабких моделей, кожна з яких виправляє помилки попередньої.

Оцінювання моделей часових рядів принципово відрізняється від стандартного випадкового розбиття вибірки. Навчання на майбутніх даних з тестуванням на минулих є методологічно некоректним. З цієї причини у роботі застосовується holdout останніх часових періодів, що відповідає реальному сценарію використання системи.

Порівняння груп методів наведено у таблиці 1.3.

Таблиця 1.3 – Порівняння інтелектуальних підходів для задачі роботи

Метод	Переваги	Обмеження для роботи
Лінійна регресія	Простота, швидкість, зрозумілі коефіцієнти	Слабко описує складні нелінійні взаємодії
Дерево рішень	Добре працює з порогами і нелінійністю	Нестабільне без ансамблю
Random Forest	Стійкість, усереднення багатьох дерев	Гірше працює з екстраполяцією трендів і може бути важким
Gradient Boosting	Висока якість для табличних ознак, гнучкість	Потребує контролю перенавчання
CatBoost	Нативна робота з категоріями, ordered boosting	Потребує окремого збереження ознак і контролю параметрів
Нейронні мережі	Можливість складних представлень	Надмірні для прототипу на табличних даних без великого tuning

Для прогнозування ціни у роботі обрано CatBoostRegressor, а для часових рядів ринкових індикаторів – GradientBoostingRegressor. Такий вибір виглядає збалансованим: перша модель краще працює з категоріями оголошень, друга зручно використовує лагові ознаки агрегованого ряду.

1.5 Огляд існуючих програмних рішень для аналізу ринкових даних

Перед розробкою власної системи доцільно розглянути наявні рішення у трьох категоріях: спеціалізовані маркетплейси та агрегатори оголошень про вживані автомобілі, аналітичні платформи для оцінювання ринкової вартості та інструменти машинного навчання для прогнозування ціни.

1.5.1 Маркетплейси та агрегатори оголошень

Для аналізу обрано ринок вживаних автомобілів США з кількох причин. По-перше, це один із найбільших вторинних ринків автомобілів у світі з мільйонами активних оголошень щомісяця. По-друге, у відкритому доступі існує структурований датасет Craigslist Vehicles, що містить сотні тисяч реальних оголошень з достатньою кількістю атрибутів для побудови повноцінної моделі.

Найбільш поширеними платформами для купівлі-продажу вживаних автомобілів у США є Craigslist, CarGurus, AutoTrader та Cars.com. Кожна з них надає базові функції пошуку, фільтрації та сортування оголошень, однак аналітичні можливості платформ суттєво відрізняються.

Craigslist (рисунок 1.1) є найбільшим відкритим майданчиком з мінімальною модерацією. Платформа не надає аналітичних інструментів: користувач отримує список оголошень без оцінки справедливості ціни, без виявлення підозрілих записів і без ринкових агрегатів. Відкритість платформи та наявність публічного датасету Craigslist Vehicles визначили його використання як основи для даної роботи.

CarGurus реалізує власну систему оцінювання угод – IMV (Instant Market Value), яка порівнює ціну оголошення з аналогічними пропозиціями у регіоні та присвоює одну з категорій: «great deal», «good deal», «fair price», «high price» або «overpriced». Методика є закритою: користувач не має доступу до інформації про те, які параметри вплинули на оцінку і яке відхилення фактичної ціни від прогнозованої. Крім того, система орієнтована виключно на ринок США і не є відтворюваною для інших датасетів.

AutoTrader і Cars.com пропонують схожий функціонал: фільтрацію за параметрами та приблизну оцінку вартості на основі агрегованих ринкових даних. Однак методика оцінювання також залишається непрозорою, а

механізм класифікації оголошень за рівнем довіри на цих платформах відсутній.

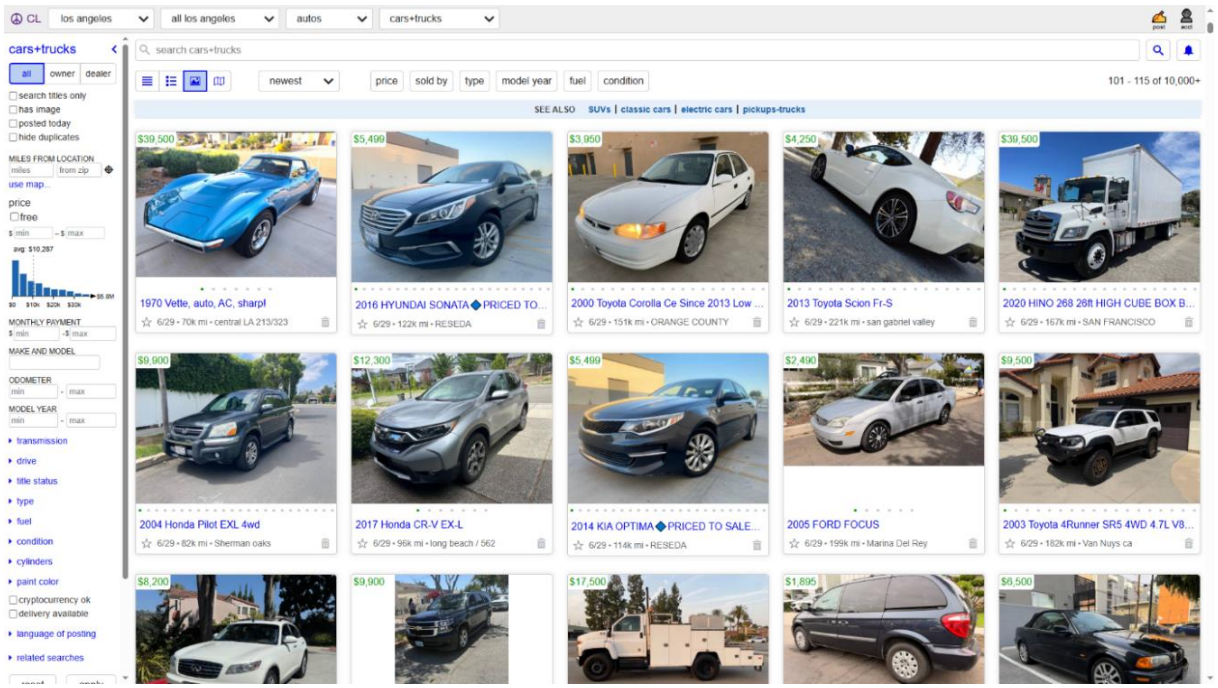

Рисунок 1.1 – Маркетплейс Craigslist

Зазначені обмеження існуючих платформ свідчать про те, що жодне з наявних рішень не забезпечує повного циклу аналізу – від очищення даних до оцінювання довіри до оголошень.

1.5.2 Спеціалізовані сервіси оцінювання вартості автомобілів

Окрему категорію складають сервіси, орієнтовані на оцінювання ринкової вартості транспортних засобів: Kelley Blue Book (KBB), Edmunds та Black Book.

Kelley Blue Book є найвідомішим сервісом оцінювання у США. Він надає три типи вартості: trade-in value, private party value та dealer retail value. Оцінка базується на власній закритій базі даних транзакцій і враховує стан автомобіля, пробіг, комплектацію та регіон. KBB інтегрований у CarGurus

та AutoTrader як додаткове джерело оцінки. Однак методика є повністю закритою, що унеможливлює її відтворення або адаптацію для інших датасетів. Механізм виявлення фінансової реклами або підозрілих оголошень у системі відсутній.

Edmunds використовує власний показник True Market Value (TMV), що відображає середню ціну реальних угод у регіоні. На відміну від KBB, Edmunds частково розкриває методологію: враховуються дані дилерів, аукціонів і приватних продажів. Проте система також є закритою і не надає прогнозу на рівні окремого оголошення з поясненням відхилення від ринкової вартості.

Усі розглянуті сервіси мають спільне обмеження: вони орієнтовані виключно на американський ринок, залежать від власних закритих баз даних і не дозволяють відтворити логіку оцінювання на відкритому датасеті. Саме ця закритість визначає необхідність розробки прозорої та відтворюваної системи, реалізованої у даній роботі.

1.5.3 Методи машинного навчання для прогнозування ціни

У задачах прогнозування ціни на табличних даних застосовуються кілька груп методів, що відрізняються за складністю та здатністю працювати з різнорідними ознаками.

Лінійні моделі є найпростішим підходом і слугують базовою лінією для порівняння результатів. Вони добре інтерпретуються, однак погано описують нелінійні залежності – наприклад, різний вплив пробігу на ціну залежно від марки або класу автомобіля.

Дерева рішень краще справляються з нелінійністю, але є нестабільними: незначна зміна навчальної вибірки може суттєво змінити структуру моделі. З цієї причини на практиці використовуються ансамблеві методи на їх основі. Random Forest усереднює прогнози великої кількості

дерев, підвищуючи стабільність результату, однак поступається градієнтному бустингу за точністю на більшості табличних задач.

Градієнтний бустинг – XGBoost, LightGBM і CatBoost – є поточним стандартом для задач регресії на табличних даних. XGBoost і LightGBM демонструють високу якість прогнозування, однак обидва потребують явного кодування категоріальних ознак. У випадку полів з великою кількістю унікальних значень – модель автомобіля, регіон, штат – це суттєво збільшує розмірність простору ознак або вносить штучний порядок між категоріями.

CatBoost вирізняється нативною підтримкою категоріальних ознак через ordered target statistics, що усуває необхідність їх попереднього кодування без ризику витоку цільової змінної. Для ринку вживаних автомобілів, де поля manufacturer, model та region містять сотні унікальних значень, це є принциповою перевагою. Як видно з рисунку 1.2, CatBoost демонструє найкращі показники MAE, RMSE та MAPE серед розглянутих моделей, що підтверджує доцільність його застосування у даній роботі.

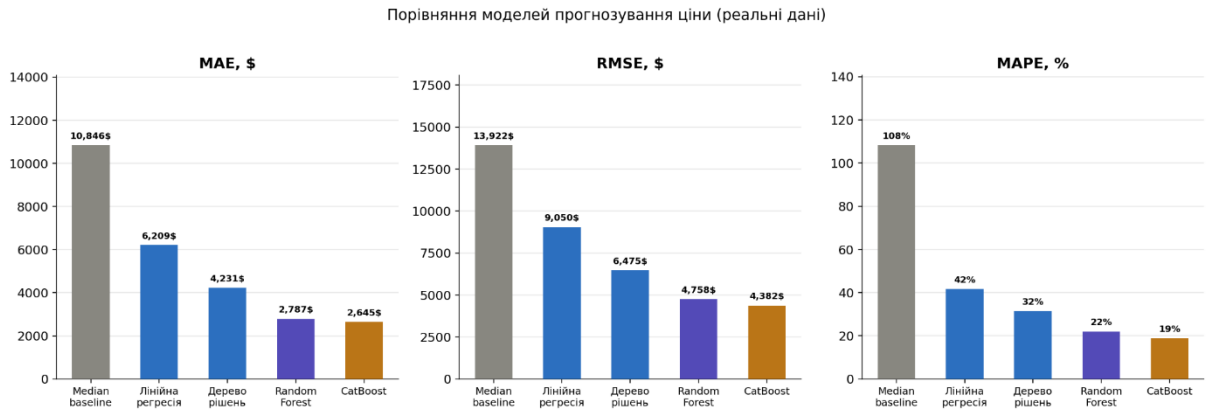

Рисунок 1.2 – Порівняння різних моделей на датасеті Craigslist Vehicles

Нейронні мережі для табличних даних середнього розміру, як правило, поступаються ансамблевим методам при значно більших витратах на підбір архітектури, що робить їх менш доцільними для прототипу.

1.6 Постановка задачі дослідження

На основі аналізу предметної галузі сформульовано задачу розробки системи, яка реалізує повний цикл роботи з ринковими даними. Вхідними даними є CSV-файл з оголошеннями про продаж вживаних автомобілів, а результатом – очищений набір даних, навчена модель прогнозування ціни, ринкові індикатори, прогноз агрегованих показників, оцінка довіри до оголошень та інтерактивний дашборд.

Задача не обмежується побудовою однієї моделі, оскільки окремі компоненти системи не є самодостатніми: навчання моделі на необробленому наборі призводить до нестійких результатів, побудова лише агрегатів не забезпечує оцінки конкретного оголошення, а формування trust-score без інтерфейсу ускладнює інтерпретацію результатів. З цієї причини система реалізована у вигляді взаємопов'язаних модулів із чітко визначеними входами та виходами.

Основними функціональними вимогами є:

– завантаження та очищення вихідних даних;

– формування ознак для моделі;

– навчання моделі прогнозування ціни та збереження артефактів;

– розрахунок ринкових індикаторів;

– прогнозування часових рядів;

– формування trust-score;

– інтерактивний перегляд результатів;

– автоматизоване тестування працездатності.

До нефункціональних вимог належать відтворюваність результатів, винесення параметрів у конфігураційний файл, збереження звітів окремо від артефактів моделі та наявність автоматизованих тестів для контролю коректності змін у коді.

Вхідні дані обмежено такими критеріями: ціни від 500 до 80 000 доларів, рік випуску не раніше 2000, контроль пробігу та наявності ключових полів.

Система вирішує дві прогнозні задачі – оцінювання ринкової вартості окремого транспортного засобу та короткостроковий прогноз агрегованих показників ринку, – а також формує trust-score і клас угоди для виявлення підозрілих оголошень. Результатом є інтерактивний Streamlit-дашборд із фільтрами, таблицями, графіками та поясненнями прогнозів.

2 ОБҐРУНТУВАННЯ МЕТОДІВ ТА ТЕХНОЛОГІЙ РОЗРОБКИ СИСТЕМИ

2.1 Загальний підхід до побудови системи аналізу та прогнозування

Архітектура системи побудована навколо конвеєра обробки даних, в якому кожен модуль має чітко визначені вхідні дані, результат та зону відповідальності. Така структура забезпечує прозорість реалізації: можна простежити, як вихідні оголошення поступово перетворюються на прогнози, ринкові показники та візуалізацію. У разі виникнення помилки її локалізація спрощується завдяки чіткому розмежуванню між етапами.

На відміну від типового дослідницького підходу, де етапи завантаження даних, побудови графіків, формування ознак і збереження моделі часто перемішані в межах одного файлу, у даній роботі кожен етап реалізовано окремим пронумерованим Python-скриптом. Це забезпечує зрозумілий порядок запуску без необхідності додаткової документації.

Конвеєр системи виконується у такій послідовності: очищення вихідних даних, формування ознак, навчання CatBoostRegressor та збереження артефактів, розрахунок ринкових індикаторів, формування trust-score, прогнозування часових рядів і запуск інтерактивного дашборду. Послідовність є невипадковою: кожен наступний етап використовує результати попереднього.

Конвеєрна архітектура також спрощує повторний запуск системи (рисунок 2.1). Зміна порогу очищення потребує повторного виконання pipeline з першого кроку, тоді як зміна параметрів дашборду не вимагає перенавчання моделі. Така організація коду відрізняє відтворювану програмну реалізацію від одноразового експерименту та є принциповою вимогою до аналітичних систем.

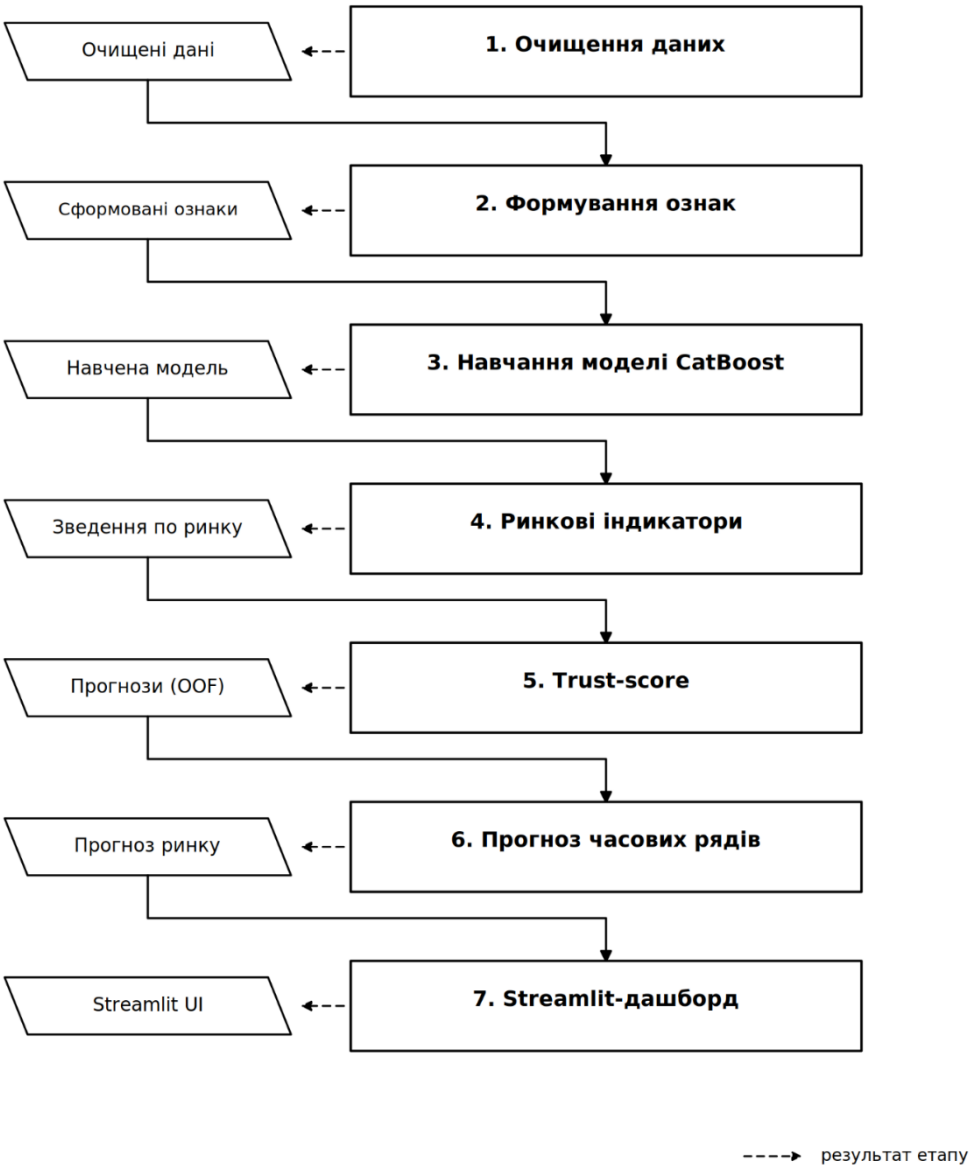

Рисунок 2.1 – Блок-схема обробки даних

Перевага такої схеми полягає у прозорості. Кожен файл має зрозуміле місце в системі, а користувач може запускати кроки послідовно або перевіряти окремий модуль.

2.2 Методи очищення, фільтрації та контролю якості даних

Очищення даних у роботі реалізовано не як механічне видалення нетипових записів, а як набір правил, що враховують природу ринку. Наприклад, оголошення з ціною 400 доларів у більшості випадків є

нерелевантним, фінансовою рекламою або записом з неповною ціною. Водночас низька ціна сама по собі не є однозначною ознакою шахрайства, тому в системі застосовано комбінацію цінових порогів і текстових сигналів.

Порогові значення ціни встановлено в діапазоні від 500 до 80 000 доларів. Ці межі відповідають задачі аналізу масового споживчого сегмента і не претендують на універсальність. Колекційні автомобілі, екзотичні моделі та лоти преміального класу виключені з розгляду, оскільки їх ціноутворення підпорядковується іншій логіці і може вносити суперечливі приклади до навчальної вибірки.

Рік випуску обмежено нижньою межею 2000 року. Старіші автомобілі часто належать до сегменту класики або реставраційних проєктів, вартість яких визначається факторами, що не представлені у табличних даних. Включення таких записів до навчальної вибірки може призводити до суперечливих прикладів для моделі.

Окрему увагу приділено виявленню фінансових сигналів у текстових полях. У частині оголошень зазначена ціна відповідає не повній вартості автомобіля, а щомісячному платежу або мінімальному внеску. Для їх виявлення аналізуються характерні лексичні одиниці: «financing», «bad credit», «down payment», «monthly payment». Виявлені ознаки використовуються як на етапі очищення, так і під час подальшого розрахунку trust-score.

Контроль якості даних також передбачає обробку пропусків. Для категоріальних полів застосовано заповнення значенням unknown, що дозволяє зберегти записи з відсутніми другорядними атрибутами – кольором або типом приводу – без втрати інформації. Для ключових полів, таких як ціна, рік випуску, модель та опис, застосовуються суворіші вимоги до повноти даних.

Орієнтовні пороги очищення даних наведені у таблиці 2.2.

Таблиця 2.2 – Орієнтовні пороги очищення даних

Порог	Орієнтовне значення
Мінімальна ціна	500
Максимальна ціна	80000
Мінімальний рік	2000
Максимальний рік	2025
Максимальний пробіг	300000
Мінімальний пробіг	10

Пороги винесені у конфігураційний файл. Це зроблено спеціально, щоб не шукати «магічні числа» у різних частинах коду. Якщо ринок або датасет зміниться, ці параметри можна буде скоригувати централізовано.

2.3 Формування ознак для прогнозування ринкової ціни

Формування ознак є проміжним етапом між очищенням даних і навчанням моделі. На цьому етапі вихідні поля перетворюються на змінні, що точніше описують економічний зміст оголошення. Наприклад, рік випуску є менш інформативним предиктором, ніж вік автомобіля, а абсолютний пробіг поступається показнику пробігу на рік як характеристиці інтенсивності експлуатації.

Для числових ознак у системі формуються: age, miles_per_year, description_length, word_count, caps_ratio, listing_duration_days та географічні координати. Частина цих ознак безпосередньо характеризує транспортний засіб, частина – стиль оголошення. Стилістичні ознаки не визначають ціну так само суттєво, як технічні характеристики, однак можуть бути корисними для виявлення дилерської або спамної поведінки.

Для категоріальних ознак застосовано два підходи. Поля з невеликою кількістю унікальних значень – fuel, transmission, title_status, type, drive – передаються в CatBoost як категоріальні без додаткового кодування. Поля з

високою кардинальністю – manufacturer, model, region, state – додатково отримують частотне кодування, що відображає відносну частоту появи кожного значення в датасеті.

Частотне кодування не є точним представленням ринкової популярності, однак є практично корисним для прототипу. Висока частота певної моделі може свідчити про її належність до масового сегмента, тоді як рідкісні значення, як правило, характеризуються більшою невизначеністю прогнозу. Словники частот зберігаються як артефакти, що забезпечує узгодженість перетворень між етапом навчання та інтерфейсом системи.

Окремою ознакою є luxury_brand – бінарний індикатор належності марки до преміального сегмента. Ця ознака не замінює категоріальне поле марки, однак надає моделі додатковий сигнал у випадках, коли конкретна преміальна модель рідко представлена в датасеті і її категоріальне кодування є розрідженим.

Групи ознак наведено у таблиці 2.3.

Таблиця 2.3 – Групи ознак для моделі прогнозування ціни

Група	Ознаки	Пояснення
Базові числові	year, age, odometer, miles_per_year	Описують вік і фізичне використання автомобіля
Частотні	manufacturer_freq, model_freq, region_freq, state_freq	Передають поширеність значень у наборі даних
Текстові сигнали	finance_signal, has_dealer_terms, description_length, word_count, caps_ratio	Описують характер оголошення і можливі ризики
Категоріальні	condition, fuel, title_status, transmission, drive, type, size, paint_color, cylinders	Передають властивості авто, які важко коректно описати числами

Продовження таблиці 2.3

Група	Ознаки	Пояснення
Географічні	lat, long	Наближено враховують локальний ринок
Сегментні прапори	luxury_brand	Додають простий сигнал про преміальність бренду

Таке групування також спрощує подальше пояснення моделі. Коли користувач бачить, що вартість залежить не від одного поля, а від набору ознак, прогноз виглядає більш зрозумілим.

2.4 Обґрунтування використання CatBoostRegressor для прогнозування ціни автомобіля

Перед вибором моделі визначено підхід до оцінювання її помилок та інтерпретації результатів. Для регресійних задач використання єдиної метрики є недостатнім, тому в роботі враховано обмеження різних показників точності. Прогноз розглядається не як остаточне рішення, а як інструмент підтримки аналізу з урахуванням можливих ризиків.

Основною моделлю прогнозування ціни обрано CatBoostRegressor. Вибір обумовлений природою даних: набір містить значну кількість категоріальних ознак, а цільова змінна має правосторонньо скошений розподіл. CatBoost ефективно працює з табличними даними такого типу [7] і дозволяє передавати категоріальні поля без попереднього one-hot encoding.

Модель навчається не на вихідних значеннях ціни, а на log1p(price). Це зменшує вплив дорогих автомобілів на функцію втрат і забезпечує більш збалансований розподіл помилок. Після отримання прогнозу значення повертається у доларову шкалу через операцію expm1, що забезпечує інтерпретовані результати для користувача.

Навчальна вибірка розбивається на три частини: train, validation і test у співвідношенні 70/15/15. Валідаційна частина використовується для early

stopping, тестова – виключно для фінального оцінювання. Такий підхід унеможливлює оцінювання моделі на даних, що брали участь у навчанні, і забезпечує достовірність отриманих метрик.

CatBoost надає показники важливості ознак, що є корисним як для аналізу, так і для контролю якості реалізації. Якщо модель активно використовує вік автомобіля, пробіг і марку, це відповідає логіці предметної області. Домінування нерелевантних ознак, наприклад довжини опису, є сигналом можливого витоку даних або артефакту формування ознак.

Разом із навченою моделлю зберігаються метадані: feature_columns, categorical_features, numeric_features, frequency_maps та preprocessing_config. Збереження артефактів є обов'язковою умовою стабільної роботи системи, оскільки забезпечує однаковий формат вхідних даних на етапах навчання та формування прогнозів у дашборді.

Вибір CatBoost не розглядається як універсальне рішення для будь-якої задачі. Для інших наборів даних або інших типів прогнозування можуть бути доречнішими альтернативні підходи.

Перевага CatBoost у даній роботі визначається його практичною відповідністю задачі: ефективна робота з табличними даними, нативна обробка категоріальних ознак та зрозумілі результати для інтеграції в інтерфейс системи.

2.5 Розрахунок ринкових індикаторів та показників сегментів ринку

Ринкові індикатори потрібні для того, щоб перейти від окремого прогнозу до загальної картини. Користувач може захотіти не тільки оцінити одну машину, а й зрозуміти, як поводиться сегмент: які марки дорожчі, як ціна залежить від віку, де більше оголошень, який розкид цін у різних типах кузова.

Сегментні показники також допомагають перевірити модель: якщо модель прогнозує високі ціни для новіших автомобілів і преміальних

брендів, а ринкові агрегати показують те саме, результат виглядає узгодженим. Якщо ж модель і агрегати суперечать одне одному, це привід переглянути очищення або ознаки.

У системі розраховуються такі ринкові індикатори:

– median_price – медіанна ціна на рівні загального ринку, окремого бренду, типу кузова або року випуску. Використовується як стійка оцінка типового цінового рівня, менш чутлива до викидів порівняно із середнім;

– listing_count – кількість оголошень за період або сегмент. Дозволяє оцінити активність ринку та наповненість окремих сегментів;

– price_cv – коефіцієнт варіації ціни за період або сегмент. Низьке значення свідчить про стабільний ринок, високе – про значний розкид і неоднорідність пропозицій;

– depreciation_curve – крива знецінення за віком автомобіля. Якщо автомобіль певної марки втрачає вартість швидше, це видно не за одним оголошенням, а за агрегованою картиною;

– depreciation_by_brand – знецінення в розрізі бренду та віку. Дозволяє порівняти швидкість втрати вартості між різними марками автомобілів;

– segment_summary – агреговані показники за маркою, типом кузова та регіоном. Використовується для пошуку відмінностей між групами ринку та порівняльного аналізу сегментів.

Індикатори не замінюють модель прогнозування ціни, але роблять систему більш повною. Вони дають контекст, без якого прогноз окремого рядка сприймався б відірвано від ринку.

2.6 Прогнозування часових рядів ринкових показників

Прогнозування часових рядів у роботі виконується для агрегованих показників і є задачею, що принципово відрізняється від прогнозування ціни окремого оголошення. Одиницею аналізу тут є не окремий запис, а часовий період – місяць або тиждень.

Для кожного періоду обчислюються значення median_price, listing_count і price_cv, після чого модель прогнозує їх майбутні значення. Горизонт прогнозування у прототипі є короткостроковим і охоплює кілька наступних періодів.

Підготовка часових ознак включає лагові значення ковзні середні, ковзні стандартні відхилення, порядковий номер періоду та номер місяця [8]. Лагові ознаки забезпечують модель інформацією про попередні значення ряду. Ковзне середнє згладжує випадкові коливання. Номер місяця дозволяє частково врахувати сезонні закономірності за їх наявності.

Для оцінювання якості прогнозу використано holdout останніх часових періодів як тестову вибірку. Такий підхід є простішим за повну walk-forward перевірку, однак методологічно коректнішим за випадкове перемішування.

У задачах з часовими рядами порядок спостережень є принциповим: навчання моделі на майбутніх даних з тестуванням на минулих призводить до завищених метрик. З цієї причини навчальна вибірка охоплює більш ранні періоди, а тестова – останні.

Коректна побудова часового ряду також залежить від якості вхідних даних. Нерівномірний розподіл оголошень за часом, наявність пропусків у певних періодах або різка зміна обсягу даних можуть суттєво вплинути на стабільність прогнозу. Тому перед навчанням моделі необхідно перевірити часовий ряд на наявність таких аномалій та застосувати відповідні методи їх усунення.

Отриманий прогноз порівнюється з baseline, яким може слугувати просте ковзне середнє або значення попереднього періоду. Якщо складніша модель не перевищує baseline за якістю прогнозу, доцільність її застосування потребує перегляду. Такий підхід забезпечує об'єктивне оцінювання практичної цінності моделі відносно її складності.

Ознаки для прогнозування часових рядів наведено у таблиці 2.6.

Таблиця 2.6 – Ознаки для прогнозування часових рядів ринкових показників

Ознака	Зміст	Роль у прогнозі
lag_1, lag_2, lag_3	Значення показника у попередніх періодах	Передають короткострокову пам’ять ряду
rolling_mean	Ковзне середнє	Згладжує випадкові стрибки
rolling_std	Ковзне стандартне відхилення	Показує нестабільність останніх періодів
time_idx	Порядковий номер періоду	Додає інформацію про тренд
month	Номер місяця	Наближено враховує сезонність
target_name	median_price, listing_count або price_cv	Визначає, який показник прогнозується

Такий набір ознак не є надмірним. Він достатній для прототипу і зрозумілий під час пояснення результатів. У майбутньому його можна розширити зовнішніми економічними факторами або сезонними індексами.

2.7 Методика оцінювання довіри до оголошень на основі trust-score

Trust-score у роботі призначений для розмежування справді вигідних пропозицій і неправдоподібних оголошень. Відхилення фактичної ціни від прогнозованої на 15–20% може відповідати реальній ринковій вигоді, тоді як відхилення на 90% у поєднанні з фінансовою лексикою в описі свідчить про інший характер оголошення і не може інтерпретуватися як надвигідна пропозиція.

Основою розрахунку trust-score є out-of-fold прогноз: для кожного запису прогноз формується моделлю, що не навчалася на цьому записі.

Такий підхід забезпечує коректність залишку actual minus predicted для всієї вибірки, а не лише для тестової частини. Прогнозування навчальних записів тією самою моделлю призводить до оптимістично занижених відхилень і спотворює оцінку довіри.

Після отримання прогнозу розраховується відносне відхилення pct_vs_predicted, на основі якого застосовуються прозорі штрафи: за аномально низьку ціну, наявність фінансової лексики в описі, підозріло великий обсяг тексту та статус title_status, відмінний від clean. Результуюче значення обмежується діапазоном від 0 до 100, де нижчий показник відповідає більшій кількості підстав для ручної перевірки оголошення.

Важливо зазначити, що trust-score не є юридичною або технічною експертизою транспортного засобу. Система не перевіряє VIN-код, історію аварій або реальний технічний стан. Функція показника є вужчою: виявлення оголошень, ціна та текст яких є нетиповими відносно загального масиву даних. Для користувача це дозволяє скоротити кількість явно ризикових варіантів без проведення повної експертизи кожного запису.

Класифікація оголошень за trust-score базується на відсотковому відхиленні фактичної ціни від прогнозованої:

Підозріла ціна – якщо ціна нижча за прогноз більш ніж на 50% або trust-score опускається нижче 40. Свідчить про занадто низьку ціну або наявність сильних тривожних сигналів.

Вигідна угода – відхилення від -50% до -12% за достатнього trust-score. Ціна нижча за прогноз, але залишається правдоподібною.

Ринкова ціна – відхилення від -12% до +15%. Оголошення близьке до прогнозованої ринкової вартості.

Переоцінено – ціна перевищує прогноз більш ніж на 15%. Пропозиція помітно вища за ринковий рівень.

Потребує перегляду – нестача критичних полів або нестандартний випадок. Результат не слід трактувати автоматично без додаткової перевірки.

Саме прозорість правил робить trust-score придатним для пояснювальної записки. Користувач може побачити не лише число, а й причину штрафу.

2.8 Вибір програмних засобів та бібліотек для реалізації системи

Основною мовою реалізації обрано Python, що є природним вибором для задач, які поєднують обробку даних, машинне навчання та створення інтерактивного інтерфейсу. Мова має зрілу екосистему бібліотек, а модульна структура скриптів дозволяє виконувати окремі етапи pipeline незалежно без повного перезапуску системи.

Для роботи з табличними даними використовуються pandas [12] і NumPy [13]: перша забезпечує очищення, групування та агрегацію даних, друга – числові перетворення та підготовку масивів для моделей. Для машинного навчання застосовується scikit-learn [14], що надає інструменти розбиття вибірки, обчислення метрик та реалізацію GradientBoostingRegressor, і CatBoost як основна модель прогнозування ціни.

Для побудови інтерактивного інтерфейсу обрано Streamlit [15], який дозволяє створити аналітичний дашборд засобами Python-коду без розробки окремої frontend-частини. Візуалізація реалізується через Plotly [16], тестування – засобами pytest [17], збереження артефактів моделі – через joblib [18] і JSON-файли.

Зведення технологій подано у таблиці 2.8.

Таблиця 2.8 – Програмні засоби та їх роль у системі

Засіб	Використання	Причина вибору
Python	Основна мова реалізації	Зручність для data science та модульних скриптів

Продовження таблиці 2.8

Засіб	Використання	Причина вибору
pandas	Очищення, агрегації, CSV	Зручна робота з табличними даними
NumPy	Числові операції	Швидкі масиви та математичні перетворення
scikit-learn	Метрики, розбиття, GradientBoostingRegressor	Стандартні інструменти ML
CatBoost	Прогнозування ціни	Нативна обробка категорій
Streamlit	Дашборд користувача	Швидке створення інтерактивного інтерфейсу
Plotly	Графіки	Інтерактивні візуалізації

Обраний стек не є надмірним. У ньому немає важкого веб-фреймворку чи складної інфраструктури, але є все необхідне для повного прототипу аналітичної системи.

2.9 Метрики оцінювання якості моделей та результатів прогнозування

Оцінювання якості моделі має забезпечувати інтерпретовані результати незалежно від технічної підготовки користувача. З цієї причини метрики прогнозування ціни подаються у доларовій шкалі. MAE відображає середню абсолютну помилку, RMSE сильніше штрафує великі відхилення, MAPE надає відносну помилку у відсотках, а R^2 показує частку поясненої варіації цільової змінної.

Кожна з перелічених метрик має власні обмеження. MAPE є нестабільною при малих значеннях цільової змінної, RMSE чутлива до викидів, а R^2 не завжди є інтуїтивно зрозумілим для кінцевого користувача. З цієї причини у роботі метрики розглядаються у сукупності, без виділення єдиного головного показника. Такий підхід відповідає загальноприйнятій

практиці оцінювання моделей [9], де якість визначається у контексті конкретної задачі.

Для моделей часових рядів обов'язковим є порівняння з baseline. Якщо прогноз медіанної ціни не перевищує за якістю просте ковзне середнє, застосування складнішої моделі не є обґрунтованим. Для trust-score окремо перевіряється коректність логіки класифікації: зокрема, чи потрапляють явно неправдоподібні оголошення з аномально низькою ціною до класу «підозріла ціна».

Важливим аспектом є також відповідальне використання результатів прогнозування. Модельна оцінка ціни не є гарантією реальної угоди – вона відображає очікуване значення на основі доступних полів датасету. Для прийняття реального рішення про придбання транспортного засобу необхідна додаткова перевірка, що узгоджується з принципами управління ризиками при застосуванні аналітичних систем.

Основні метрики наведено нижче у таблиці 2.9).

Таблиця 2.9 – Метрики оцінювання якості моделей

Метрика	Формула або зміст	Що показує
MAE	Середнє абсолютне відхилення	Типова помилка у доларах
RMSE	Корінь середньоквадратичної помилки	Чутливість до великих помилок
MAPE	Середня відносна помилка у відсотках	Зручне порівняння між ціновими рівнями
R^2	Коефіцієнт детермінації	Частка поясненої варіації
Baseline comparison	Порівняння з простим прогнозом	Перевірка, чи має модель практичний сенс
Trust classification check	Аналіз класів угод	Перевірка поведінки правил довіри

Метрики є не фінальною метою, а способом контролю. Якщо модель має хороші числа, але погано поводиться на підозрілих оголошеннях, її не можна вважати достатньо якісною для цієї роботи.

Додатково передбачається ручна перевірка граничних випадків: оголошень з аномально низькою ціною, записів з фінансовою лексикою та автомобілів з нетиповим співвідношенням віку і пробігу. Такі випадки є найбільш показовими для оцінювання практичної цінності системи, оскільки саме вони найчастіше вводять користувача в оману. Якщо система коректно класифікує подібні записи як підозрілі, це свідчить про її відповідність поставленій меті.

3 РОЗРОБКА СИСТЕМИ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ РИНКОВИХ ПОКАЗНИКІВ

3.1 Загальна структура програмної реалізації та конвеєр обробки даних

Програмна реалізація оформлена як набір окремих модулів, імена файлів яких починаються з порядкового номера кроку. Це забезпечує зрозумілий порядок запуску без необхідності додаткової документації: користувач бачить, з якого файлу починати і який модуль відповідає за певний етап обробки.

Окрім основних модулів, проєкт містить config.py, requirements.txt, директорії reports і tests. Конфігураційний файл централізовано зберігає пороги очищення, групи ознак, шляхи до артефактів і параметри trust-score, що унеможливлює дублювання значень у різних частинах коду. Зміна будь-якого параметра, наприклад PRICE_MAX, автоматично застосовується до всіх модулів системи.

У директорії reports зберігаються текстові звіти: model_card, trust_score_rules, data_quality_report та інші. Ці документи фіксують логіку прийнятих рішень, застосовані правила та обмеження результатів. Наявність звітів забезпечує прозорість системи і дозволяє відтворити хід обробки без аналізу вихідного коду.

Програмна реалізація складається з семи функціональних модулів, що утворюють послідовний конвеєр обробки даних. Кожен модуль має чітко визначені вхідні дані та результат, що забезпечує незалежність етапів і можливість окремого запуску кожного з них. Склад модулів відповідає структурі роботи і охоплює всі етапи від очищення вихідних даних до інтерактивного інтерфейсу користувача.

Склад модулів системи:

– модуль очищення даних – обробляє сирий датасет, видаляє дублікати, фільтрує некоректні записи та виявляє фінансові оголошення;

– модуль формування ознак – створює числові, частотні та текстові ознаки на основі очищених даних;

– модуль навчання моделі ціни – навчає CatBoostRegressor на логарифмованій цільовій змінній та зберігає артефакти;

– модуль ринкових індикаторів – розраховує агреговані показники ринку, криві знецінення та сегментну аналітику;

– модуль оцінювання довіри – формує out-of-fold прогнози та розраховує trust-score для кожного оголошення;

– модуль прогнозування часових рядів – будує часові ряди агрегованих показників і прогнозує їх на майбутні періоди;

– модуль інтерактивного дашборду – об'єднує результати всіх модулів в єдиний інтерфейс для користувача.

Склад модулів відповідає плану роботи. Кожний етап можна описати, запустити і перевірити окремо, що спрощує захист програмної реалізації.

3.2 Реалізація модуля очищення даних

Модуль очищення є першим кроком конвеєру. Він зчитує вихідний CSV-файл, перевіряє типи даних, видаляє дублікати, фільтрує некоректні ціни, обробляє пропуски і формує два результати: cleaned_data.csv та flagged_listings.csv. Збереження підозрілих записів у окремому файлі є принциповим рішенням – такі записи не зникають безслідно, а залишаються доступними для аудиту.

У модулі реалізовано виявлення фінансової лексики в текстових полях оголошення. Якщо опис містить характерні слова про кредитування або щомісячні платежі, а зазначена ціна є аномально низькою, запис

позначається як підозрілий. Такий підхід є практичним: він простий у реалізації, швидкий у виконанні та зрозумілий для перевірки.

Дедублікація виконується не лише за повним збігом рядків. Для ринку оголошень про автомобілі характерні репости, у яких змінено лише ідентифікатор або дату публікації, тоді як сам об'єкт залишається тим самим. З групи дублікатів зберігається найновіше оголошення. Логіка роботи модуля очищення даних представлена на рисунку 3.2.

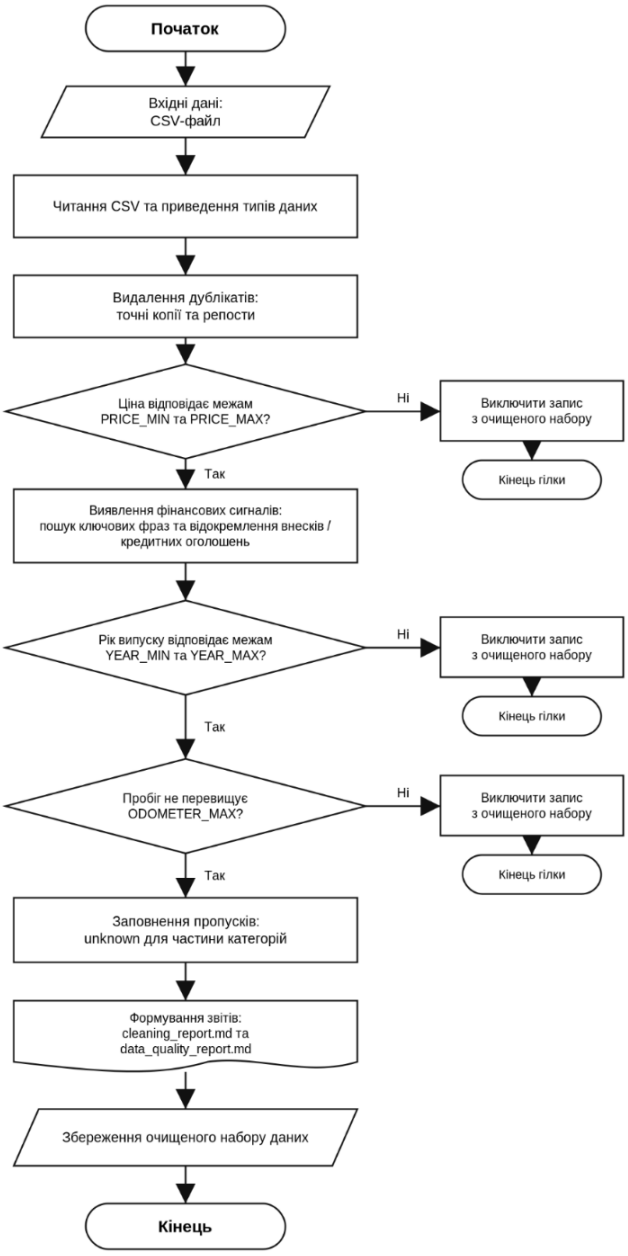

Рисунок 3.2 – Модуль очищення даних

Після цього модуль передає у наступний крок уже контрольований набір. Це не означає, що дані ідеальні, але їхня якість достатня для формування ознак і навчання моделі.

3.3 Реалізація модуля формування ознак

Формування ознак є проміжним етапом між очищенням даних і навчанням моделі. На цьому етапі вихідні поля перетворюються на більш інформативні змінні, що краще описують стан транспортного засобу, інтенсивність його використання та особливості оголошення. Завдяки цьому модель отримує не просто вихідні значення датасету, а узагальнені характеристики з вищим кореляційним зв'язком із цільовою змінною.

Для числових ознак доцільно сформувати такі показники: вік автомобіля (age), пробіг на рік (miles_per_year), довжина опису, кількість слів та частка великих літер. Окремо передбачається включення географічних координат як наближеного відображення локального ринку. Частина цих ознак безпосередньо характеризує транспортний засіб, частина – стиль оголошення, що може бути корисним для виявлення дилерської або спамної поведінки.

При обчисленні похідних числових ознак необхідно передбачити контроль ділення на нуль. Зокрема, для автомобілів з віком 0 років обчислення miles_per_year як odometer / 0 є некоректним, тому планується застосування безпечної логіки обчислення. Подібні технічні деталі суттєво впливають на стабільність роботи системи і мають бути враховані на етапі реалізації.

Для категоріальних ознак передбачається застосування двох підходів залежно від кардинальності поля. Поля з невеликою кількістю унікальних значень – fuel, transmission, title_status, type, drive – плануються до передачі в CatBoost як категоріальні без додаткового кодування. Для полів з високою кардинальністю – manufacturer, model, region, state – доцільно застосувати

частотне кодування, що відображає відносну частоту появи кожного значення в датасеті.

Для текстових полів передбачається формування компактних числових сигналів замість повноцінного векторного представлення: довжина опису, кількість слів, частка великих літер, наявність фінансової або дилерської лексики. Для прототипу, орієнтованого на табличні характеристики автомобіля, такого представлення є достатньо, оскільки основна модель спирається на структуровані поля датасету.

Додатково планується введення бінарної ознаки luxury_brand, що позначає належність марки до преміального сегмента. Ця ознака не замінює категоріальне поле марки, однак надає моделі додатковий сигнал у випадках, коли конкретна преміальна модель рідко представлена в датасеті.

Орієнтовні перетворення наведені на рисунку 3.3.

```
def add_age_features(df):
    df["age"] = (posting_year - df["year"]).clip(lower=0)

def add_mileage_features(df):
    df["miles_per_year"] = np.where(
        age > 0, df["odometer"] / age, df["odometer"]
    )

def add_price_target(df):
    df["price_log"] = np.log1p(df["price"])

def add_frequency_encoding(df):
    for col in FREQ_ENCODE_COLUMNS:
        freq = df[col].value_counts(normalize=True)
        df[f"{col}_freq"] = df[col].map(freq).fillna(0.0)

def add_brand_features(df):
    df["luxury_brand"] = (
        df["manufacturer"].str.lower().isin(LUXURY_BRANDS).astype(int)
    )

def add_text_features(df):
    df["description_length"] = desc.str.len()
    df["word_count"]  = desc.str.split().apply(len)
    df["caps_ratio"]  = (caps / letters.replace(0, np.nan)).fillna(0.0)
    df["has_dealer_terms"] = desc.str.contains(
        DEALER_PATTERN, case=False, regex=True
    ).astype(int)
```

Рисунок 3.3 – Основні перетворення даних

У результаті сирий набір перетворюється на таблицю, у якій кожна колонка має певну роль. Це значно зменшує хаотичність даних перед навчанням.

3.4 Реалізація модуля навчання моделі прогнозування ціни

Модуль читає features_engineered.csv і формує три вибірки: 70% – навчальна, 15% – валідаційна, 15% – тестова. Валідаційна вибірка використовується для early stopping – навчання зупиняється якщо протягом 100 ітерацій якість на валідації не покращується. Тестова вибірка не бере участі у навчанні і використовується виключно для фінальної оцінки.

Цільова змінна подається у логарифмованому вигляді – log1p(price). Це стабілізує розподіл: без трансформації дорогі автомобілі непропорційно впливають на функцію втрат і модель гірше працює на масовому сегменті. Після прогнозу значення повертається у доларову шкалу через expm1.

Категоріальні ознаки (condition, fuel, transmission, drive та інші) передаються в CatBoost напряму через Pool з явним зазначенням їх індексів – без попереднього кодування. Числові ознаки (age, odometer, miles_per_year, *_freq) подаються стандартно.

Метрики MAE, RMSE, MAPE і R^2 обчислюються на тестовій вибірці у доларах і порівнюються з baseline – медіанною ціною навчальної вибірки. За результатами CatBoost знизив MAE з $10,846 до $2,645, MAPE – до 18.94% при $R^2 = 0.897$, що підтверджує доцільність використання моделі відносно простої оцінки.

Разом із навченою моделлю зберігаються метадані необхідні для відтворення прогнозу в дашборді: списки ознак, їх порядок і словники частотного кодування. Додатково формується feature_importances.csv який показує що модель спирається на економічно логічні фактори – вік автомобіля, пробіг та характеристики бренду.

3.5 Реалізація модуля розрахунку ринкових індикаторів

Модуль ринкових індикаторів працює з очищеними даними, а не з features_engineered.csv, оскільки агреговані показники мають описувати ринок у вихідних полях: ціна, бренд, модель, тип кузова, рік випуску та регіон. Модуль групує записи і формує таблиці, що використовуються у звітах та інтерактивному дашборді.

Одним із ключових результатів є крива знецінення, що будується за віком автомобіля і відображає динаміку медіанної ціни з роками експлуатації. Цей показник дозволяє порівнювати швидкість втрати вартості між різними марками та сегментами ринку.

Крива знецінення також слугує додатковим інструментом перевірки адекватності моделі прогнозування ціни: якщо прогнозовані значення для автомобілів різного віку відповідають загальній тенденції знецінення, це підтверджує коректність навченої моделі.

Додатково розраховуються сегментні показники: медіанна ціна, кількість оголошень та коефіцієнт варіації в розрізі брендів, типів кузова та регіонів. Невідповідність агрегатів очікуваним значенням може свідчити про наявність залишкового шуму в даних і є підставою для перегляду правил очищення.

Модуль також генерує текстовий звіт market_summary.md, який доступний для перегляду без запуску дашборду. Звіт фіксує ключові показники поточного запуску pipeline і дозволяє порівнювати результати між запусками при зміні правил очищення або складу вхідних даних. Таким чином, модуль ринкових індикаторів виконує не лише аналітичну функцію, а й роль проміжного контролю якості системи.

Сценарії використання ринкових індикаторів наведено у таблиці 3.5. Завдяки цьому користувач бачить не лише прогноз моделі, а й більш широкий ринковий контекст.

Таблиця 3.5 – Використання ринкових індикаторів у системі

Індикатор	Де використовується	Практичний зміст
market_summary	Звіт і дашборд	Короткий опис стану ринку
depreciation_overall	Аналітичний розділ дашборду	Загальна крива знецінення
depreciation_by_brand	Порівняння брендів	Аналіз різної швидкості втрати вартості
market_trend_weekly	Часова аналітика	Перегляд динаміки за періодами
indicators_by_segment	Фільтри дашборду	Порівняння марок, типів, років і регіонів

3.6 Реалізація модуля оцінювання довіри до оголошень

Модуль trust-score запускається після формування ознак і використовує CatBoost у режимі out-of-fold. Для кожного запису формується прогноз ціни моделлю, що не навчалася на цьому записі. На основі отриманого прогнозу розраховується залишок, відсоткове відхилення та числовий показник довіри.

Реалізація модуля свідомо побудована на прозорих правилах. Замість складної класифікаційної моделі застосовано зрозумілу систему штрафів, що дозволяє пояснити причину низького балу для конкретного оголошення. Результат у вигляді числового значення без пояснення є недостатнім для практичного використання системи.

Окремим випадком є оголошення з аномально низькою ціною. До введення trust-score такі записи могли інтерпретуватися як вигідні пропозиції. Після застосування модуля вони отримують клас «підозріла ціна», якщо відхилення від прогнозу перевищує допустимий поріг або

присутній фінансовий сигнал у тексті. Значна знижка без обґрунтування є ознакою проблемного оголошення, а не реальної ринкової вигоди.

Модуль формує два вихідні файли: predictions_oof.csv, що містить усі записи з прогнозом і класом угоди, та flagged_implausible.csv – лише підозрілі випадки. Такий поділ забезпечує зручність використання результатів у дашборді: загальна картина відображається окремо від списку записів, що потребують аудиту.

Вихідний файл predictions_oof.csv містить такі поля:

– actual_price – реальна ціна оголошення. Використовується для порівняння з прогнозованим значенням;

– predicted_price – прогнозована модельна ціна, отримана через out-of-fold передбачення CatBoost;

– residual – абсолютне відхилення: різниця між фактичною і прогнозованою ціною (actual − predicted);

– pct_vs_predicted – відсоткове відхилення від прогнозу. Є основою для класифікації оголошення за deal_class;

– trust_score – числова оцінка правдоподібності від 0 до 100, що формується на основі прозорих штрафів за цінові аномалії та ризикові ознаки;

– deal_class – підсумковий клас оголошення: «підозріла ціна», «вигідна угода», «ринкова ціна» або «переоцінено». Призначений для зрозумілої інтерпретації результату користувачем;

– finance_signal – бінарна ознака наявності фінансової мови в описі. Є однією з причин штрафу при розрахунку trust_score.

Результат trust-score особливо важливий для кінцевого інтерфейсу. Саме він перетворює прогноз на практичну підказку: вигідно, нормально, переоцінено або підозріло.

3.7 Реалізація модуля прогнозування ринкових показників

Модуль прогнозування ринкових показників будує часовий ряд на основі очищених оголошень. Спочатку дані агрегуються за часовими періодами, після чого формуються лагові ознаки, ковзні статистики та трендові змінні. Для кожного цільового показника навчається окрема модель GradientBoostingRegressor.

У реалізації враховано граничний випадок, коли тестовий період дорівнює нулю. У попередній версії коду така ситуація призводила до формування порожньої навчальної вибірки через некоректний зріз iloc[:-0]. Після виправлення додано окрему гілку для навчання на всьому доступному часовому ряді без виділення тестової частини.

Також виправлено логіку використання моделі для прогнозування майбутніх періодів. Модель model_full, навчена на повному часовому ряді, має використовуватися саме для формування майбутнього прогнозу. Використання моделі, навченої лише на частині даних, призводить до систематичного заниження якості прогнозу на останніх періодах.

Вихідними файлами модуля є market_timeseries.csv, market_forecast.csv, market_forecast_by_manufacturer.csv і market_forecast_metrics.md, що забезпечують можливість аналізу як загальної ринкової динаміки, так і показників окремих сегментів.

Етапи прогнозування ринкових показників включають кілька послідовних дій. На етапі агрегації виконується групування оголошень за часовими періодами з формуванням файлу market_timeseries.csv. Далі формуються лагові ознаки lag_1, lag_2, lag_3, що забезпечують моделі пам'ять про попередні періоди, та обчислюються ковзні статистики rolling_mean і rolling_std для згладжування випадкових коливань.

На етапі навчання для кожного цільового показника будується окрема модель GradientBoostingRegressor. Якість отриманих моделей оцінюється шляхом порівняння з baseline, результати якого фіксуються у файлі

market_forecast_metrics.md. Завершальним етапом є побудова прогнозу на майбутні періоди: модель навчається на всьому доступному масиві даних, після чого формується таблиця future_df і зберігається у файлі market_forecast.csv.

Для прототипу цього достатньо, щоб показати не лише поточний зріз ринку, а й короткострокову тенденцію.

3.8 Реалізація інтерактивного дашборду користувача

Дашборд реалізовано на Streamlit і об'єднує результати всіх модулів системи: очищені дані, метрики моделі, ринкові індикатори, trust-score і прогноз часових рядів. Для користувача це головна точка входу, де результати обробки стають доступними для перегляду та аналізу.

Інтерфейс спроєктовано з урахуванням принципу мінімальної когнітивної навантаженості: одночасне відображення всіх таблиць, графіків і технічних полів ускладнює сприйняття результатів. З цієї причини інформація структурована за тематичними блоками з можливістю послідовного перегляду.

Ключовим елементом інтерфейсу є система фільтрів, що дозволяє переглядати результати в розрізі окремих марок, років випуску, типів кузова або класів угод. Це перетворює дашборд із статичного звіту на інструмент дослідження ринку. Фільтри також забезпечують перевірку адекватності результатів у межах окремих сегментів: наприклад, прогноз для конкретної моделі можна порівняти з медіанною ціною відповідного сегмента.

Для підвищення швидкодії інтерфейсу застосовано кешування завантаження даних засобами Streamlit. Обробка великих CSV-файлів при кожному оновленні сторінки є ресурсозатратною, тому кешування забезпечує зручність роботи без зміни логіки обчислень.

У дашборді реалізовано чотири основних блока:

– огляд ринку – кількість оголошень, типові ціни, криві знецінення та порівняння брендів за сегментами;

– прогноз ціни – очікувана ціна для конкретного автомобіля та її відхилення від ринкової медіани;

– довіра та вигідні пропозиції – класи угод, підозрілі оголошення та розподіл trust-score.

Прогноз ринкових показників – динаміка та прогноз медіанної ціни, кількості оголошень і коефіцієнта варіації.

Дашборд є останнім шаром системи, але не менш важливим за модель. Саме він робить результати доступними для перегляду без читання коду.

Інтерфейс дашборду наведен на рисунку 3.8.

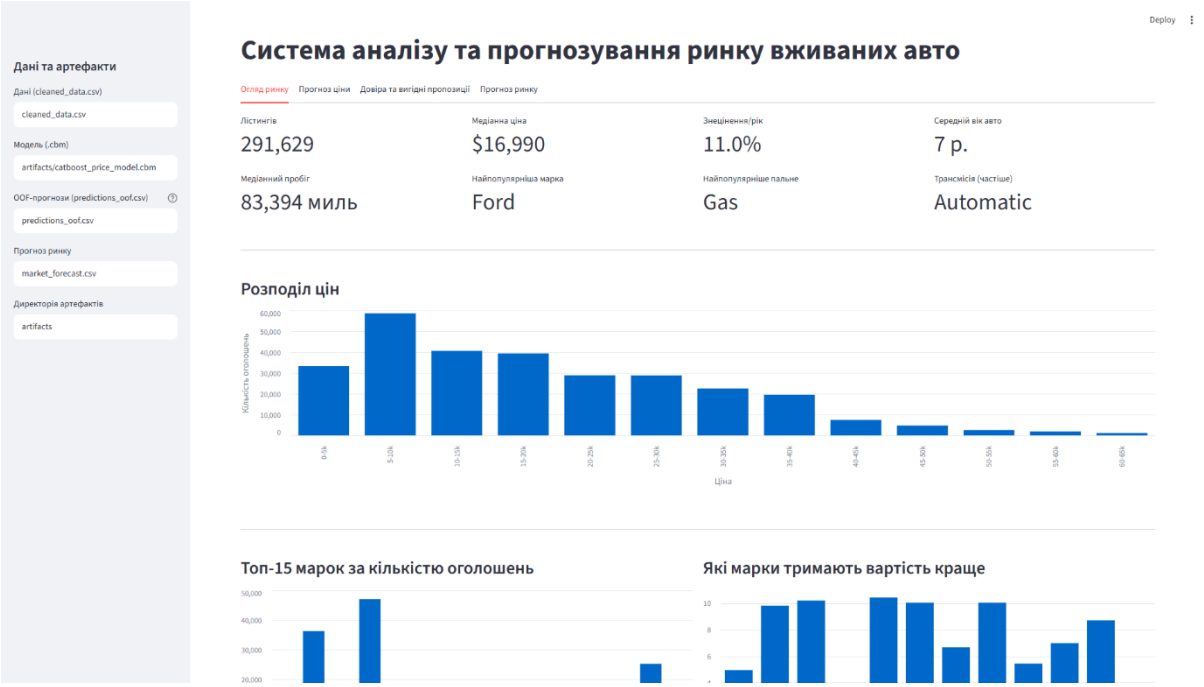

Рисунок 3.8 – Вигляд дашборду

3.9 Тестування програмної системи та перевірка працездатності

Перевірка системи виконувалася на рівні синтаксису, запуску модулів і автоматичних тестів. У файлі tests/test_all.py реалізовано перевірку

ключових функцій очищення, формування ознак, розрахунку trust-score і прогнозування. За результатами виконання pytest підтверджено успішне проходження 26 тестів.

Тестування виконувалося на локальному середовищі з використанням синтетичних мінімальних датафреймів, що відтворюють структуру реального датасету Craigslist Vehicles. Такий підхід дозволяє перевіряти логіку модулів незалежно від наявності повного датасету та забезпечує швидке виконання тестів без завантаження сотень тисяч записів.

Наявність автоматичних тестів є важливою вимогою для системи з модульною архітектурою. Зміна назви колонки в одному модулі може порушити роботу іншого, і саме тести дозволяють виявити такі помилки без повного запуску pipeline. Це підтверджує, що реалізація має базовий рівень контролю якості, а не є одноразовим експериментом.

Окремо виконувалася перевірка модуля прогнозування часових рядів на синтетичному ряді. Це дозволило виявити проблему з граничним випадком test_periods=0 і некоректним використанням model_full для майбутнього прогнозу. Виявлені помилки були описані та усунені, що відображає реальний процес розробки програмного забезпечення.

Поточне покриття тестами охоплює основні функціональні сценарії. Для подальшого розвитку системи доцільно додати інтеграційні тести повного запуску конвеєру на зменшеному зразку датасету, що дозволить виявляти помилки на рівні взаємодії між модулями.

3.10 Аналіз отриманих результатів та обмеження реалізації

У результаті розроблено прототип, який демонструє повний цикл аналізу ринкових показників. Система виконує очищення даних, формування ознак, навчання моделі ціни, розрахунок агрегатів, прогнозування часових рядів, оцінювання довіри до оголошень та

відображення результатів через інтерактивний дашборд, що відповідає поставленій меті роботи.

Реалізований конвеєр складається з семи послідовних етапів. На першому етапі вихідний CSV-файл проходить очищення: видаляються дублікати, фільтруються некоректні ціни та виявляються фінансові оголошення – підозрілі записи зберігаються окремо у flagged_listings.csv. На другому формуються ознаки: вік автомобіля, пробіг на рік, частотне кодування високо-кардинальних категорій та текстові сигнали опису. Третій етап – навчання CatBoostRegressor на логарифмованій цільовій змінній з розбиттям 70/15/15 та збереженням артефактів. Четвертий розраховує ринкові індикатори: медіанні ціни за сегментами, криві знецінення та тижневу динаміку. П'ятий формує out-of-fold прогнози і розраховує trust-score з прозорою системою штрафів. Шостий будує часові ряди агрегованих показників і прогнозує їх засобами GradientBoostingRegressor. Сьомий – інтерактивний Streamlit-дашборд, що об'єднує результати всіх модулів.

Перевагою реалізації є модульність: кожен етап виконується незалежно, а результати зберігаються у файлах, що спрощує перевірку системи та локалізацію помилок. Додатковою перевагою є поєднання прогнозу ціни з trust-score, що дозволяє уникнути хибної інтерпретації аномально низьких цін як вигідних пропозицій. Водночас реалізація має ряд обмежень: залежність від якості датасету Craigslist Vehicles, спрощена обробка текстових описів та короткий горизонт прогнозування часових рядів. Крім того, система не замінює експертну перевірку транспортного засобу і не має доступу до інформації про технічний стан або юридичні ризики — результати слід використовувати як аналітичну підтримку, а не як підставу для автоматичного рішення.

Напрями подальшого розвитку включають підключення нових джерел даних, розширення NLP-аналізу описів, додавання SHAP-пояснень для окремих прогнозів, вдосконалення часових моделей та реалізацію механізму регулярного оновлення даних.

ВИСНОВКИ

У кваліфікаційній роботі розроблено програмну систему аналізу та прогнозування ринкових показників із застосуванням інтелектуальних підходів. Як прикладну область використано ринок вживаних автомобілів, що дозволило перевірити систему на складних табличних даних з числовими, категоріальними, текстовими та часовими ознаками.

У першому розділі проаналізовано предметну область, особливості ринку вживаних автомобілів, проблеми вихідних ринкових даних та наявні підходи до прогнозування ціни і ринкової динаміки. Встановлено, що для вирішення поставленої задачі недостатньо побудови окремої моделі – необхідний повний конвеєр, що охоплює очищення даних, формування ознак, аналіз сегментів, прогнозування та візуалізацію результатів.

У другому розділі обґрунтовано вибір методів та технологій. Для прогнозування ціни обрано CatBoostRegressor як модель, що ефективно працює з табличними даними та категоріальними ознаками. Для прогнозування часових рядів ринкових показників застосовано GradientBoostingRegressor з лаговими та ковзними ознаками. Обґрунтовано правила формування trust-score, метрики оцінювання, пороги очищення та склад програмного стеку.

У третьому розділі описано програмну реалізацію семи функціональних модулів: очищення даних, формування ознак, навчання моделі ціни, розрахунку ринкових індикаторів, оцінювання довіри до оголошень, прогнозування часових рядів та інтерактивного Streamlit-дашборду. Ключові параметри системи централізовано винесено у config.py.

Практичним результатом є прототип, що реалізує повний цикл обробки: від очищення вихідних даних до інтерактивного інтерфейсу з прогнозами, ринковими індикаторами та класифікацією оголошень за рівнем довіри. Окремим результатом є методика trust-score на основі out-of-

fold прогнозів і прозорої системи штрафів, що дозволяє розмежовувати реально вигідні пропозиції та неправдоподібні оголошення.

Працездатність системи підтверджено успішним виконанням 26 автоматичних тестів pytest.

Якість прогнозування визначається не лише вибором моделі, а й якістю підготовки даних. Вихідні оголошення містять пропуски, аномальні ціни та дублікати, тому значну увагу приділено очищенню та формуванню ознак. Використання логарифмованої цільової змінної дозволило збалансувати вплив дорогих автомобілів на функцію втрат і забезпечити адекватні прогнози для масових сегментів ринку.

Система поєднує два рівні прогнозування: оцінювання вартості окремого транспортного засобу та аналіз агрегованої динаміки ринку. Таке поєднання забезпечує практичну цінність як для окремого користувача, так і для аналітики ринкових сегментів у цілому.

До обмежень роботи належать залежність від датасету Craigslist Vehicles, спрощена обробка текстових описів, короткий горизонт прогнозування та відсутність зовнішніх економічних факторів. Для промислового використання системи додатково потрібно реалізувати регулярне оновлення даних, моніторинг якості моделей та розширене логування.

Напрямами подальшого розвитку є поглиблення пояснюваності прогнозів засобами SHAP, розширення NLP-аналізу текстових описів оголошень та адаптація системи до інших предметних областей з аналогічною структурою табличних даних – ринку нерухомості, побутової техніки або товарів на маркетплейсах.

Таким чином, поставлену мету досягнуто. Розроблена система демонструє завершений приклад застосування інтелектуальних підходів до аналізу та прогнозування ринкових показників і може слугувати основою для подальшого розвитку аналітичних сервісів електронних ринків.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Craigslist vehicles dataset. *Kaggle*. URL: https://www.kaggle.com/datasets/mbaabuharun/craigslist-vehicles (date of access: 10.06.2026).

2. Акімова Г. М., Висоцька В. А., Карпенко М. Ю. Методи та засоби штучного інтелекту. Львів: Видавництво Львівської політехніки, 2020. 412 с.

3. Géron A. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. 3rd ed. O'Reilly Media, 2022. 861 p.

4. Provost F., Fawcett T. Data science for business. O'Reilly Media, 2013. 409 p.

5. Breiman L. *Machine learning*. 2001. Vol. 45, no. 1. P. 5–32. URL: https://doi.org/10.1023/a:1010933404324 (date of access: 16.06.2026).

6. Chen T., Guestrin C. XGBoost. *KDD '16: the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, San Francisco California USA. New York, NY, USA, 2016. URL: https://doi.org/10.1145/2939672.2939785 (date of access: 07.07.2026).

7. CatBoost: unbiased boosting with categorical features / L. Prokhorenkova et al. *arXiv*. 2019. URL: https://doi.org/10.48550/arXiv.1706.09516 (date of access: 07.07.2026).

8. Hyndman R. J., Athanasopoulos G. Forecasting: principles and practice. 3rd ed. OTexts, 2021. URL: https://otexts.com/fpp3/ (date of access: 07.07.2026).

9. Kuhn M., Johnson K. Applied predictive modeling. Springer, 2013. 600 p.

10. Molnar C. Interpretable machine learning. 2nd ed. 2022. URL: https://christophm.github.io/interpretable-ml-book/ (date of access: 07.07.2026).

11. CatBoost documentation. *CatBoost*. URL: https://catboost.ai/docs/ (date of access: 10.06.2026).

12. Pandas documentation. *Pandas*. URL: https://pandas.pydata.org/docs/ (date of access: 10.06.2026).

13. NumPy documentation. *NumPy*. URL: https://numpy.org/doc/ (date of access: 10.06.2026).

14. Scikit-learn user guide. *Scikit-learn*. URL: https://scikit-learn.org/stable/user_guide.html (date of access: 10.06.2026).

15. Streamlit documentation. *Streamlit*. URL: https://docs.streamlit.io/ (date of access: 10.06.2026).

16. Plotly Python graphing library documentation. *Plotly*. URL: https://plotly.com/python/ (date of access: 10.06.2026).

17. Pytest documentation. *Pytest.* URL: https://docs.pytest.org/ (date of access: 10.06.2026).

18. Joblib documentation. *Joblib*. URL: https://joblib.readthedocs.io/ (date of access: 10.06.2026).