

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

До захисту допущено:

Завідувач кафедри

_____ Олександр РОЛІК

«___» _____ 2025 р.

**Дипломний проєкт
на здобуття ступеня бакалавра
за освітньо-професійною програмою «Інформаційні управляючі системи
та технології»
спеціальності 126 «Інформаційні системи та технології»
на тему: «Система прогнозування попиту на ринку нерухомості на основі
аналізу великих даних»**

Виконав:

студент IV курсу, групи ІС-12

Шипко Станіслав Володимирович

Керівник:

доцент кафедри ІСТ, к.т.н., доцент,

Писаренко Андрій Володимирович

Рецензент:

доцент кафедри ІСТ, к.т.н., доцент,

Лісовиченко Олег Іванович

Засвідчую, що у цьому дипломному
проєкті немає запозичень з праць інших
авторів без відповідних посилань.

Студент _____

Київ – 2025 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 126 «Інформаційні системи та технології»

Освітньо-професійна програма «Інформаційні управляючі системи та технології»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Олександр РОЛІК

«___» _____ 2025 р.

ЗАВДАННЯ
на дипломний проєкт студенту
Шипку Станіславу Володимировичу

1. Тема проєкту «Система прогнозування попиту на ринку нерухомості на основі аналізу великих даних», керівник проєкту Писаренко Андрій Володимирович, кандидат технічних наук, доцент, затверджені наказом по університету від «23» травня 2025 р. № 1705-с.
2. Термін подання студентом проєкту: 9 червня 2025 р.
3. Вихідні дані до проєкту: середня похибка моделі на контрольній вибірці не більше 10%, коефіцієнт детермінації більше 80%, середній час на обробку одного об'єкта не перевищує двох секунд.
4. Зміст пояснювальної записки: опис предметної області, аналіз сучасних рішень, проєктування системи прогнозування попиту, робота з великими даними, реалізація системи прогнозування попиту, математичне забезпечення, тестування та верифікація.
5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо) діаграма класів, загальна структура системи, алгоритм обробки запиту на прогноз, діаграма прецедентів.
6. Дата видачі завдання 8 березня.

Календарний план

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту	Примітка
1	Ознайомлення з технічним завданням	15.04.–21.04.2025	
2	Аналіз предметної області	22.04.–28.04.2025	
3	Формування мети та задач	29.04.–05.05.2025	
4	Проектування архітектури системи	06.05.–20.05.2025	
5	Розроблення системи	21.05.–04.06.2025	
6	Оформлення текстової документації	05.06.–09.06.2025	

Студент

Станіслав ШИПКО

Керівник

Андрій ПИСАРЕНКО

АНОТАЦІЯ

Система прогнозування попиту на ринку нерухомості на основі аналізу великих даних.

Проект містить 68 с. тексту, 20 рисунків, 4 таблиці, посилання на 22 літературних джерел, додатки та 4 конструкторських документи.

ІНФОРМАЦІЙНА СИСТЕМА, ПРОГНОЗУВАННЯ, НЕРУХОМІСТЬ, XGBOOST, МАШИННЕ НАВЧАННЯ, РИНКОВИЙ ПОПИТ.

Об'єктом розроблення є інформаційна система для прогнозування попиту на житлову нерухомість у великих містах.

Предметом – методи й технічні рішення, які дозволяють реалізувати прогноз ринкової ціни об'єкта нерухомості на основі його характеристик і зовнішніх факторів.

Мета розроблення – створення системи прогнозування на основі підвищення точності оцінки ринкового попиту та вартості житла за рахунок великих даних.

У проєкті використані методи регресійного прогнозування, попередньої обробки даних, логарифмічної трансформації цільової змінної та генерації ознак. Реалізація системи побудована на Python, модель прогнозування створена з використанням XGBoost. Серверна частина реалізована на Flask, база даних – на SQLite. Передбачене збереження історії прогнозів і можливість виводу графіків. Обробка пакетних запитів із CSV-файлів реалізована через окремий модуль.

Отримані результати можуть бути використані агентствами нерухомості, муніципальними структурами, аналітичними платформами, а також окремими користувачами, які бажають об'єктивно оцінити житло перед купівлею або продажем.

SUMMARY

System for predicting interest in the real estate market based on big data analysis.

The project contains 68 pages of text, 20 figures, 4 tables, references to 22 literary sources, annexes, and 4 design documents.

Keywords: information system, real estate, demand forecasting, XGBoost, machine learning, price analysis.

The object of the development is an information system for forecasting demand for residential real estate in large cities.

The subject of the study includes methods and technical solutions that enable the prediction of a property's market price based on its characteristics and external factors.

The goal of the development is to create a forecasting system that enhances the accuracy of estimating market demand and housing prices by leveraging big data.

The project utilizes methods of regression forecasting, data preprocessing, logarithmic transformation of the target variable, and feature generation. The system is implemented in Python, with the forecasting model built using XGBoost. The server-side is developed using Flask, and the database is based on SQLite. The system supports saving the history of forecasts and visualizing the results through charts. Batch request processing from CSV files is implemented via a separate module.

The obtained results can be used by real estate agencies, municipal organizations, analytical platforms, as well as individual users who wish to objectively assess property value before purchasing or selling.

[illegible]

Пояснювальна записка
до дипломного проєкту
на тему: «Система прогнозування попиту на ринку
нерухомості на основі аналізу великих даних»

Київ – 2025 року

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	4
ВСТУП.....	5
1 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ	7
1.1 Сучасний стан ринку нерухомості України та його особливості	7
1.2 Великі дані та їх роль у прогнозуванні попиту.....	9
Висновки до розділу 1	12
2 АНАЛІЗ СУЧАСНИХ РІШЕНЬ	14
2.1 Платформа нерухомості Lun.ua	14
2.2 Платформа нерухомості Zillow.com.....	16
Висновки до розділу 2.....	19
3 МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ	20
3.1 Змістовна постановка задачі	20
3.2 Математична постановка задачі	21
3.3 Обґрунтування методу розв’язання.....	24
3.4 Опис методу розв’язання	26
Висновки до розділу 3.....	27
4 ПРОЄКТУВАННЯ СИСТЕМИ ПРОГНОЗУВАННЯ ПОПИТУ	29
4.1 Визначення вимог до системи	29
4.2 Архітектура системи	31
4.3 Вибір технологій та інструментів.....	33
4.4 Проєктування сховища даних.....	36
Висновки до розділу 4.....	39
5 РОБОТА З ВЕЛИКИМИ ДАНИМИ.....	40
5.1 Методи аналізу великих даних	40
5.1.1 Машинне навчання для погнозування попиту	40

					ІС12.300БАК.005 ПЗ			
Зм.	Лист	№ докум.	Підпис					
Розробив		Шипко С.В.			Система прогнозування попиту на ринку нерухомості на основі аналізу великих даних. Пояснювальна записка	Літ.	Арк.	Аркушів
Перевірив		Писаренко А.В.				Т	2	68
						КПІ ім. Ігоря Сікорського Група ІС-12		
Затв.								

5.1.2 Методи кластеризації та класифікації даних	42
5.2 Алгоритми прогнозування	44
5.3 Навчання моделі	47
Висновки до розділу 5	49
6 РЕАЛІЗАЦІЯ СИСТЕМИ ПРОГНОЗУВАННЯ ПОПИТУ	51
6.1 Розроблення MVP	51
6.2 Демонстрація роботи системи	52
6.3 Подальші кроки з розроблення системи	56
Висновки до розділу 6	57
7 ТЕСТУВАННЯ ТА ВЕРИФІКАЦІЯ	59
7.1 Підходи до тестування програмного забезпечення	59
7.2 Тестування сценаріїв	60
7.3 Аналіз результатів	63
Висновки до розділу 7	64
ВИСНОВКИ	66
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	67

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД – база даних.

ЖК – житловий комплекс.

ІС – інформаційна система.

ПЗ – програмне забезпечення.

API – Application Programming Interface.

CSV – Comma-Separated Values.

MAE – Mean Absolute Error.

ML – Machine Learning.

MLS – Multiple Listing Service.

MVP – Minimum Viable Product.

UI – User Interface.

XGBoost – Extreme Gradient Boosting.

					ІС12.300БАК.005 ПЗ	Арк.
						4
Зм.	Лист	№ докум.	Підпис	Дата		

ВСТУП

Ринок житлової нерухомості в Україні переживає період глибокої перебудови. У таких умовах класичні підходи до оцінки об'єктів нерухомості та прогнозування попиту вже не відповідають новим вимогам до точності, швидкості реагування та адаптивності. Виникає потреба у системах, здатних працювати з великою кількістю різномірних даних і формувати оцінки на основі багатофакторного аналізу.

Розвиток інформаційних технологій і, зокрема, методів машинного навчання відкриває нові можливості для побудови систем, що забезпечують оперативний аналіз ринку, оцінку тенденцій та формування прогнозів. Ідея створення інструмента, який би дозволяв прогнозувати попит на житло на основі оголошень, характеристик об'єкта, місця розташування та ряду зовнішніх факторів, стає актуальною не лише з академічного погляду, а й з прикладної точки зору. Вона дає змогу запровадити нову парадигму прийняття рішень на ринку, де кожна транзакція підтримується об'єктивною оцінкою.

Сучасний стан цифрової екосистеми в Україні сприятливий для розроблення подібних інструментів. Платформи на кшталт Lun.ua, DOM.RIA та Flatfy уже забезпечують доступ до об'єктивних характеристик ринку, однак їх функціональність обмежується переважно переглядом і пошуком оголошень. Інструменти прогнозування, які враховують не лише поточні ціни, а й просторово-часову динаміку попиту, залишаються відсутніми. Впровадження такої системи здатне суттєво змінити підхід до оцінювання об'єктів житлової нерухомості як з боку пересічних користувачів, так і для агентств чи міських департаментів.

Метою проєкту є створення системи прогнозування на основі підвищення точності оцінки ринкового попиту та вартості житла за рахунок великих даних. Це передбачає не лише побудову моделі, яка повертає числову оцінку попиту, а й створення інтерфейсу, бази для зберігання запитів і результатів, а також модулів інтеграції з зовнішніми джерелами. Реалізація системи дозволяє перейти від

					IC12.300БАК.005 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		5

ручного аналізу оголошень до автоматизованої аналітики, яка формує об'єктивні прогнози на основі зібраної статистики.

Основні завдання, які потрібно вирішити в межах проєкту, охоплюють: аналіз сучасного стану ринку та типових патернів попиту; збір та обробку інформації з відкритих платформ; формування навчальної вибірки; побудову моделі прогнозування; реалізацію інтерфейсу взаємодії з користувачем; виведення результатів у форматі таблиць і графіків; перевірку стабільності й точності роботи системи.

Актуальність теми обумовлюється високим попитом на інструменти оцінки вартості об'єктів у нестабільних умовах. Зміни, що відбулися на ринку нерухомості протягом останніх років, змушують адаптуватися як забудовників, так і покупців. Оцінка перспективності локації, доцільності інвестиції або рівня попиту – усе це сьогодні має базуватися не на припущеннях, а на реальних даних. Саме тому система, що базується на аналізі оголошень, просторових характеристиках і поведінкових індикаторах, здатна дати якісно новий інструмент для учасників ринку.

Новизна підходу полягає у поєднанні кількох джерел інформації: агрегованих оголошень, просторових характеристик, ознак інфраструктури та логіки машинного навчання. Це дозволяє відійти від традиційних методів статистичної аналітики та реалізувати динамічний прогноз, який змінюється у відповідь на оновлення даних.

У перспективі результати цієї роботи можуть бути використані як основа для впровадження подібних систем на рівні міських платформ, агентств нерухомості або аналітичних сервісів. Підхід до поєднання оголошень з прогновною моделлю також може знайти застосування в суміжних галузях, від ринку комерційної нерухомості до аналізу туристичних чи логістичних об'єктів.

					ІС12.300БАК.005 ПЗ	Арк.
						6
Зм.	Лист	№ докум.	Підпис	Дата		

1 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Сучасний стан ринку нерухомості України та його особливості

Ринок житлової нерухомості в Україні перебуває у стані активної трансформації під впливом низки зовнішніх і внутрішніх обставин. Починаючи з 2022 року, звичні патерни попиту і пропозиції почали змінюватися, і на перший план вийшли безпека, мобільність, автономність та гнучкість. Пріоритет споживачів перемістився з умовно стандартних параметрів на нові фактори: наявність укриттів, близькість до критичної інфраструктури, альтернативні джерела енергопостачання.

Ситуація в різних регіонах країни значно відрізняється. У центральних і західних областях ринок демонструє відносну активність. Київ, незважаючи на складну безпекову ситуацію, зберігає лідерство за обсягами будівництва та інвестиційною привабливістю. Це пов'язано як із наявністю економічної активності, так і з доступністю фінансування – програма «єОселя», пільгові іпотечні кредити, відносна стабільність банківської системи. У столиці фіксується поступове відновлення первинного ринку, деякі девелопери навіть запускають нові проекти. Водночас на вторинному ринку темпи угод залишаються помірними, із явною перевагою об'єктів у центральних районах або поблизу метро.

У Львові та Івано-Франківську зафіксовано зростання цін як на первинному, так і на вторинному ринку, що частково обумовлено внутрішньою міграцією. Збільшення попиту відбулося переважно за рахунок населення, яке переїхало з прифронтових регіонів. Такі покупці часто розглядають житло в західних містах як довгострокову альтернативу, що позначається на формуванні цін. Одночасно із цим у згаданих містах загострилася проблема з пропозицією: більшість об'єктів розковуються ще на етапі котловану або першої черги.

Для Харкова та Дніпра характерна протилежна динаміка. Через регулярні загрози, обмежене банківське кредитування та нерівномірне відновлення забудов, у цих містах відзначається зменшення кількості активних забудовників, зниження попиту на первинне житло і зростання частки угод у сегменті вторинної

					ІС12.300БАК.005 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		7

нерухомості. Попит сконцентрований переважно у тих районах, які мають відносну безпеку та хорошу транспортну доступність. У таких умовах рішення про купівлю об'єкта супроводжується детальним аналізом ризиків, включно з картами обстрілів і доступністю укриттів.

Новим трендом стало активне зацікавлення заміськими будинками у передмістях великих міст або в селах західних регіонів. Це не лише альтернатива міським квартирам, а й реакція на загальну нестабільність. Особливо привабливими стали об'єкти з автономними системами: генераторами, сонячними панелями, свердловинами. Додатковим фактором стало прагнення мати власну ділянку, яку можна використовувати для забезпечення продовольчої незалежності у разі тривалих надзвичайних ситуацій.

Поступово цифровізація змінює структуру комунікації між продавцями та покупцями. Платформи на кшталт LUN.ua, DOM.RIA, Flatfy перетворюються з агрегаторів оголошень на повноцінні аналітичні сервіси. Наприклад, LUN надає доступ до статистики за цінами, кількістю угод у конкретних районах, активністю забудовників тощо. Проте навіть найрозвиненіші з українських платформ поки що не пропонують інструментів прогнозування, йдеться лише про відображення поточної ситуації. Саме ця особливість формує підґрунтя для побудови систем нового типу, здатних не тільки фіксувати дані, а й обчислювати очікувані зміни на основі поточної динаміки.

За даними порталу Flatfy, середня вартість квадратного метра в новобудовах у Києві на початку 2024 року сягала приблизно 1450 доларів США, у Львові – 1080, у Харкові – 720. При цьому вторинний ринок характеризується більшою варіативністю, залежно від стану будинку, поверху, транспортної доступності та навіть наявності бомбосховища або підвалу у будівлі [1].

Актуальна статистика демонструє: попит не просто відновився, а став набагато сегментованішим. У певних локаціях спостерігається дефіцит об'єктів через зростання кількості охочих, водночас інші райони, особливо в містах, що постраждали від воєнних дій, фактично втрачають інвестиційну привабливість.

					ІС12.300БАК.005 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		8

Традиційні поняття «елітний центр» чи «доступне передмістя» змінюються залежно від зовнішніх загроз і змін у способі життя.

Інфографіка на рисунку 1.1 ілюструє, наскільки змінилася структура пропозиції купівлі будинку за останні роки.

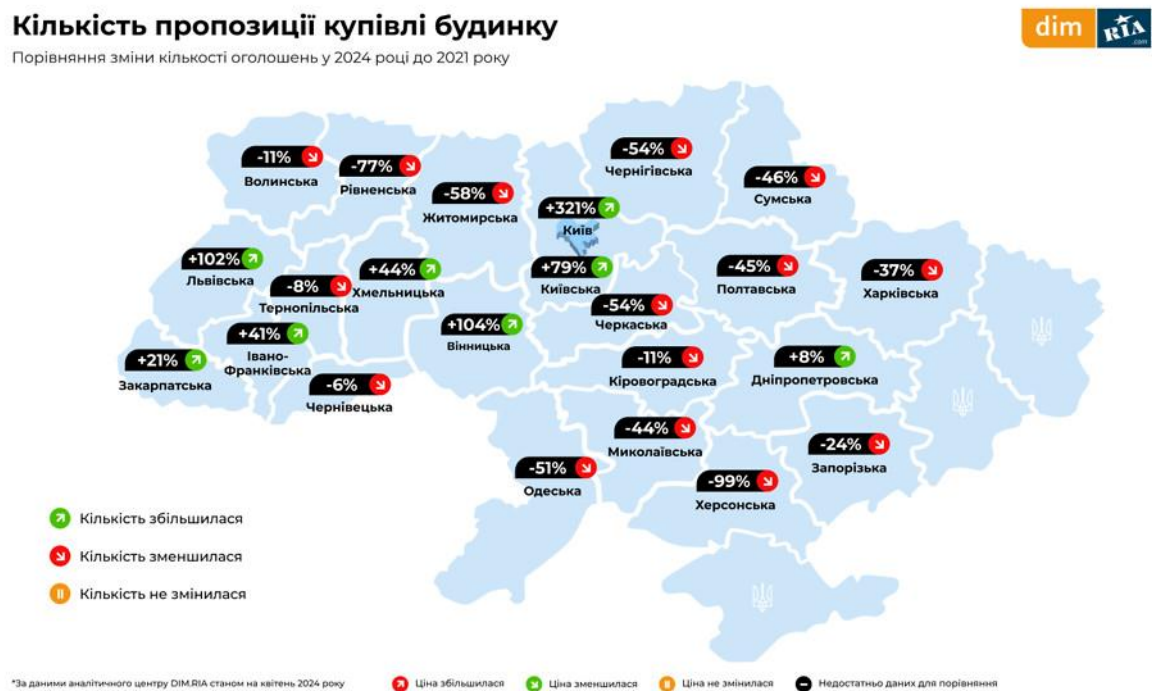


Рисунок 1.1 – Порівняння пропозицій купівлі будинку у 2024 році з періодом до 2021 року [2]

Якщо у 2021 році основний попит фокусувався навколо Києва, Одеси та Харкова, то у 2024-му центр активності змістився у бік західних областей: Львівської, Івано-Франківської, Закарпатської. Частково це пояснюється змінами у демографічній структурі населення, а частково – активною позицією місцевих органів влади щодо підтримки новобудов та спрощення дозвільних процедур.

1.2 Великі дані та їх роль у прогнозуванні попиту

Сучасний ринок нерухомості перетворюється на складну екосистему, де класичні методи оцінювання вже не дають потрібної гнучкості. Поведінка покупців

змінюється швидше, ніж з'являються нові методи аналітики, а ринкові умови реагують на десятки факторів, від законодавчих змін до погоди в конкретному місті. Щоб встигати за ритмом змін, потрібен підхід, що дозволяє бачити не лише загальні тенденції, а й контекст кожної конкретної ситуації. Саме тут великі дані стають тим ресурсом, який дозволяє перейти від загального бачення до конкретного передбачення.

У буквальному сенсі "великі дані" означають не тільки великі обсяги інформації, а й високу швидкість їхнього оновлення, неоднорідність структур, наявність значного шуму та потребу в автоматизованих інструментах обробки. Це дані, які вже неможливо опрацювати вручну або звичайними табличними інструментами. Для ринку нерухомості до таких даних належать оголошення з цифрових платформ, записи в державних реєстрах, результати супутникового моніторингу, поведінкові патерни користувачів, демографічна статистика, дані про інфраструктуру, показники безпеки, транспортна доступність, відгуки з соцмереж і ще з десятків інших типів джерел.

На прикладі українського ринку вже зараз помітна поява цифрових слідів, які можна використовувати як джерело сигналів. Наприклад, кількість переглядів конкретного об'єкта на Lun.ua чи Dom.gia часто прямо корелює з реальним попитом. Аналогічно, частота появи нових оголошень в певному районі дає непряме уявлення про динаміку пропозиції. Проте без автоматизованої системи обробки ці сигнали губляться у потоці інформації. Тут і з'являється сенс побудови аналітичної моделі, яка зможе цікавити не лише кількісні показники, а й якісні взаємозв'язки.

Одним із найважливіших аспектів роботи з великими даними у сфері нерухомості є поєднання різнорідних джерел. Наприклад, оголошення про продаж житла можуть бути зібрані з кількох платформ, кожна з яких має свою структуру. Дані з кадастру мають іншу логіку, часто містять багаторівневі вкладення, а карти транспортної доступності додають геопросторовий компонент. Щоб об'єднати ці джерела, потрібно не просто склеїти їх у єдину таблицю – необхідно побудувати логіку трансформації: очистити, нормалізувати, заповнити пропущені значення,

					ІС12.300БАК.005 ПЗ	Арк.
						10
Зм.	Лист	№ докум.	Підпис	Дата		

виявити суперечності. І лише після цього ці дані можна подавати в модель для подальшого аналізу.

Інструменти машинного навчання дозволяють у таких даних побачити закономірності, які неочевидні при звичайному перегляді. Наприклад, виявляється, що на зростання попиту більше впливає не загальна площа, а співвідношення площі до кількості кімнат. Або що попит на житло в новобудовах зростає після завершення побудови соціальної інфраструктури в межах 500 метрів.

Особливо цікавими є поведінкові сигнали користувачів. Наприклад, підвищений інтерес до певного району можна зафіксувати ще до появи офіційної статистики. Це можливо завдяки тому, що користувачі починають частіше відкривати оголошення з цієї зони, зберігати їх у закладки, надсилати повідомлення продавцям. У поєднанні з даними про цінову динаміку та кількість угод це дозволяє моделі "відчувати", що інтерес до району зростає ще до того, як він стане помітним для широкого загалу. Саме така багатосарова інтеграція джерел – ключ до підвищення точності прогнозів у швидкозмінному середовищі [3].

Велика частина ефективності моделі залежить не лише від алгоритму, а й від того, які ознаки сформовані. У випадку ринку нерухомості це можуть бути не лише класичні "площа" чи "ціна", а й похідні характеристики: відстань до найближчої школи, середня кількість оголошень у радіусі 1 км, тип будівлі (моноліт, цегла, панель), дата останнього ремонту, тривалість оголошення на платформі, середня кількість переглядів тощо. Іноді навіть такі нетривіальні параметри, як наявність фото у оголошенні або опис у певному стилі, можуть вказувати на вищу ймовірність укладання угоди.

З технічного боку, для роботи з великими даними використовуються інструменти, які дозволяють автоматизувати весь цикл обробки. На першому етапі йде збір, зазвичай через парсери або API. Потім виконується очистка і нормалізація, далі – формування ознак і тренування моделі. Після побудови моделі система переходить до етапу прогнозування та виводу результатів у вигляді графіків, таблиць або інтерфейсу. У проєктах, подібних до Zillow, такий підхід реалізований повноцінно, із використанням тисяч ознак і мільйонів записів [4].

					ІС12.300БАК.005 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		11

Застосування великих даних дозволяє змінити сам підхід до прийняття рішень. Якщо раніше аналітик міг робити висновки лише на основі історичної статистики або власного досвіду, то тепер можна побачити вплив кожного окремого параметра у відносному вираженні. Наприклад, модель може показати, що при наявності метро в межах 10 хвилин пішки ймовірність продажу зростає на 27%, а якщо поруч є парк – ще на 12%. У результаті формується набір пріоритетів, які можна враховувати при оцінці житла або плануванні забудови.

Ще одна перевага використання великих даних – це можливість адаптації моделі у режимі реального часу. Система може постійно оновлювати вхідні дані, автоматично адаптувати модель, перераховувати значення прогнозу. Це особливо важливо в умовах нестабільності: наприклад, після локальної події зміни в попиті можуть відбутись протягом доби, і лише гнучка система здатна на це зреагувати.

У контексті воєнного стану додаткову роль відіграють показники безпеки. Дані про частоту повітряних тривог, наявність укриттів, розташування житла відносно потенційних об'єктів ризику стають дуже важливими. Як стверджується у [5], в країнах із підвищеним ризиком інфраструктурних інцидентів (Ізраїль, Колумбія, частково Туреччина), моделі, які враховують фактори безпеки, суттєво перевершують за точністю ті, що працюють лише з економічними показниками. Україна сьогодні потребує саме таких моделей: адаптованих, чутливих до змін і орієнтованих на локальні умови.

Висновки до розділу 1

Ринок житлової нерухомості в Україні нині перебуває у стані глибокої перебудови. Ситуація на ринку нерівномірна, якщо для Києва характерне поступове відновлення забудов і збереження інвестиційного інтересу, то в західних регіонах попит виріс через внутрішню міграцію, а східні області залишаються у фазі невизначеності. Формується нова логіка поведінки: безпека, автономність, наявність укриттів і транспортна доступність стають важливішими за площу чи

					ІС12.300БАК.005 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		12

поверх. Усе це створює потребу в системах, які можуть аналізувати ситуацію не в загальних цифрах, а з урахуванням місцевого контексту та конкретних умов.

Цифрові сервіси поступово змінюють спосіб взаємодії учасників ринку. Платформи типу Lun.ua стали головним джерелом інформації для тисяч користувачів. Водночас такі сервіси поки не пропонують прогностичних функцій, вони лише відображають поточну ситуацію. Прогнозування попиту, що враховує не лише ціни й площу, а й інфраструктуру, поведінкові сигнали, наявність укриттів чи навіть сезонність, вимагає зовсім іншого рівня аналітики.

Суть роботи з великими даними полягає не лише в обробці великих обсягів інформації. Необхідним стає вміння поєднувати десятки різномірних джерел, очищати, нормалізувати й трансформувати їх у такий вигляд, який дозволяє побачити приховані закономірності. Це можуть бути неочевидні зв'язки між типом будинку й динамікою попиту, зміни інтересу до району після відкриття нової школи або зростання ціни через зменшення кількості оголошень. Такі сигнали не завжди фіксуються класичною статистикою, але добре вловлюються системами з автоматизованою логікою.

Поведінка покупців теж починає залишати цифрові сліди. Перегляди оголошень, збереження в закладки, фільтри при пошуку – усе це можна конвертувати в ознаки для моделі. Саме завдяки таким даним система може визначати підвищення інтересу до конкретної зони ще до того, як почнуться активні продажі. Аналогічно, після зовнішніх подій: повітряних тривог, обстрілів, запуску нових маршрутів – модель, що має актуальні вхідні дані, зможе адаптувати прогноз практично в режимі реального часу.

Важливим також є поєднання технічних інструментів із логікою інтерпретації. Система повинна не просто видавати число – вона має пояснювати, як саме ці значення пов'язані з конкретними характеристиками об'єкта, району чи навіть соціальними змінами. У цьому контексті великі дані стають не тільки джерелом інформації, а й середовищем, у якому формується новий підхід до аналітики ринку нерухомості.

2 АНАЛІЗ СУЧАСНИХ РІШЕНЬ

2.1 Платформа нерухомості Lun.ua

Lun.ua – це одна з найпомітніших українських онлайн-платформ, яка спеціалізується на збиранні та візуалізації інформації про житлову нерухомість. Вона працює як агрегатор, що поєднує пропозиції забудовників і ріелторів з багатьох джерел, зокрема з офіційних сайтів, порталних платформ і базових державних реєстрів. Завдяки цьому Lun.ua охоплює більшість великих міст України, включаючи Київ, Львів, Харків, Дніпро, Одесу.

Головна перевага платформи полягає в тому, що всі оголошення проходять автоматичну обробку: фільтруються дублікати, структурується опис, нормалізуються поля типу «площа» чи «поверховість», додаються мапи з точками розташування.

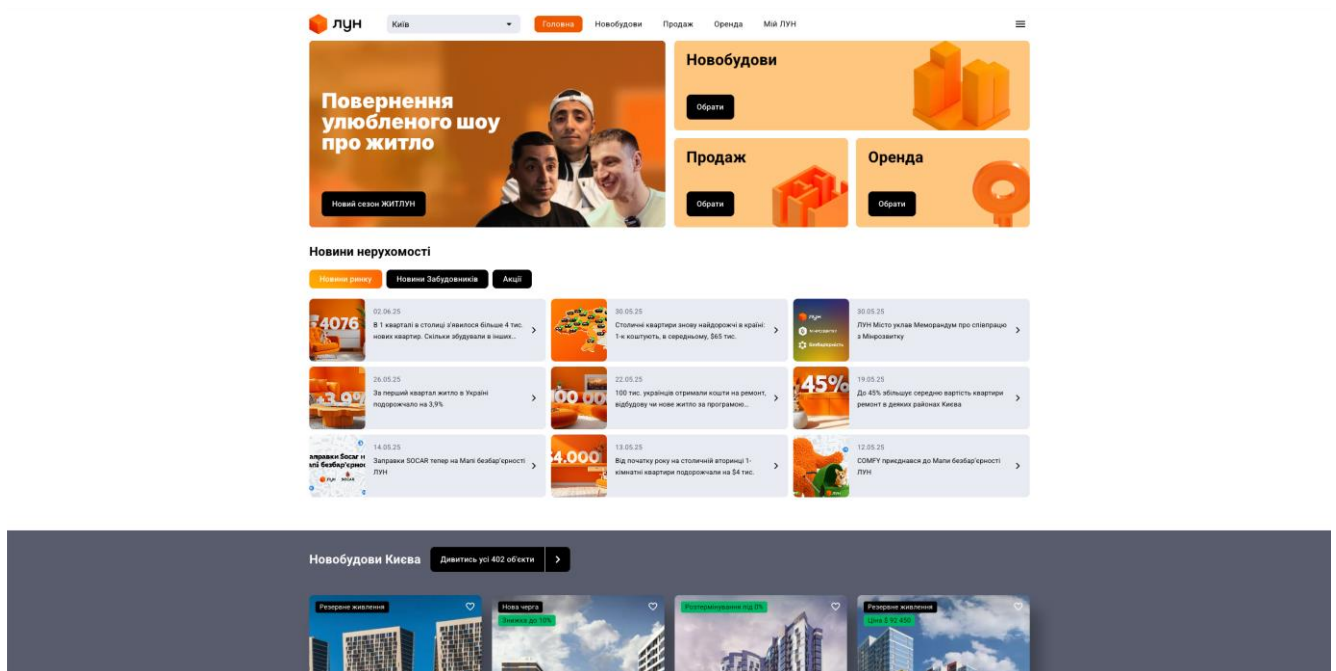


Рисунок 2.1 – Головна сторінка сайту Lun.ua [6]

На рисунку 2.1 наведено вигляд головної сторінки платформи. Інтерфейс орієнтований на швидкий пошук за головними параметрами: тип об'єкта, ціна, район, кількість кімнат, площа. У верхній частині розміщено карту, яка дозволяє

					ІС12.300БАК.005 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		14

одразу побачити географічний розподіл активних оголошень. Візуальна подача результатів робить сайт зручним як для пересічного користувача, так і для тих, хто працює з великими обсягами даних у форматі парсингу.

Формат оголошень на Lun добре підходить для побудови інформаційної системи, оскільки містить усі головні характеристики об'єкта: назву ЖК (якщо є), адресу, тип об'єкта, кількість кімнат, площу, поверх, загальну ціну, вартість за квадратний метр, фото, дату розміщення, розташування на мапі. В окремих випадках також доступні дані про забудовника, терміни введення в експлуатацію, клас житла, матеріали, паркінг, наявність комунікацій. Сторінку пошуку нерухомості наведено на рисунку 2.2.

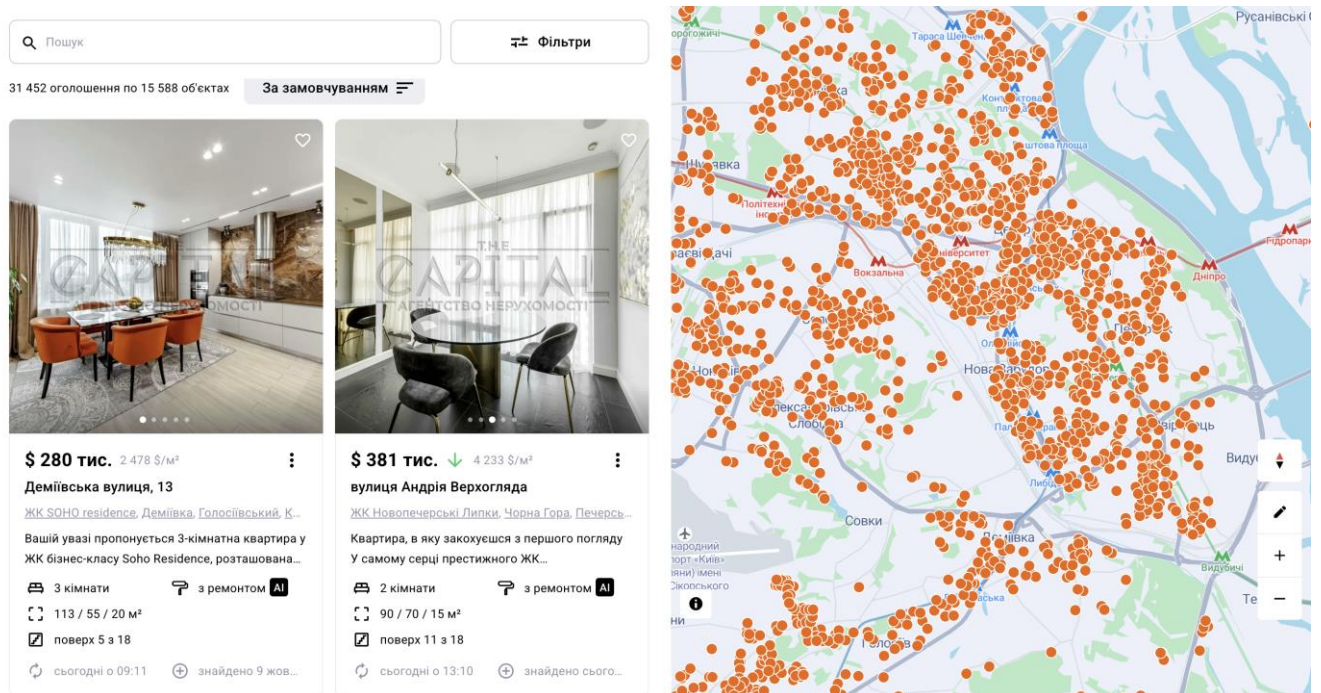


Рисунок 2.2 – Сторінка пошуку нерухомості на платформі Lun.ua [6]

Ці дані дозволяють сформувати базу для подальшого аналізу: можна автоматично підрахувати середню ціну в районі, оцінити динаміку появи нових оголошень, виявити райони з дефіцитом або надлишком пропозиції. Завдяки цьому Lun.ua потенційно може слугувати джерелом як для формування навчальної вибірки, так і для регулярного оновлення прогнозної моделі.

Однією з особливостей Lun є високий ступінь стандартизації опису об'єктів. Хоча оголошення надходять із різних джерел, структура сторінок витримана у єдиному форматі.

Окрім основної бази даних про об'єкти, Lun також надає додаткові шари інформації: мапи житлових комплексів, фото ходу будівництва, рейтинг довіри до забудовника. Усе це поки не є обов'язковим для прогнозу, але може бути корисним при подальшому розвитку системи.

Серед недоліків платформи варто згадати про відсутність відкритого API. Автоматизоване збирання даних потребує побудови спеціального скрипта для парсингу HTML-структур. Крім того, деякі об'єкти мають неповні або застарілі дані – наприклад, не вказано точну адресу або не оновлено дату будівництва. Для таких випадків потрібна попередня валідація та очистка перед збереженням у базу.

І останній нюанс – обмеження по містах. Хоч Lun охоплює більшість обласних центрів, проте дрібніші міста або селища поза межами великих агломерацій не завжди мають повноцінне представлення. Якщо в майбутньому виникне потреба у масштабуванні прогнозу на всю територію України, доведеться комбінувати дані Lun із додатковими джерелами.

2.2 Платформа нерухомості Zillow.com

Zillow.com – це одна з наймасштабніших онлайн-платформ у сфері житлової нерухомості у США, яка об'єднує функціональність пошукового сервісу, інструментів аналітики, а також алгоритмів автоматичної оцінки. Основна мета цього ресурсу полягає не лише в інформуванні користувача, а у формуванні комплексного цифрового середовища, де доступна повна інформація про об'єкти, історію їх продажів, прогнози вартості та ринкові тренди.

Головна сторінка платформи, показана на рисунку 2.3, одразу надає доступ до основних функцій: пошук житла за містом або індексом, розширена мапа з позначеними об'єктами, статистика по регіонах, рейтинг шкіл, податки, зони страхування, потенційні ризики тощо.

					ІС12.300БАК.005 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		16

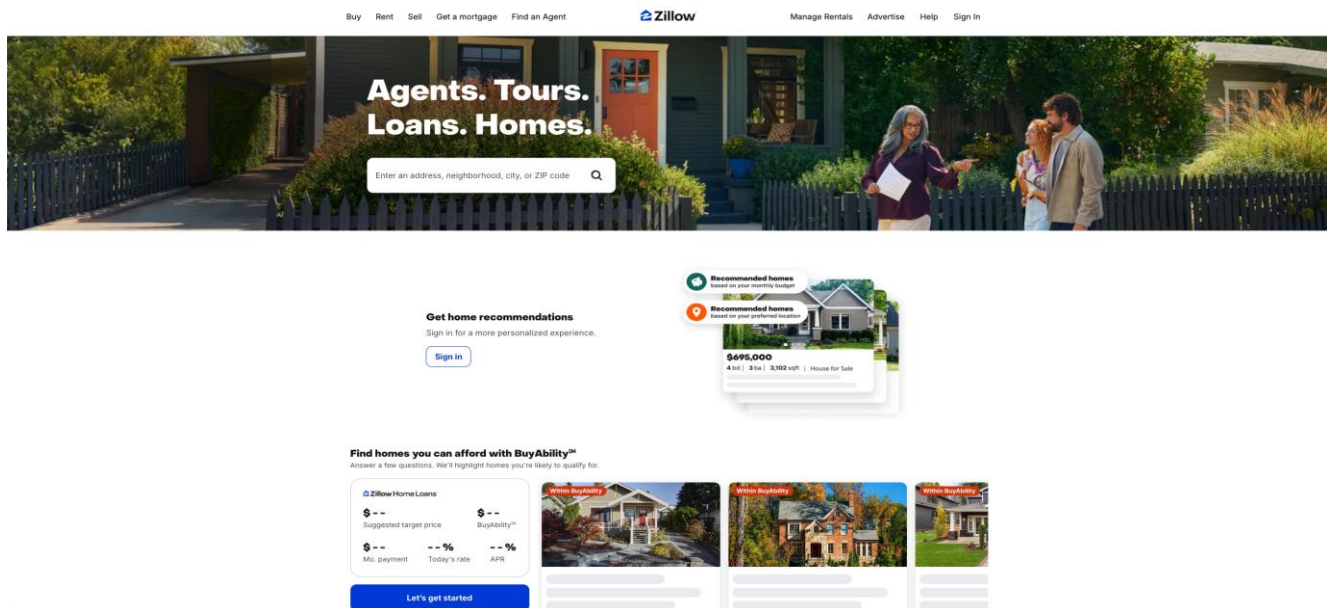


Рисунок 2.3 – Головна сторінка сайту Zillow.com [7]

Компонент, який виділяє Zillow серед аналогічних сервісів – інструмент Zestimate. Це прогнозна модель, яка обчислює очікувану ринкову вартість об'єкта нерухомості на основі набору змінних. До розрахунку враховуються: площа, кількість кімнат, рік будівництва, тип об'єкта, історія продажів, сусідні об'єкти, тренди в районі, показники шкіл, податки та десятки інших параметрів. За словами самих розробників, у певних регіонах точність цього інструмента коливається в межах $\pm 3\%$ для популярних об'єктів.

На відміну від більшості українських сервісів, Zillow реалізує принцип прогносної аналітики. Користувач не лише бачить поточну ціну, а й отримує оцінку ймовірної майбутньої вартості з урахуванням загальної ринкової динаміки. Цей підхід дозволяє оцінити доцільність покупки або продажу з точки зору часу.

Оголошення на Zillow структуровані у зручному для машинної обробки форматі. Сторінка пошуку, яка зображена на рисунку 2.4, поєднує текстову інформацію з мапою, фільтрами, статистичними графіками та динамікою зміни цін у регіоні.

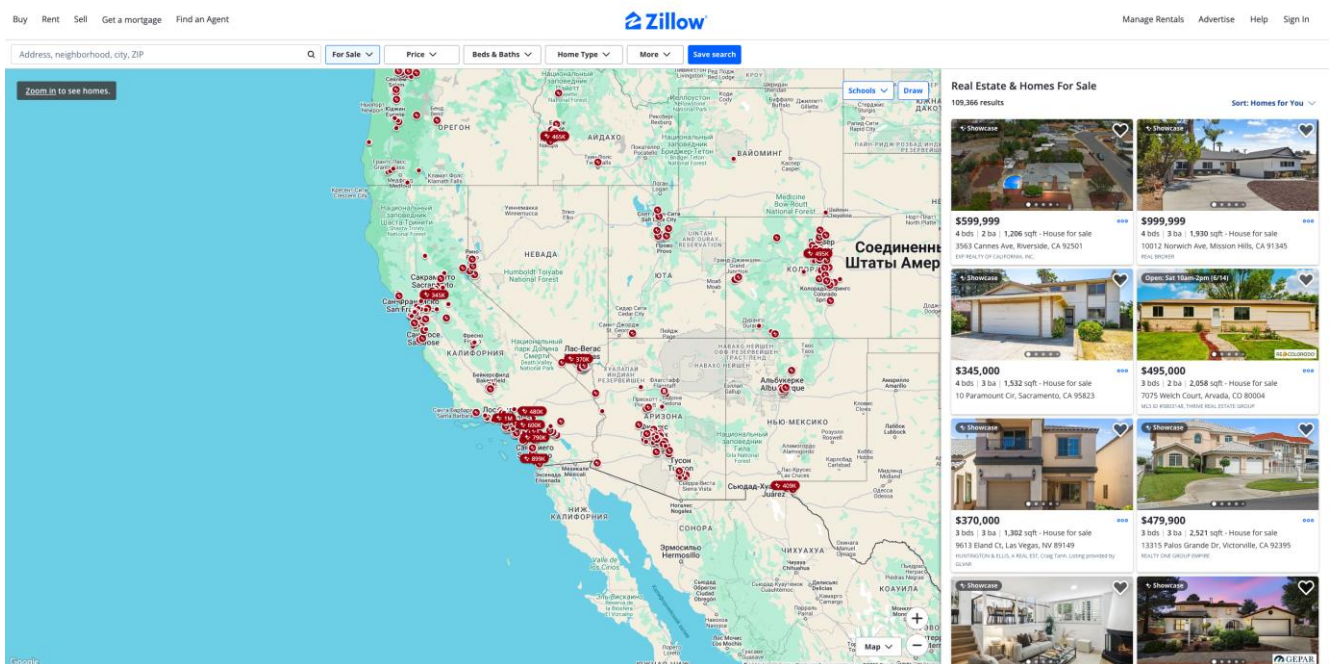


Рисунок 2.4 – Сторінка пошуку нерухомості на платформі Zillow.com [7]

Крім самої оцінки об'єктів, Zillow також генерує тренди для мікрорайонів. Користувач отримує графіки змін цін за квартали, прогноз на кілька місяців уперед, дані про кількість оголошень у конкретному районі, середній час, протягом якого об'єкти перебувають у продажу, і навіть частоту знижок. Це відкриває нові можливості для системного аналізу.

Особливо цінною є інтеграція з локальними базами MLS, яка забезпечує автоматичне оновлення інформації. Завдяки цьому зміни на ринку миттєво потрапляють у систему, а моделі прогнозу регулярно оновлюються.

Однак є і певні обмеження. По-перше, Zillow не надає повноцінного API, через що сторонні системи не можуть отримати дані напряму. Тому для цілей збору інформації часто доводиться застосовувати техніки парсингу HTML-структури. Незважаючи на технічні складнощі, це дозволяє отримати вибірку, яка підходить для тренування моделей прогнозу у локальних умовах, особливо якщо потрібен приклад функціонального інтерфейсу або структура вхідних ознак.

В українських реаліях логіка роботи Zillow може бути адаптована до локального ринку. Попри відсутність аналогічних інструментів, сам принцип поєднання історичних даних, аналізу сусідніх об'єктів, оцінки динаміки та

прогнозу попиту може стати основою для створення нових рішень. Навіть спрощена модель, яка працює на базі даних із Lun.ua, здатна імітувати частину функцій Zestimate. При цьому система може працювати автономно, без доступу до закритих інструментів Zillow, і бути орієнтованою на українську специфіку.

Висновки до розділу 2

Порівняння платформ Lun.ua та Zillow дозволяє побачити два різних підходи до роботи з інформацією на ринку нерухомості. Український сервіс фокусується на зборі та поданні оголошень у структурованому форматі, з акцентом на зручність пошуку та візуалізацію. Водночас йому поки не притаманна аналітична чи прогностична складова. Це створює передумови для використання Lun як джерела даних для зовнішніх систем, зокрема в межах автоматизованих моделей прогнозування.

Zillow, навпаки, поєднує в собі функції пошуку, аналітики та машинного передбачення. Його модель Zestimate працює на основі десятків параметрів, щоденно оновлюється й формує очікувану ціну об'єкта. Такий підхід демонструє, як великі дані та алгоритми можуть перетворювати статистику в практичну рекомендацію.

У результаті аналізу стає очевидним, що на українському ринку поки не існує рішення, яке б поєднувало широке охоплення, структурованість даних і можливості прогнозу. Саме на цю нішу може орієнтуватися інформаційна система, яка буде створена у межах дипломного проєкту. Вона зможе об'єднати досвід платформ, використовуючи відкриті джерела на зразок Lun і реалізуючи функціонал прогнозування, подібний до Zestimate.

3 МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

3.1 Змістовна постановка задачі

Постановка задачі полягає в оцінці ринкової вартості житлової нерухомості на основі набору її характеристик. Йдеться про прогноз, який дозволяє користувачеві зрозуміти, наскільки зазначена ціна в оголошенні відповідає типовим значенням на ринку, з урахуванням об'єктивних параметрів: площі, типу житла, кількості кімнат, розташування тощо.

Кожен об'єкт описується вектором ознак, що включає числові, бінарні та категоріальні параметри. Частина значень може бути відсутньою або заповненою некоректно. Це робить задачу нестандартною: модель має бути стійкою до неповноти даних, адаптивною до різних форматів та придатною для застосування в широкому діапазоні оголошень.

У проєкті враховується, що оголошення надходять з різних джерел, зокрема, LUN та Zillow. Усі дані проходять попередню уніфікацію: структуруються, очищуються, приводяться до єдиних одиниць виміру. Ціни в доларах автоматично конвертуються у гривні за фіксованим курсом, дати трансформуються у формат ISO, ознаки уніфікуються за назвами. Це забезпечує однорідність вхідної інформації для побудови моделі.

Ціль полягає в тому, щоб на основі набору ознак, притаманних кожному об'єкту, сформулювати функцію, яка повертатиме приблизне ринкове значення нерухомості. Така оцінка не замінює експертного висновку, але дозволяє швидко отримати орієнтир для аналізу або виявлення аномалій.

Для кожного джерела формується окрема вибірка та модель, що дозволяє уникнути змішування контекстів і підвищити точність. Наприклад, у Zillow розміщуються переважно об'єкти у США, а в LUN – по Україні, тому їх цінові діапазони та структура сильно відрізняються. Розділення моделей дозволяє зберігати точність без потреби у складній нормалізації.

Структура датасету дозволила застосовувати крос-валідацію з урахуванням груп, що зменшило ризик переоцінки точності при тестуванні моделі [8].

					ІС12.300БАК.005 ПЗ	Арк.
						20
Зм.	Лист	№ докум.	Підпис	Дата		

Після формування прогнозу система також порівнює його з фактичною ціною з оголошення. На основі цього розраховується тренд: якщо ціна значно перевищує прогноз, тоді попит вважається зростаючим, якщо нижча, тоді зменшується. Ця функція особливо корисна для агентств, які можуть швидко оцінити, наскільки конкурентним є конкретне оголошення.

3.2 Математична постановка задачі

Метою розв'язання задачі прогнозування попиту є побудова моделі, яка за відомими характеристиками об'єкта нерухомості (ознаками) повертає числову оцінку – очікувану ринкову вартість. Формально модель має відображати відображення типу:

$$f: X \rightarrow y,$$

де X – простір вхідних ознак;

$y \in R_+$ – шукане значення вартості об'єкта в грошовому еквіваленті.

Кожен об'єкт описується вектором:

$$x = (x_1, x_2, \dots, x_n),$$

де x_i – окрема характеристика: площа, кількість кімнат, рік побудови, місто, район, тип об'єкта тощо.

Частина з цих ознак – числові, частина – категоріальні. Нехай $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^m$ – навчальна вибірка об'єктів з відомою вартістю. Під час тренування модель отримує доступ до повного набору X та відповідного значення y для кожного об'єкта.

Через суттєву варіативність цін і наявність викидів у реальних оголошеннях, застосовується логарифмічне перетворення цільової змінної (3.1):

$$\tilde{y}^i = \log(1 + y^{(i)}), \quad (3.1)$$

де $y^{(i)}$ – реальна вартість об'єкта для i -го прикладу (у гривнях);

$\tilde{y}^i$ – логарифмована цільова змінна (те, що подається на вхід моделі);

$\log(1 + y)$ – логарифмічне перетворення для згладжування великих значень.

Це дозволяє зменшити вплив поодиноких дуже дорогих об'єктів, нормалізувати розподіл та покращити стабільність навчання. Метою моделі стає знаходження функції $\hat{f}(x)$, яка мінімізує середню абсолютну похибку у лог-просторі (3.2):

$$L = \frac{1}{m} \sum_{i=1}^m |\hat{f}(x^{(i)}) - \tilde{y}^{(i)}|, \quad (3.2)$$

де L – загальна функція втрат, яка мінімізує модель;

m – кількість прикладів у навчальній вибірці;

$x^{(i)}$ – вектор ознак i -го прикладу;

$\hat{f}(x^{(i)})$ – прогноз логарифмованої вартості, який видає модель;

$\tilde{y}^{(i)}$ – логарифмована реальна ціна.

Після завершення навчання модель повертає оцінку $\hat{y} = \exp(\hat{f}(x)) - 1$, яка наближено відповідає реальній ринковій ціні.

У рамках цього підходу застосовується комбінація ознак різних типів:

- числові: площа об'єкта, кількість кімнат, рік побудови, дні на ринку;
- бінарні: наявність пропущеного значення року (year_missing);
- категоріальні: тип об'єкта, місто, район.

Категоріальні ознаки перетворюються на числові через кодування (one-hot), у результаті чого кожен об'єкт описується у високовимірному, але лінійно оброблюваному просторі.

Окрім прямого прогнозу значення, система також формує текстову оцінку тренду попиту. Для цього виконується порівняння фактичної вартості y_{real} із прогнозованою $\hat{y}$, після чого обчислюється відносне відхилення (3.3):

$$\delta = \frac{\hat{y} - y_{real}}{y_{real}}, \quad (3.3)$$

де $\tilde{y}^{(i)}$ – прогнозована ціна;

y_{real} – фактична ціна об'єкта;

δ – відносне відхилення між прогнозом і реальністю.

Для оцінки точності моделі після навчання використовується: MAE (Mean Absolute Error) у гривнях (3.4):

$$MAE = \frac{1}{n} \sum_{i=1}^n |y^{(i)} - \hat{y}^{(i)}|, \quad (3.4)$$

де n – кількість прикладів у вибірці;

$y^{(i)}$ – реальна ціна i -го об'єкта;

$\hat{y}^{(i)}$ – прогнозована ціна.

Коефіцієнт детермінації R^2 , що вимірює, наскільки добре модель пояснює дисперсію в реальних даних (3.5):

$$R^2 = 1 - \frac{\sum_{i=1}^n (y^{(i)} - \hat{y}^{(i)})^2}{\sum_{i=1}^n (y^{(i)} - \bar{y})^2}, \quad (3.5)$$

де $\bar{y}$ – середнє значення реальних цін.

Ці метрики дозволяють інтерпретувати якість не тільки в абсолютному значенні, а й порівняно з наївними чи лінійними моделями.

Описане формулювання дозволяє будувати гнучку систему прогнозу, яка здатна адаптуватися до змін у поведінці користувачів, появи нових типів об'єктів або змін у розподілі цін по регіонах. Застосування логарифмічного перетворення та МАЕ як функції втрат дозволяє досягнути стабільного навчання навіть за наявності аномалій і нерівномірного розподілу цільової змінної.

3.3 Обґрунтування методу розв'язання

Для розв'язання задачі прогнозування попиту на ринку нерухомості обрана модель градієнтного бустингу XGBoost, яка добре підходить для роботи з табличними даними. Основною перевагою такого підходу є здатність враховувати складні нелінійні взаємозв'язки між параметрами, стійкість до вибірових шумів, а також висока ефективність у задачах, де частина ознак може бути відсутньою або суттєво варіюватися.

XGBoost працює на основі ансамблю дерев рішень, які навчаються послідовно. Кожне нове дерево будується таким чином, щоб мінімізувати помилки попередніх. У результаті формується сильна модель, яка поєднує рішення великої кількості слабших моделей. Цей підхід дозволяє отримати високу точність без необхідності складної інженерії ознак. Ще однією важливою властивістю XGBoost є ефективність при великих обсягах даних – модель швидко навчається, добре масштабується та має гнучку систему налаштування параметрів.

Під час навчання прогнозової моделі система розділяє дані на підвибірки за джерелом походження – Lun та Zillow. Для кожного з джерел створюється окрема модель, що дозволяє краще адаптуватися до особливостей форматів і цінових діапазонів. Вибіркова сегментація гарантує більшу точність у межах конкретного джерела, уникнення зміщення та некоректного перенесення закономірностей між суттєво різними вибірками.

Навчання моделі відбувається з використанням пайплайну, який складається з двох основних етапів: попередньої обробки ознак і самої моделі XGBRegressor. Для числових ознак (площа, кількість кімнат, рік побудови тощо) застосовується

					IC12.300БАК.005 ПЗ	Арк.
						24
Зм.	Лист	№ докум.	Підпис	Дата		

масштабування та заповнення пропущених значень медіаною. Для категоріальних змінних (тип об'єкта, місто, регіон) використовується one-hot кодування з ігноруванням невідомих значень. Такий підхід дозволяє забезпечити стабільність роботи навіть при нестандартних чи нових значеннях у даних.

Окремий акцент на підборі гіперпараметрів моделі. Для цього застосовується RandomizedSearchCV, який шукає оптимальну комбінацію параметрів серед заданих діапазонів. Серед параметрів, що варіюються: кількість дерев (`n_estimators`), максимальна глибина (`max_depth`), швидкість навчання (`learning_rate`), а також параметри вибірки ознак та об'єктів (`colsample_bytree`, `subsample`). Такий підбір дозволяє збалансувати точність моделі, уникнути перенавчання та забезпечити хорошу узагальнюваність результатів.

У разі наявності достатньої кількості регіонів у вибірці застосовується стратегія крос-валідації з групуванням (`GroupKFold`), де роль групи відіграє регіон розміщення об'єкта. Це дозволяє оцінювати модель у більш реалістичних умовах – не тільки за окремими об'єктами, але й з урахуванням географічної розбіжності. У випадках, коли кількість регіонів недостатня, використовується проста крос-валідація з трьома фолдами.

Моделю оптимізується за метрикою середньої абсолютної помилки (MAE), яка дозволяє інтерпретувати похибку у грошових одиницях, зрозумілих для кінцевого користувача. Після завершення тренування виконується оцінка якості на тестовій вибірці, а також обчислення коефіцієнта детермінації (R^2), який показує ступінь відповідності прогнозу до реальних значень.

Вибір XGBoost як основного алгоритму прогнозування зумовлений кількома факторами:

- підтримка роботи з великою кількістю вхідних ознак, включаючи пропущені значення;
- гнучка система налаштувань;
- висока точність при відносно короткому часі навчання;
- можливість пояснення отриманого результату через важливість ознак (feature importance);

- стійкість до колінеарності між ознаками.

Порівняно з іншими поширеними методами, такими як лінійна регресія або дерева рішень без ансамблів, XGBoost дає змогу враховувати складну структуру даних, адаптуватися до змішаних джерел і забезпечувати хороші результати на практиці.

3.4 Опис методу розв'язання

У межах реалізації системи обраний метод XGBoost, який ґрунтується на градієнтному бустингу з використанням дерев рішень. Цей підхід дозволяє будувати послідовність моделей, кожна з яких уточнює помилки попередньої, у результаті чого загальний прогноз значно точніший, ніж у випадку класичних дерев або простих регресійних моделей.

Метод використовує числові та категоріальні вхідні параметри, які проходять етапи попередньої обробки: заповнення пропущених значень, масштабування та кодування. До обчислення входять як безпосередні характеристики об'єкта (площа, рік побудови, кількість кімнат), так і контекстуальні ознаки (район, місто, тип житла, час на ринку). Такий набір параметрів дозволяє моделі враховувати як структурні, так і ринкові фактори. Алгоритми роботи:

- збір і поєднання даних – з баз даних lun.db і zillow.db витягується інформація про актуальні оголошення;
- формування додаткових ознак – обчислюється кількість днів, які об'єкт перебуває на ринку (days_on_market), додається змінна year_missing для фіксації відсутнього року побудови, і створюється поле year_built_imputed – заповнене значення року на основі медіани;
- передобробка ознак – для числових змінних застосовується масштабування (StandardScaler), а для категоріальних – кодування через one-hot з обробкою невідомих значень (OneHotEncoder(handle_unknown='ignore'));
- розділення даних – вибірка ділиться на навчальну та тестову у пропорції 80/20;

					ІС12.300БАК.005 ПЗ	Арк.
						26
Зм.	Лист	№ докум.	Підпис	Дата		

- побудова пайплайну – створюється конвеєр з двох частин: передобробка і XGBRegressor;
- налаштування гіперпараметрів – задано діапазони для таких параметрів, як кількість дерев (`n_estimators`), глибина (`max_depth`), коефіцієнт навчання (`learning_rate`), частка підмножини вибірки (`subsample`) і вибірка ознак (`colsample_bytree`);
- крос-валідація – якщо у вибірці є п'ять і більше унікальних регіонів, застосовується GroupKFold, де регіон використовується як група, у разі меншої кількості – використовується звичайна 3-кратна крос-валідація;
- навчання та збереження – після вибору кращої конфігурації модель навчається на всій навчальній вибірці;
- оцінка якості – після навчання модель перевіряється на тестовій вибірці.

Припустимо, в систему надійшов запит на об'єкт: площа – 65 м², кількість кімнат – 2, рік побудови – 2010, місто – Київ, район – Центр, тип житла – квартира, джерело – LUN, дата додавання – 2025-06-10. Після обробки ознак, побудови вектору та подачі його в модель, прогнозована ціна склала близько 1,950,000 грн. Якщо фактична ціна відрізняється менш ніж на 5%, тренд попиту визначається як стабільний, у разі сильного перевищення – як зростаючий.

Для кожного джерел система обчислює MAE і R^2 після навчання. Результати засвідчили стабільну якість у межах прийнятного відхилення, що дозволяє використовувати модель для цілей як одиничного, так і масового прогнозування. Всі результати фіксуються у логах і можуть використовуватись для подальшого порівняння версій моделі.

Висновки до розділу 3

Сформульована задача прогнозування вартості житлової нерухомості на основі структурованих вхідних параметрів. Уточнена змістовна постановку задачі та визначено формат вхідних і вихідних змінних. Пріоритет зосереджений на тому,

					ІС12.300БАК.005 ПЗ	Арк.
						27
Зм.	Лист	№ докум.	Підпис	Дата		

щоб побудова моделі дозволяла працювати з реальними оголошеннями, у яких можуть бути частково відсутні дані або зафіксовані суттєві відхилення від середніх значень.

Для обробки таких випадків система формує адаптивний набір ознак, до якого входять як числові параметри (площа, рік побудови, кількість кімнат), так і категоріальні змінні (тип об'єкта, місто, регіон, джерело). Перед подачею до моделі виконується уніфікація даних та логарифмічне перетворення цільової змінної для згладжування впливу викидів. Такий підхід дозволяє сформувати стійку до шуму та перекосів вибірку, на основі якої проводиться навчання.

Обраний метод XGBoost має низку переваг у контексті табличних структур: підтримка великої кількості ознак, робота з відсутніми значеннями, можливість оптимізації через регуляризацію, швидкість навчання та зручність подальшого масштабування. Саме ці властивості роблять його доцільним для систем, що працюють з динамічними або змішаними джерелами даних.

Навчання виконується окремо для кожного джерела, щоб зберегти внутрішню узгодженість вибірки. Також застосовується стратегія крос-валідації з урахуванням регіональної групи, що дозволяє точніше оцінювати узагальнюючу здатність моделі. У процесі навчання автоматично підбираються оптимальні гіперпараметри за допомогою RandomizedSearchCV, а якість перевіряється за MAE та коефіцієнтом детермінації R^2 .

4 ПРОЄКТУВАННЯ СИСТЕМИ ПРОГНОЗУВАННЯ ПОПИТУ

4.1 Визначення вимог до системи

Система, яка має забезпечити прогнозування попиту на ринку нерухомості, розглядатиметься як приклад простої, зрозумілої та корисної прикладної реалізації, здатної працювати у форматі окремого локального програмного рішення. У центрі уваги перебуває не складність алгоритмів, а логіка, яка дозволяє на основі вихідних параметрів і статистики ринку зробити попередню оцінку ймовірного попиту на житло у конкретному місті, районі або типі забудови. Запропоноване рішення має працювати з уже зібраною інформацією, а також дозволяти оновлювати й доповнювати її в напівавтоматичному режимі.

Функціонал системи умовно поділяється на кілька рівнів: збір і збереження даних, обробка та побудова моделі прогнозу, а також вивід результату у зручному форматі. Кожен з етапів накладає свої вимоги як до технічного забезпечення, так і до логіки взаємодії між модулями. Рішення повинно бути максимально адаптивним до змін у структурі вихідних даних або в потребах кінцевих користувачів.

З технічного погляду, майбутня система може базуватись на типовому стеку Python, SQLite і простому вебфреймворку. Перевагою такого підходу є легкість у розгортанні, мінімальна залежність від зовнішніх сервісів, можливість використання на звичайному комп'ютері без необхідності підключення до хмарної інфраструктури.

Для кращого розуміння логіки вимог до функціоналу сформульована узагальнена таблиця 4.1, яка демонструє основні цілі і конкретні програмні кроки, що мають бути реалізовані.

Таблиця 4.1 – Вимоги до функціоналу

Функціональна вимога	Очікувана реалізація
Збір інформації	Парсинг або підключення до API платформ з оголошеннями про продаж житла (наприклад, LUN.ua)
Зберігання даних	Локальна база даних SQLite зі структурованими таблицями для об'єктів нерухомості

Функціональна вимога	Очікувана реалізація
Попередня обробка	Видалення дублікатів, нормалізація числових значень, переведення категорій у кодовану форму
Побудова прогнозу	Модель лінійної регресії з навчанням на локальній вибірці
Візуалізація	Побудова графіків зміни попиту залежно від параметрів нерухомості
Інтерфейс	Консольний ввід запитів та браузерний вивід результатів через Flask

Серед вхідних параметрів, які планується враховувати у моделі, передбачено тип об'єкта, місто, район, рік будівництва, загальна площа, кількість кімнат, стартова ціна продажу та дата публікації оголошення. За цими характеристиками очікується оцінка інтенсивності звернень до подібних об'єктів.

Оскільки модель працюватиме з відносно невеликими локальними даними, її складність не повинна перевищувати базового рівня. Пріоритетом є швидкість роботи, стабільність та простота налагодження.

Необхідно звернути увагу на організацію структури збереження даних. Таблиця об'єктів нерухомості повинна мати фіксовану кількість полів та бути достатньо гнучкою для подальшого розширення. Програмна частина повинна бути здатна самостійно проводити валідацію отриманих значень та попередню перевірку на відповідність мінімальним критеріям.

Механізм обчислення попиту базуватиметься на навчанні моделі з фіксованим набором параметрів. Для цього передбачено етап тренування моделі на основі реальних оголошень. Після цього буде можливо виконати прогноз для нових або гіпотетичних об'єктів, вводячи відповідні характеристики.

Результати прогнозу виводитимуться у вигляді інтерактивних графіків, що допоможе краще сприймати тренди. Окрім цього, повинна бути реалізована можливість експорту результату у вигляді CSV.

4.2 Архітектура системи

Загальна логіка побудови передбачає модульну архітектуру, що дозволяє відокремити кожен підсистему за функціональністю та спростити підтримку. Окремі модулі відповідатимуть за парсинг інформації з джерел, збереження у базі даних, підготовку даних, моделювання та побудову прогнозів, а також генерацію графічного виводу результатів. Модульна структура також відкриває можливості для масштабування, наприклад, підключення додаткових джерел або заміну алгоритмів прогнозування без втручання в інші частини системи.

Основні компоненти архітектури можна умовно розділити на декілька логічних блоків. Перший – це блок збору даних. У ньому передбачається реалізація скриптів або парсерів, які отримуватимуть інформацію з відкритих джерел. У даному випадку мова йде про платформи нерухомості, такі як LUN.ua і Zillow. Ці сервіси пропонують об'ємну інформацію про житлову нерухомість, що дозволяє формувати вибірку для аналізу. Важливо, що на етапі збору даних уже може проводитися початкова обробка: фільтрація непотрібних полів, нормалізація форматів і перевірка на дублі.

Наступним логічним компонентом виступає база даних. Планується використання реляційної СУБД типу SQLite для зберігання структурованих таблиць з усіма необхідними параметрами об'єктів нерухомості. До основних полів належатимуть: місто, район, тип об'єкта, площа, кількість кімнат, ціна, рік побудови, дата старту продажу. Уся база проєктується з урахуванням можливості агрегування, фільтрації та побудови запитів для передачі даних у модуль прогнозування. Сховище має бути оптимізоване для читання, оскільки запити до бази переважно здійснюватимуться для виводу аналітики та побудови прогнозів.

Модуль обробки даних та підготовки ознак займатиме головну позицію між збереженими сирими записами та моделлю машинного навчання. У цьому блоці виконуватиметься перевірка повноти записів, перетворення категоріальних ознак, масштабування числових значень. Очікується, що саме тут закладатимуться правила фільтрації даних, які не відповідають логіці (наприклад, нульова площа

або ціна). Завдяки цьому модуль зможе забезпечити стабільність подальших прогнозів та зменшити похибку результату.

Модуль прогнозування базуватиметься на лінійній регресії. Обраний алгоритм є достатньо простим і зрозумілим, при цьому забезпечує інтерпретованість результату, що є перевагою для користувачів без глибокої технічної підготовки. Алгоритм будуватиме залежність між вихідними ознаками та цільовим параметром попиту. Важливо, що реалізація моделі не вимагатиме зовнішніх ресурсів або потужного серверного обладнання, що узгоджується із загальною концепцією локального розгортання.

Вебінтерфейс виступатиме головним каналом взаємодії користувача із системою. На основі мікрофреймворку Flask буде реалізована можливість введення параметрів, запуску прогнозу та перегляду результатів у зручному вигляді. Інтерфейс має бути мінімалістичним, з акцентом на графіки, які дозволяють швидко оцінити ситуацію на ринку. Таблиця 4.2 нижче ілюструє загальний поділ обов'язків між модулями.

Таблиця 4.2 – Рольові функції архітектурних модулів

Компонент	Основна функція
Парсер даних	Парсинг або підключення до API платформ з оголошеннями про продаж житла (наприклад, LUN.ua)
Сховище	Локальна база даних SQLite зі структурованими таблицями для об'єктів нерухомості
Модуль обробки	Видалення дублікатів, нормалізація числових значень, переведення категорій у кодовану форму
Прогностичний модуль	Модель лінійної регресії з навчанням на локальній вибірці
Вебінтерфейс	Побудова графіків зміни попиту залежно від параметрів нерухомості

Усі модулі матимуть чітко визначені точки входу та виходу. Передача даних між ними планується через файлову систему або внутрішні функції, залежно від реалізації. Такий підхід забезпечить простоту налагодження та тестування.

З технічної точки зору архітектура відповідатиме концепції клієнт-сервер, але без виділеного серверного компонента. Усі дії виконуватимуться на локальному комп'ютері користувача. Перевага такого рішення – відсутність затримок через мережу, контроль над усіма файлами та можливість запуску на будь-якому середовищі, де встановлений Python. Локальні рішення є зручними для прототипування систем машинного навчання та не потребують складного налаштування безпеки [9].

4.3 Вибір технологій та інструментів

Під час проєктування системи прогнозування попиту виникла потреба у виборі технологій, які б одночасно забезпечували простоту реалізації, стабільність, масштабованість і швидкість розроблення. Обраний стек мав відповідати як технічним, так і практичним потребам – працювати з реальними вхідними даними, підтримувати аналітичні операції й бути гнучким для подальшого розвитку.

Центральною мовою розроблення стала Python. Це рішення зумовлене високою популярністю мови у сфері аналітики, обробки даних і машинного навчання, наявністю великої кількості бібліотек і активної спільноти. Python дозволяє інтегрувати різні модулі, писати компактний і читабельний код, а також розгорнути додатки як локально, так і в хмарному середовищі [10].

Для створення вебінтерфейсу обраний Flask. Це мікрофреймворк на Python, який підходить для реалізації легких застосунків, API та інтерфейсів без надлишкового навантаження [11]. Його структура дозволяє швидко організувати маршрутизацію запитів, обробку форм, генерацію HTML-відповідей через шаблонізатор Jinja2, і підключення до бази через ORM. Завдяки цьому користувач отримує можливість взаємодіяти з системою без встановлення додаткових клієнтів.

					IC12.300БАК.005 ПЗ	Арк.
						33
Зм.	Лист	№ докум.	Підпис	Дата		

Обробка даних у системі базується на використанні бібліотек pandas, NumPy та scikit-learn. pandas дозволяє працювати з табличними даними, агрегувати значення, виконувати сортування, фільтрацію, об'єднання. NumPy забезпечує ефективну роботу з масивами, а scikit-learn – набір інструментів для попередньої обробки, зокрема масштабування, кодування категорій та формування pipeline для подальшої роботи з моделлю.

Для збору інформації з онлайн-ресурсів використовуються інструменти вебскрейпінгу. У проєкті реалізовані окремі парсери на основі фреймворку Scrappy для двох платформ – LUN.ua та Zillow. Scrappy дозволяє організовувати черги запитів, обробку відповіді, побудову пайплайнів для очищення і запису даних [12]. Усі зібрані об'єкти проходять попередню валідацію, після чого зберігаються в базі. Для обробки HTML-структури також використовується lxml.

Для взаємодії з базою даних застосовується SQLAlchemy. Це ORM-бібліотека, яка дозволяє працювати з об'єктами замість сирих SQL-запитів, що значно спрощує код, підвищує його читабельність і дозволяє уникати типових помилок [13]. Через SQLAlchemy визначені моделі для записів, прогнозів, а також завантажених файлів. Перевага використання ORM у тому, що зміни в структурі бази або міграції можна проводити централізовано, без необхідності змінювати SQL вручну в кількох місцях.

Щодо вибору бібліотеки для машинного навчання, то рішення прийнято на користь XGBoost. Цей інструмент побудований на принципах градієнтного бустингу, який є одним із найефективніших способів регресійного моделювання в умовах великої кількості змінних та можливих нелінійних зв'язків [14]. Серед переваг XGBoost – стійкість до перенавчання, підтримка пропущених значень, гнучкість у налаштуванні гіперпараметрів, висока швидкість обчислень навіть на великих наборах даних.

XGBoost не потребує обов'язкового масштабування ознак і добре працює з категоріальними змінними, що особливо важливо у сфері нерухомості, де серед ознак є не лише числові, а й текстові параметри: тип об'єкта, район, місто. Модель дозволяє точно прогнозувати числові значення попиту або очікуваної вартості,

навіть у складних випадках з кореляційними ознаками. Вона часто демонструє кращу точність порівняно з лінійною регресією або класичними деревами рішень, що підтверджується у багатьох практичних застосуваннях [15].

Для зберігання моделі й швидкого її використання після навчання використовується `joblib` – бібліотека, яка дозволяє серіалізувати об'єкти Python з високою швидкістю, що зручно для зберігання конвеєрів даних і моделей. Це дозволяє не навчати модель щоразу при запуску застосунку, а лише завантажувати готовий файл.

Щодо візуалізації, її роль зведена до надання користувачу наочного уявлення про ситуацію на ринку. Для побудови графіків, зокрема лінійних залежностей, гістограм, діаграм тренду, застосовуються `matplotlib` та `seaborn`. Вони легко інтегруються з веб-фреймворком, дозволяють створювати зображення у форматі PNG, які виводяться у браузер.

Структура системи дозволяє обробляти як одиничні запити з форми, так і пакетні прогнози на основі CSV-файлів. Це робить її зручною як для приватних користувачів, так і для невеликих агенцій, які хочуть аналізувати кілька об'єктів одночасно. Такий підхід не потребує складного розгортання й дозволяє використовувати систему навіть на ноутбучі або ПК без підключення до хмарних платформ. В таблиці 4.3 коротко описаний зібраний стек для системи.

Таблиця 4.3 – Технологічний стек для реалізації системи

Компонент	Технологія	Призначення
Мова розроблення	Python	Основна мова для реалізації логіки та ML
Фреймворк вебсервера	Flask	Вебінтерфейс і обробка запитів
ORM	SQLAlchemy	Робота з БД через моделі
Парсинг	Scrapy, lxml	Збір оголошень з сайтів
Обробка даних	Pandas, NumPy	Структурування і підготовка даних

Компонент	Технологія	Призначення
Машинне навчання	XGBoost	Побудова прогностичної моделі
Препроцесинг	scikit-learn	Масштабування, імпутація, кодування
Серіалізація	joblib	Збереження моделі у файл
Візуалізація	seaborn, matplotlib	Побудова графіків і трендів

Така конфігурація дозволяє реалізувати повноцінну інформаційну систему з підтримкою збору, обробки, прогнозування й виведення результатів без потреби у зовнішніх або платних сервісах. Крім того, кожен із вибраних інструментів має активну документацію та підтримується великою кількістю прикладів і шаблонів [16].

Діаграма класів продемонстрована на кресленику IC12.300БАК.005 Д1.

Загальна структура системи продемонстрована на кресленику IC12.300БАК.005 Д2.

4.4 Проєктування сховища даних

Коректне проєктування бази даних є основою стабільної роботи інформаційної системи, особливо в умовах великого обсягу вхідної інформації. У межах системи прогнозування попиту на нерухомість реалізовані три окремі бази даних: lun.db, zillow.db і forecast.db. Такий поділ обраний для зручності обробки джерел, розмежування відповідальностей модулів та уникнення конфліктів при паралельній роботі скриптів.

Перша з баз даних, lun.db, містить оголошення, отримані з української платформи LUN. У структурі таблиці зберігаються всі основні параметри, притаманні об'єктам нерухомості: назва міста, район, площа, кількість кімнат, тип

					IC12.300БАК.005 ПЗ	Арк.
						36
Зм.	Лист	№ докум.	Підпис	Дата		

об'єкта, ціна, рік побудови та дата старту продажу. Таблиця має чітко визначений склад полів, які дозволяють побудувати повну картину щодо розміщених оголошень. Окремо фіксується дата створення запису, що дає змогу аналізувати динаміку появи нових пропозицій на ринку. Всі записи зберігаються у форматі, зручному для подальшого фільтрування й агрегації.

Друга база, `zillow.db`, є аналогічною за структурою до `lun.db`, але з урахуванням специфіки платформи Zillow, яка використовується переважно на ринку США. Основний набір полів схожий, проте деякі з них можуть бути відсутні або заповнені неповністю через відмінності у форматі даних. Наприклад, район або рік побудови об'єкта можуть не вказуватись. Усе це враховано під час побудови моделі, а також при збереженні записів, які проходять первинну перевірку на валідність.

Обидві бази – `lun.db` і `zillow.db` – виконують роль накопичувачів сирих даних. Дані з них використовуються або для тренування моделі, або для аналітики, але не змінюються в процесі роботи користувача. Вони створюються парсерами й оновлюються лише під час нового збору. Це дозволяє зберігати оригінальну інформацію з оголошень та мінімізувати ризики втрати важливих деталей у разі автоматичної обробки.

Третя база, `forecast.db`, є основною у системі. Саме вона забезпечує збереження записів, з якими безпосередньо взаємодіє користувач: результати прогнозу, параметри, передані через форму або CSV-файл, дані про джерело, ідентифікатор запиту, а також технічну інформацію. У базі реалізована таблиця `property`, структура якої наведена у таблиці 4.4.

Таблиця 4.4 – Структура основної таблиці бази даних

Поле	Тип даних	Опис
<code>source</code>	<code>varchar(20)</code>	Джерело даних
<code>city</code>	<code>varchar(100)</code>	Місто
<code>region</code>	<code>varchar(100)</code>	Район або мікрорайон
<code>property_type</code>	<code>varchar(100)</code>	Тип нерухомості
<code>area</code>	<code>float</code>	Площа, м ²

Поле	Тип даних	Опис
rooms	int	Кількість кімнат
price	float	Ціна
year_built	int	Рік будівництва
sale_start_date	text	Дата початку продажу
prediction	float	Результат прогнозу
request_id	int	Ідентифікатор запиту

Така структура дозволяє зберігати не лише оригінальні параметри об'єкта нерухомості, а й результат прогнозу, що розрахований на основі цих даних. Крім того, поле request_id дозволяє об'єднувати кілька записів у межах одного запиту, наприклад, коли завантажується CSV-файл з декількома об'єктами. Це полегшує навігацію історією звернень і дає змогу формувати звіти за певними сесіями.

Особливістю проєктування бази forecast.db стало врахування сценарію використання системи – зокрема, можливість багаторазового повторення прогнозу на основі нових даних, збереження кожного запиту та прив'язки до джерела. У полі source зазначається, з якої платформи отримані дані: LUN, Zillow або введені вручну. Це дозволяє в майбутньому аналізувати точність прогнозів залежно від походження записів.

Під час проєктування баз використовувалися принципи нормалізації даних: кожна таблиця відповідає окремому типу інформації, відсутнє дублювання, що знижує ризик суперечностей. Дані організовані у форматі, придатному як для оперативного аналізу, так і для глибокої аналітики. Структура з урахуванням потенційного розширення дозволяє безболісно додавати нові поля або таблиці без порушення логіки роботи.

Операції над базою даних реалізовані через ORM-рівень SQLAlchemy, що дозволяє уникнути ручного написання SQL-запитів. Завдяки цьому легко будуються фільтри, сортування, пошук за запитом, а також зберігаються нові записи після прогнозу. Робота з базою не залежить від конкретної реалізації, що дозволяє легко перейти на іншу СУБД у разі потреби.

Висновки до розділу 4

Архітектура системи побудована на модульному підході. Це означає, що кожен компонент функціонує окремо: один відповідає за збір даних, інший за передобробку, наступний – за запуск моделі, а ще один – за формування відповіді. Такий підхід дозволяє легко замінювати або допрацьовувати окремі блоки без втрати стабільності роботи всієї системи. Наприклад, можна окремо протестувати парсер Lun, не зачіпаючи роботу прогнозного модуля.

Розроблена структура враховує і технічні обмеження, і можливості масштабування. Завдяки обраному стеку забезпечується швидке розроблення, висока гнучкість, прозоре налаштування моделей і проста інтеграція з базами даних. Всі компоненти легко розгортаються локально й не потребують складного серверного оточення.

Першочергове завдання приділене проєктуванню сховища даних. Створені окремі бази для платформ Lun і Zillow, а також центральну базу для взаємодії з моделлю прогнозу та збереження результатів. Такий розподіл дозволяє уникнути конфліктів, підвищити продуктивність і логічно відокремити зовнішні джерела від внутрішніх процесів. База має зручну структуру таблиць, де кожна виконує чітко окреслену роль – від зберігання сирових даних до фіксації історії прогнозів.

На цьому етапі також сформоване уявлення про типову взаємодію користувача з системою. Передбачені два основні сценарії: одиничний запит і пакетна обробка. У першому випадку користувач вводить параметри вручну, у другому – завантажує файл із великою кількістю об'єктів. Обидва сценарії мають зручний формат виводу й орієнтовані на результат, який легко прочитати та використати для прийняття рішень.

5 РОБОТА З ВЕЛИКИМИ ДАНИМИ

5.1 Методи аналізу великих даних

5.1.1 Машинне навчання для прогнозування попиту

У контексті прогнозування попиту на ринку нерухомості машинне навчання посідає головне місце серед інструментів, що дозволяють працювати з великою кількістю змінних і враховувати як числові, так і категоріальні фактори. Попит не можна визначити простою лінією чи стабільною формулою, оскільки на нього впливають різні обставини: коливання цін, інфляція, інфраструктура, динаміка будівництва, міграція населення, економічні потрясіння, зміни в законодавстві. Усе це створює поле для роботи саме тих алгоритмів, які здатні адаптуватися до складних багатовимірних патернів.

Машинне навчання працює не за чіткими правилами, як традиційне програмування, а шляхом виведення закономірностей із даних. Алгоритм аналізує попередні приклади, визначає приховані зв'язки між параметрами й використовує ці залежності для передбачення майбутніх значень. При цьому модель не просто повторює тренди, вона намагається виявити глибинні зв'язки між факторами. Наприклад, якщо у певному мікрорайоні попит зростає за умов поєднання невеликої площі, близькості до метро й нової інфраструктури, така залежність може бути виявлена навіть без прямого зазначення цих умов.

Один з основних етапів перед навчанням – підготовка даних. Сюди входить нормалізація числових параметрів, кодування категорій, обробка відсутніх значень, виявлення викидів. На ринку нерухомості зустрічаються як дуже типові, так і аномальні об'єкти. Наприклад, квартира з шістьма кімнатами у промисловому районі за низькою ціною або ж студія у новобудові, що коштує як трикімнатна. Такі приклади можуть спотворити результат, тому моделі треба «вчити» правильно реагувати на подібні випадки.

Найпоширенішими алгоритмами для прогнозування попиту є регресійні моделі. Наприклад, лінійна регресія дозволяє передбачити числовий показник (ймовірну кількість заявок або зміну вартості) на основі заданих вхідних

					ІС12.300БАК.005 ПЗ	Арк.
						40
Зм.	Лист	№ докум.	Підпис	Дата		

параметрів: площі, кількості кімнат, року побудови, типу об'єкта тощо. Проте лінійна модель має обмеження: вона припускає, що зв'язок між змінними є прямолінійним, що не завжди відповідає реальності.

У таких випадках доцільно використовувати моделі з більшою здатністю до узагальнення, наприклад, дерева рішень або ансамблеві методи. Один із найпотужніших підходів – градієнтний бустинг (наприклад, LightGBM або CatBoost), що дозволяє досягати високої точності навіть на даних із великою кількістю категорій. Застосування подібних методів у сфері нерухомості вже дало хороші результати у закордонних системах на кшталт Zillow, де модель оцінює не лише поточну вартість об'єкта, а й очікуване зростання або спад за регіоном [17].

Оцінка якості моделі проводиться за допомогою метрик: середньої абсолютної похибки (MAE) та коефіцієнта детермінації (R^2). Кожна з цих метрик дає змогу побачити, наскільки добре модель справляється зі своєю задачею. Якщо модель надто сильно узагальнює, тоді вона недооцінює або переоцінює об'єкти; якщо модель перенавчена, то вона запам'ятовує дані, але погано працює на нових прикладах. Оптимальний варіант – знайти баланс.

Для зображення процесу можна використати рисунок 5.1, що ілюструє типовий цикл роботи моделі машинного навчання в контексті прогнозу попиту:

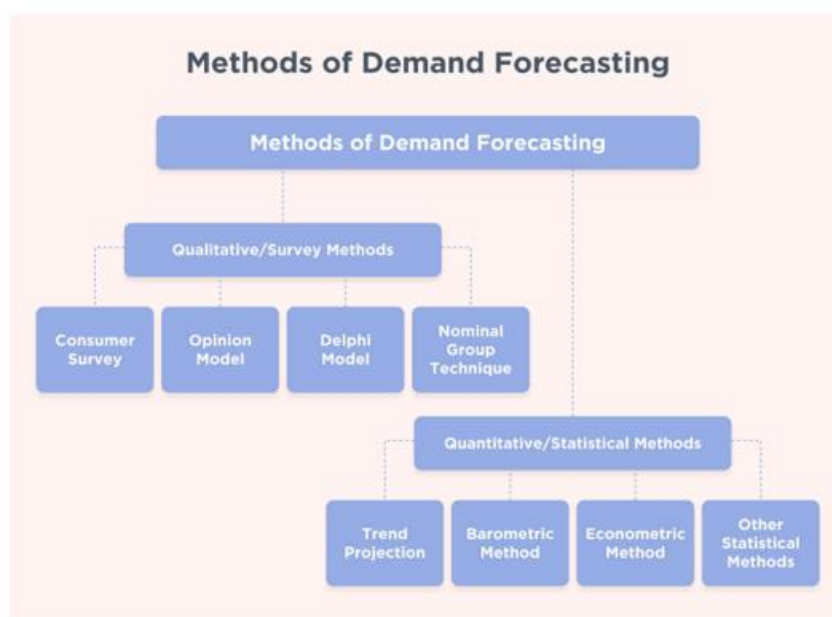


Рисунок 5.1 – Цикл навчання ML моделі [18]

Цей цикл повторюється при надходженні нової інформації, що забезпечує адаптивність моделі до змін у ринковому середовищі. Машинне навчання у цій сфері стає не просто інструментом, а фактично ядром аналітики, навколо якого вибудовується вся логіка системи.

Можливість прогнозувати попит із врахуванням багатьох чинників дозволяє створювати точніші, адаптивніші рішення, що не залежать від інтуїтивних оцінок чи ручного аналізу. Саме в цьому й полягає одна з головних переваг використання машинного навчання у моделюванні ринку нерухомості [19].

5.1.2 Методи кластеризації та класифікації даних

Методи кластеризації та класифікації використовуються у багатьох задачах аналізу великих обсягів даних, зокрема коли потрібно виявити приховані структури або зробити передбачення на основі вже відомих прикладів [20]. У контексті прогнозування попиту на ринку нерухомості ці підходи можуть відігравати допоміжну роль на етапах обробки, фільтрації та початкової сегментації даних. Це дозволяє не лише краще підготувати дані до побудови моделі, а й краще зрозуміти логіку поведінки ринку.

Кластеризація працює без наявності міток або категорій. Вона групує об'єкти на основі подібності характеристик, наприклад таких як площа, ціна, кількість кімнат або рік побудови. У випадку з ринком нерухомості кластеризація дозволяє знайти типові сегменти житла або виявити райони, де параметри об'єктів мають спільні риси. Це може допомогти сформувати цільові підмножини даних, які краще описують специфічну частину ринку, наприклад окремі цінові ніші або типові запити клієнтів. Алгоритми на кшталт K-середніх чи DBSCAN дозволяють знайти ці сегменти навіть у разі великого обсягу вхідної інформації.

У випадку методів класифікації ситуація дещо інша, адже йдеться про вже наявні категорії. Наприклад, можна класифікувати об'єкти як такі, що мають високий, середній або низький попит. Ця інформація може бути корисною для попереднього маркування записів перед тренуванням регресійної або

прогностичної моделі. У такому разі застосовуються алгоритми, що будують рішення на основі навчального набору – дерева рішень, метод опорних векторів або логістична регресія. Вони навчаються на позначених прикладах і далі використовуються для передбачення категорії нових об’єктів.

Крім класичних алгоритмів, для обробки великих даних все частіше застосовуються методи ансамблю – коли кілька моделей поєднуються для отримання більш надійного результату. Відомі приклади – Random Forest або Gradient Boosting. Такі підходи зменшують ризик переобучення, підвищують точність класифікації та краще адаптуються до складної структури вхідних даних. У випадку з ринком нерухомості вони допомагають правильно класифікувати об’єкти навіть у разі шуму або неструктурованості записів.

Важливо розуміти, що методи класифікації не завжди використовуються для побудови фінального прогнозу [21]. Часто вони відіграють допоміжну роль, наприклад, у визначенні пріоритетних сегментів ринку для подальшого аналізу або в автоматизованій фільтрації неякісних записів. На практиці саме поєднання кількох підходів дає найбільшу ефективність – кластеризація спрощує простір, а класифікація дозволяє зробити початкову оцінку об’єкта.

Обидва класи методів потребують ретельної підготовки даних. Важливо привести числові значення до одного масштабу, закодувати категоріальні ознаки, видалити дублікати та аномалії. Особливу роль у цьому процесі відіграє аналіз кореляцій між ознаками, що дозволяє уникнути надмірної залежності однієї змінної від іншої. Правильна підготовка даних істотно впливає на якість кластеризації та точність класифікації.

Ще однією перевагою обох підходів є можливість візуалізації результатів. Наприклад, результат кластеризації часто можна показати на діаграмі розсіювання, де об’єкти різних кластерів відображаються різними кольорами. Це значно спрощує аналіз і дозволяє швидко виявити закономірності. У свою чергу, результати класифікації легко представляються у вигляді таблиці з передбаченими категоріями для кожного об’єкта, що зручно для перегляду та формування звітів.

З технічного погляду всі ці методи мають підтримку у популярних Python-бібліотеках. Зокрема, для кластеризації та класифікації широко використовуються scikit-learn, XGBoost або CatBoost. Вони надають зручні інтерфейси для побудови моделей, валідації та тестування якості. Для обробки вхідних даних активно застосовується pandas і NumPy, а візуалізацію забезпечують matplotlib або seaborn.

Щоб забезпечити коректність результатів, класифікаційні моделі перевіряються на валідаційному наборі, а точність оцінюється за метриками точності, повноти та F1-міри. У випадку кластеризації важливими показниками вважаються середня відстань між кластерами або показники щільності. На основі таких критеріїв приймається рішення про доцільність подальшого використання моделей або про необхідність їх доопрацювання.

У контексті прогнозування попиту на нерухомість важливо не лише побудувати модель, а й правильно підготувати вхідні дані. Саме кластеризація та класифікація допомагають зробити перший крок до впорядкування інформації, виділення основних закономірностей і створення логічної структури, яка згодом використовується в основній моделі прогнозу.

5.2 Алгоритми прогнозування

Алгоритм XGBoost став центральним елементом прогнозовної частини системи, яка має забезпечити ефективне передбачення попиту на нерухомість. Його обрання базувалося не лише на популярності у спільноті практиків, а й на здатності адаптуватися до особливостей ринку, де дані мають різну природу, неповні або нерівномірно розподілені. XGBoost, як реалізація градієнтного бустингу, дозволяє будувати ансамбль рішень на основі простих дерев, кожне з яких послідовно мінімізує похибку попереднього.

Особливістю алгоритму є висока гнучкість у роботі з табличними наборами, здатність ефективно обробляти як числові, так і категоріальні змінні, і що важливо для задачі прогнозування, XGBoost не вимагає однакових масштабів для всіх ознак. Це дозволяє подавати до моделі різномірні характеристики об'єктів: площу,

					ІС12.300БАК.005 ПЗ	Арк.
						44
Зм.	Лист	№ докум.	Підпис	Дата		

кількість кімнат, ціну, тип нерухомості тощо, не турбуючись про обов'язкове попереднє масштабування. Алгоритм вміє самостійно ігнорувати пропущені значення, що робить його придатним для використання з реальними даними, де не всі поля завжди заповнені.

На рівні системи XGBoost інтегрований через окремий модуль прогнозування, який отримує дані з бази, виконує трансформацію вхідних параметрів у формат, придатний до моделі, й обробляє результат. Перед цим кожен об'єкт нерухомості фіксується в таблиці `property`, де вказані основні характеристики: місто, район, площа, тип, кількість кімнат, рік побудови, початок продажу. Також фіксується джерело (`lun`, `zillow` або `вручну`) та ідентифікатор запиту, що дозволяє групувати об'єкти в межах одного звернення користувача.

Усі ці дані передаються до модуля обробки, де відбувається базова підготовка: перетворення категорій у числові індекси, очищення некоректних значень і формування матриці ознак. На цьому етапі вже починає працювати логіка моделі. Ознаки потрапляють у внутрішню структуру XGBoost – спеціальний формат `DMatrix`, який забезпечує оптимізований розрахунок. Кожен об'єкт проходить послідовність перевірок у межах дерев регресії. Умови типу "`ціна < 50000`" або "`кількість кімнат > 2`" формують вузли, які розділяють потік даних у потрібному напрямку. Усі дерева ансамблю поступово наближають прогноз до реального значення, і на виході формується числова оцінка попиту.

Ще однією перевагою XGBoost у межах системи стало те, що він легко серіалізується, модель після навчання зберігається у вигляді файлу та може бути швидко завантажена під час запуску сервера. Це дозволяє уникнути повторного тренування при кожному зверненні. Водночас така схема забезпечує стабільність результату й однаковість прогнозу для одного й того самого набору параметрів.

У процесі прогнозування модель використовує лише ті ознаки, які мають найбільший вплив. Це дозволяє уникнути перенавантаження зайвою інформацією й зосередитися на найбільш значущих характеристиках. Наприклад, за спостереженням, особливий вплив мають площа, тип нерухомості, місто й ціна. Завдяки гнучкості дерев, система здатна фіксувати навіть складні взаємозв'язки,

					ІС12.300БАК.005 ПЗ	Арк.
						45
Зм.	Лист	№ докум.	Підпис	Дата		

наприклад, якщо в одній зоні великі квартири дешевшають, а в іншій, навпаки, зростають у ціні, XGBoost зможе це врахувати, оскільки він працює не з середніми значеннями, а з ієрархією умов.

Процес збору даних, з якими працює модель, реалізований через дві бази (lun.db і zillow.db). Вони містять реальні оголошення з онлайн-платформ, і з кожним новим запуском парсера база оновлюється. Дані зберігаються у структурованому вигляді та проходять валідацію, вносяться лише ті записи, де присутні всі необхідні значення.

Варто окремо зазначити, що система передбачає обробку як одиночних запитів, так і пакетних. У кожному випадку дані проходять однакову обробку, модель застосовується для кожного об'єкта, а результати групуються за request_id.

Для полегшення сприйняття результатів, інтерфейс системи подає прогноз не лише як число, а й у вигляді графіка. Це реалізовано на стороні Flask-серверу. Така подача дозволяє не лише побачити оцінку попиту, а й прослідкувати, як змінюється поведінка системи залежно від різних параметрів. Наприклад, якщо змінити площу або тип об'єкта – прогноз зміниться відповідно до логіки, закладеної у моделі.

На рисунку 5.2 можна побачити приклад дерева рішень, яке є одним з елементів ансамблю у XGBoost:

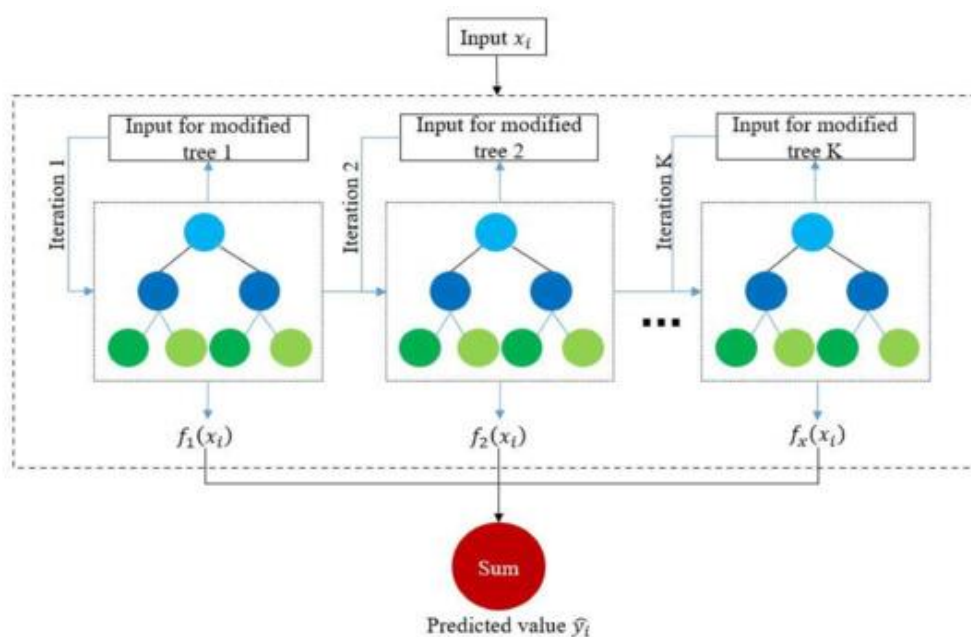


Рисунок 5.2 – Приклад дерева рішень [22]

5.3 Навчання моделі

Процес навчання моделі у межах дипломного проєкту виконує головну роль у забезпеченні точності прогнозування попиту на об'єкти нерухомості. Його структура складається з кількох послідовних етапів, кожен з яких безпосередньо впливає на якість кінцевого результату. Реалізоване навчання за допомогою алгоритму XGBoost, що вже зарекомендував себе як надійний інструмент у завданнях регресії, особливо для роботи з табличними даними зі змішаними типами ознак.

Для початку сформований тренувальний датасет, основу якого становлять дані з двох окремих джерел: lun.db та zillow.db. Після збору всі об'єкти проходять сувору фільтрацію. Видаляються записи без основних атрибутів, таких як площа, ціна, кількість кімнат, рік будівництва чи тип об'єкта. Це дозволяє уникнути викидів та шуму, що можуть суттєво погіршити якість навчання. Далі здійснюється злиття об'єктів з обох баз у загальну структуру, де додається поле source для маркування джерела.

Усі ознаки поділені на числові та категоріальні. До першої групи належать площа, ціна, кількість кімнат, рік будівництва. Категоріальні представлені містом, районом та типом нерухомості. Числові ознаки проходять етап масштабування. У проєкті обраний підхід нормалізації значень до інтервалу між нулем і одиницею. Це дозволяє моделі уникнути пріоритету тієї ознаки, яка має більший абсолютний діапазон.

Категоріальні ознаки перетворюються у числові за допомогою one-hot кодування. Наприклад, тип об'єкта квартира, будинок, новобудова – перетворюється у вектор з відповідними двійковими значеннями. Враховуючи невеликий розмір словника таких ознак, це не створює навантаження, а навпаки, дозволяє моделі працювати з кожною категорією окремо, не змішуючи їх між собою. Для міста та району також застосоване кодування, але з фільтрацією рідкісних значень.

					ІС12.300БАК.005 ПЗ	Арк.
						47
Зм.	Лист	№ докум.	Підпис	Дата		

Для уникнення переоцінки під час навчання використано схему кросвалідації. Замість класичного розбиття на тренувальну та тестову вибірки реалізований підхід GroupKFold. Його особливість полягає у тому, що об'єкти з одного району завжди потрапляють до однієї з груп, що дозволяє тестувати модель на відмінних від навчання регіонах. Такий підхід імітує реальні умови, коли модель має прогнозувати для нової географії.

У процесі підбору параметрів XGBoost застосований RandomizedSearchCV. Це алгоритм випадкового перебору комбінацій гіперпараметрів з фіксованої множини. Обрані такі параметри: кількість дерев, глибина дерева, темп навчання (learning_rate), регуляризаційні коефіцієнти. Наприклад, глибина дерева варіювалась у межах від 3 до 7, темп навчання від 0.01 до 0.1. Регуляризація (alpha і lambda) дозволила обмежити складність дерева та уникнути переобучення.

Оцінка якості навчання виконувалась за двома метриками: середньою абсолютною помилкою (MAE) та коефіцієнтом детермінації (R^2). Ці показники розраховуються окремо для кожної ітерації кросвалідації. У підсумку вибирається модель з найкращими середніми результатами. Зібрані метрики також дозволяють визначити стабільність моделі та її схильність до коливань залежно від географії даних.

Для запуску навчання створено окремий Python-модуль, де реалізований клас Forecast. Його метод fit відповідає за підготовку даних, запуск GridSearchCV, обробку результатів та збереження найкращої моделі. Збереження відбувається у вигляді pickle-файлу за допомогою бібліотеки joblib. Це дозволяє швидко завантажити модель під час роботи вебсервера без повторного тренування.

Перед запуском навчання кожна ознака проходить оцінку важливості. Ознаки з мінімальним внеском у результат видаляються. Такий підхід дозволяє зменшити розмір матриці ознак та скоротити час навчання. Для XGBoost така оптимізація є доречною, оскільки дерево приймає рішення лише на основі корисної інформації.

Також реалізована логіка формування тестової вибірки, яка повністю не перетинається з тренувальною. Це досягається за рахунок групування за унікальним request_id. Таким чином, жоден об'єкт з однієї сесії не потрапляє до

					ІС12.300БАК.005 ПЗ	Арк.
						48
Зм.	Лист	№ докум.	Підпис	Дата		

навчання і тесту одночасно. Це особливо важливо у випадках, коли модель буде використовуватись для пакетного прогнозування за CSV-файлом.

На етапі завершення навчання модель не лише зберігається, а й логуються її характеристики: дата створення, параметри, показники точності, розмір вхідної вибірки. Усе це дозволяє контролювати версії моделі та швидко знаходити оптимальні конфігурації. У майбутньому така система журналювання може бути розширена до повноцінної моделі версіонування.

Потрібно зауважити, що всі етапи навчання проводяться локально. Це зроблено свідомо, щоб забезпечити контрольованість процесу і можливість швидкого налагодження. Під час оптимізації врахована можливість локального середовища: обмеження оперативної пам'яті, час виконання, кількість доступних ядер. Усі параметри підібрані з урахуванням обмежених ресурсів, але без значної втрати якості результату.

Висновки до розділу 5

Даний розділ формує уявлення про технічну частину побудови системи. Кожен етап описаний з урахуванням конкретних завдань, які виконує система у процесі аналізу. Всі дії зосереджені навколо практичного застосування методів машинного навчання, що працюють із великою кількістю ознак і потребують стабільного, передбачуваного результату.

Для реалізації інтелектуального модуля обраний метод градієнтного бустингу, що базується на поетапному побудуванні дерев рішень. Цей підхід дозволяє працювати з різнорідними типами вхідних параметрів, ефективно обробляти пропущені значення, вловлювати складні залежності та адаптуватися до змін у структурі даних. Замість складного ручного аналізу система формує чисельну оцінку попиту на основі десятків факторів: тип об'єкта, площа, район, рік, джерело, день тижня, взаємозв'язок між ціною й попередніми оголошеннями.

Підготовка даних виконується через послідовне очищення, кодування категоріальних змінних, заповнення порожніх значень і масштабування. Усі ці

					ІС12.300БАК.005 ПЗ	Арк.
						49
Зм.	Лист	№ докум.	Підпис	Дата		

кроки реалізовані у вигляді окремого пайплайну, який легко повторити для нової вибірки.

Під час навчання модель отримує вхідні дані, розбиті на навчальну й тестову частини, з подальшою перевіркою точності за кількома метриками – середньою абсолютною помилкою та коефіцієнтом детермінації. Кожна ітерація тренування зберігається у вигляді файлу моделі, що містить параметри, список ознак, дату створення й інші службові поля.

Навчена модель працює у зв'язці з базою даних та вебінтерфейсом. Після запуску прогнозу вона повертає числове значення, яке далі перетворюється на готовий результат – прогнозовану ціну. Одночасно система виводить пояснення у вигляді тренду попиту, визначеного на основі динаміки в регіоні. Це дозволяє розширити функціональність і сформулювати короткий аналітичний опис замість сухої цифри.

Уся логіка побудована так, щоб легко оновлювати модель у майбутньому. За потреби можна підключити нові дані, перебудувати вибірку, переобчислити ознаки або змінити сам алгоритм. Завдяки модульному підходу до реалізації система готова до масштабування – як за обсягом інформації, так і за складністю прогновної задачі. Вибір інструментів та їхня конфігурація дозволяє забезпечити точність, швидкість і простоту обслуговування без втрати якості.

У результаті сформована надійна основа для прогнозування попиту, яка поєднує сучасні алгоритми, автоматизовану обробку й гнучкість у роботі з реальними даними. Така структура відкриває можливість не лише для розрахунку попиту, а й для впровадження додаткових сценаріїв використання моделі у суміжних напрямках.

					ІС12.300БАК.005 ПЗ	Арк.
						50
Зм.	Лист	№ докум.	Підпис	Дата		

6 РЕАЛІЗАЦІЯ СИСТЕМИ ПРОГНОЗУВАННЯ ПОПИТУ

6.1 Розроблення MVP

Початковим етапом створення системи стало розроблення MVP, який мав включати базовий функціонал для тестування головної ідеї: прогнозування попиту на житлову нерухомість. Такий підхід дозволив сфокусуватись не на повноцінній платформі з десятками розділів, а на базових сценаріях роботи, які справді мають значення для користувача. Архітектура MVP об'єднала декілька компонентів: модулі обробки даних, вебінтерфейс, систему завантаження CSV-файлів і модуль відображення результатів.

Акцент в розробленій системі зосереджений на зручності взаємодії. Користувач має змогу обрати між двома режимами роботи: заповненням форми з характеристиками окремого об'єкта або завантаженням цілого набору даних у вигляді CSV-файлу. Усі необхідні поля (місто, регіон, площа, кількість кімнат, тип об'єкта, рік побудови, дата публікації, фактична ціна, джерело) розміщені у формі, доступній на головній сторінці. Після заповнення параметрів система обробляє введені значення, запускає прогноз і виводить результат у зрозумілому вигляді.

Для реалізації клієнтської частини використані шаблони HTML з вбудованими Jinja2-блоками, які динамічно підставляють значення в залежності від сценарію. Головна сторінка представлена шаблоном index.html. Вона містить навігацію, форму вводу, посилання на завантаження файлів і трендову аналітику. Після натискання кнопки "Прогнозувати" форма надсилає POST-запит, який обробляється сервером через Flask.

Модуль Flask обробляє запит, перевіряє вхідні значення, формує об'єкт у базі та передає дані в модуль машинного навчання. Після розрахунку результат передається назад у шаблон result.html. На цій сторінці виводяться основні характеристики об'єкта: прогнозована ціна, тренд попиту та кнопка для повернення до нової форми. Все це реалізовано з мінімальним навантаженням, що дозволяє швидко отримувати результат навіть на локальному сервері без додаткової оптимізації.

					ІС12.300БАК.005 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		51

Окрім одиничного прогнозу, реалізована і пакетна обробка. Шаблон `upload.html` дозволяє завантажити CSV-файл з даними по об'єктах нерухомості. Форма приймає файл, перевіряє його структуру та передає на обробку. Сервер читає кожен рядок, створює об'єкт, передає його в модель і записує результат у базу даних разом з унікальним `request_id`. Після завершення обробки всі результати групуються і виводяться на сторінці `batch_results.html` у вигляді таблиці, де кожен рядок містить ID, характеристики та прогнозовану вартість об'єкта.

Ще однією важливою частиною системи стала візуалізація аналітики. Шаблон `trend.html` реалізує простий механізм побудови графіків. Користувач може обрати місто або регіон, система формує відповідний запит, обробляє дані, генерує зображення і виводить його на сторінці. Цей підхід дозволяє оцінити динаміку змін цін та попиту в різних частинах країни. Графік відображається як статичне зображення, що зберігається у папці `static` і оновлюється при кожному новому запиті.

Серверна логіка реалізована на Flask, з використанням маршрутів, що відповідають кожному типу запиту. Наприклад, маршрут `/` відповідає за головну сторінку, `/upload` – за CSV-файл, `/trend` – за візуалізацію графіка, а окремий маршрут формує сторінку з таблицею пакетних результатів. Усі запити обробляються асинхронно, без затримки в роботі інших частин системи.

Збереження об'єктів і прогнозів відбувається у відповідні таблиці БД. Вся інформація пов'язується через ідентифікатори: кожен об'єкт має свій `id`, кожен запит – `request_id`. Це дозволяє легко витягувати пов'язані дані і аналізувати результати, навіть якщо користувач працює з великим CSV-файлом.

Алгоритм обробки запиту на прогноз продемонстрована на кресленику IC12.300БАК.005 ДЗ.

6.2 Демонстрація роботи системи

Реалізована система включає вебінтерфейс, який дає змогу працювати з окремими оголошеннями або з файлами у форматі CSV. Уся логіка взаємодії

					IC12.300БАК.005 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		52

побудована навколо спрощеного сценарію використання: користувач обирає місто, вводить характеристики об'єкта, система формує прогноз цінової вартості та виводить результат разом із трендом попиту.

Після запуску системи у браузері відкривається головна сторінка, яка дозволяє одразу перейти до ручного введення даних або пакетного прогнозування. Вручну можна ввести такі параметри: місто, район, площа, кількість кімнат, рік побудови, дата додавання, тип об'єкта, джерело оголошення та фактична ціна. На рисунку 6.1 видно приклад заповненої форми для об'єкта в Дарницькому районі Києва.

The screenshot shows a web browser window with the address bar displaying 'http://127.0.0.1:5000'. Below the address bar, there are three links: 'Головна', 'Пакетне завантаження', and 'Тренд'. The main heading is 'Введіть параметри об'єкта'. The form contains the following fields: 'Місто:' with a dropdown menu showing 'Київ'; 'Регіон:' with a dropdown menu showing 'Дарницький'; 'Площа (м²):' with a text input field containing '52'; 'Кімнат:' with a text input field containing '2'; 'Рік побудови:' with a text input field containing '1976'; 'Дата додавання (YYYY-MM-DD):' with a text input field containing '10.06.2025' and a calendar icon; 'Тип об'єкта:' with a text input field containing 'flat'; 'Джерело:' with a dropdown menu showing 'LUN'; 'Фактична ціна:' with a text input field containing '55391'. Below the form, there is a button labeled 'Прогнозувати' and a link labeled 'Завантажити CSV'.

Рисунок 6.1 – Головний екран системи

Для кожного введенного оголошення формується прогнозована вартість, яку модель обчислює на основі навченої логіки. У результаті також виводиться текстовий опис попиту: він може бути зростаючим, стабільним або таким, що знижується. Це дозволяє користувачеві не просто побачити ціну, а й оцінити ринкову ситуацію. Приклад результату показаний на рисунку 6.2: видно як введені характеристики, так і прогнозовану ціну та поточний тренд.

					ІС12.300БАК.005 ПЗ	Арк.
						53
Зм.	Лист	№ докум.	Підпис	Дата		



Рисунок 6.2 – Результат роботи системи

Окремо реалізована можливість пакетного прогнозування. Користувач може завантажити CSV-файл, зібраний із платформи Lun або Zillow, вибрати джерело та натиснути кнопку прогнозу. На рисунку 6.3 показаний інтерфейс сторінки завантаження файлу.

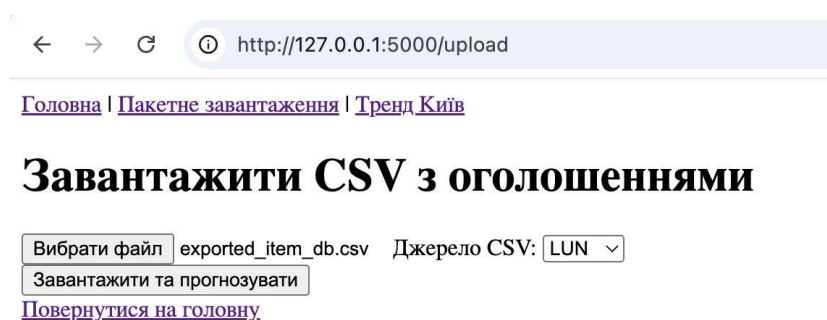


Рисунок 6.3 – Імпорт файлу у систему

Система автоматично обробляє усі записи та виводить таблицю з результатами для кожного об'єкта: місто, район, площа, кількість кімнат, тип, прогнозована ціна та тренд попиту. Рисунок 6.4 ілюструє приклад таблиці для міста

					ІС12.300БАК.005 ПЗ	Арк.
						54
Зм.	Лист	№ докум.	Підпис	Дата		

Київ – видно, що прогноз охоплює дані з кількох районів і дозволяє одразу помітити загальну ринкову картину.

Результати пакетного прогнозу

ID	Місто	Регіон	Площа	Кімнат	Тип	Ціна прогнозу (грн)	Тренд
3585	Київ	Дарницький	33.0	1		51852.92578125	зростає
3586	Київ	Дарницький	77.0	2		122042.8046875	падає
3587	Київ	Печерський	180.0	4		414406.3125	зростає
3588	Київ	Дніпровський	70.0	3		75679.5	падає
3589	Київ	Святошинський	96.1	3		93493.484375	зростає
3590	Київ	Голосіївський	106.0	3		180936.25	падає
3591	Київ	Шевченківський	128.0	3		271214.1875	стабільно
3592	Київ	Солом'янський	78.0	3		141972.765625	падає
3593	Київ	Дарницький	113.0	3		156793.65625	зростає
3594	Київ	Дарницький	106.5	3		154076.703125	зростає
3595	Київ	Шевченківський	45.2	2		63297.52734375	зростає
3596	Київ	Дарницький	90.7	3		141196.765625	зростає
3597	Київ	Печерський	519.1	5		3663003.75	зростає
3598	Київ	Подільський	86.0	3		161002.984375	зростає
3599	Київ	Деснянський	66.4	2		80405.1953125	зростає
3600	Київ	Дніпровський	32.0	1		45754.515625	стабільно
3601	Київ	Дніпровський	30.0	1		41965.88671875	зростає
3602	Київ	Святошинський	37.0	1		67618.2421875	зростає
3603	Київ	Подільський	72.0	2		113592.6328125	зростає
3604	Київ	Шевченківський	100.0	4		130001.578125	зростає

Рисунок 6.4 – Результат прогнозування даних файлу

Система також інтегрує парсери для збору оголошень. Реалізовані два окремих парсери – для Lun.ua і для Zillow. Вони запускаються вручну або за розкладом, звертаються до відповідних API або HTML-структур, отримують нові оголошення, зберігають дані в локальну базу та уникають дублювань. У процесі роботи парсера дані зберігаються через SQLAlchemy.

Усі запити, які проходять через систему, мають збереження у базі, що дозволяє формувати статистику для побудови графіків. На основі цих даних реалізована сторінка перегляду трендів. Користувач може обрати місто та конкретний район і переглянути зміну прогнозованих цін за часом. Це особливо корисно для оцінки динаміки попиту у певних локаціях. Необхідно виділити підхід до тренду попиту. Його визначення виконується на основі аналізу останніх прогнозів: якщо ціна для певного типу об'єкта і району протягом кількох періодів зростає – тренд фіксується як позитивний. Якщо ціни знижуються, то система відзначає падіння. Якщо зміни несуттєві, тоді тренд визначається як «стабільний». Така логіка дозволяє виводити не просто число, а коротке аналітичне резюме.

Рисунок 6.5 демонструє графік для Шевченківського району Києва, де помітно періоди зростання й спаду цін, що відповідає зміні ринкової активності.

Оберіть місто або регіон для графіка

Місто: Регіон:

Графік для Київ

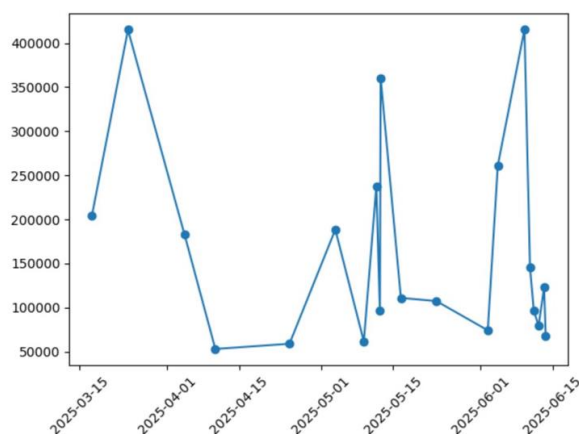


Рисунок 6.5 – Графік зміни попиту

Кожен етап, від введення параметрів до побудови графіка, працює на єдиному сервері, з мінімальними затримками. Навіть при пакетній обробці сотень оголошень час виконання залишається прийнятним. Результати можна експортувати, повторно переглядати або порівнювати з попередніми оцінками.

Діаграма прецедентів продемонстрована на кресленику IC12.300БАК.005 Д4.

6.3 Подальші кроки з розроблення системи

Реалізований MVP створює середовище для взаємодії з системою як у тестовому, так і в практичному режимі. Навіть базовий функціонал покриває основні потреби, від ручного введення до масового прогнозування з CSV-файлів. Інтерфейс побудований таким чином, щоб користувач без технічного бекграунду міг інтуїтивно орієнтуватися у структурі сторінок, вводити дані, отримувати результат і візуалізувати його в зручному форматі. Вивід результатів є максимально чітким, без перевантаження другорядною інформацією.

Система показує стабільну роботу при завантаженні великих файлів і підтримує прогноз для десятків об'єктів одночасно. Формат виводу результатів дозволяє одразу перейти до аналізу попиту у розрізі районів, типів об'єктів або площ. Завдяки цьому навіть простий звіт може стати основою для подальших висновків, що базуються на об'єктивних числових значеннях.

У нинішньому вигляді хоча система й не є привабливою, але вже виконує базові функції прогнозу попиту. Її архітектура передбачає гнучке розширення. Передбачається впровадження автоматичного оновлення моделі за розкладом, підключення додаткових джерел оголошень, урахування погодних умов, інфраструктурних обмежень і подій, які можуть впливати на ринок (як-от зміни законодавства, новини, інфраструктурні проекти). Також у перспективі планується розширення географії: наразі фокус зроблено на Києві, але логіка легко адаптується для Львова, Харкова або будь-якого іншого великого міста.

Окремий напрям розвитку – розширення користувацького інтерфейсу: додавання фільтрів, сортування, розширеної аналітики, можливості обліку, формування звітів і інтеграції з зовнішніми сервісами. Це дозволить зробити систему корисною не лише для індивідуальних користувачів, а й для агентств нерухомості, аналітичних центрів чи міських департаментів планування.

Висновки до розділу 6

Розроблена система демонструє повноцінну реалізацію базового функціоналу, необхідного для прогнозування попиту. При цьому уся логіка роботи побудована так, щоб процес залишався зрозумілим навіть для користувача без глибокої технічної підготовки.

Система підтримує як ручне введення одиничних параметрів, так і обробку великих масивів даних через CSV-файли. Це дозволяє використовувати її в різних сценаріях, від локального аналізу окремого об'єкта до масової оцінки оголошень з онлайн-платформ. Усі етапи прогнозування супроводжуються логічною послідовністю інтерфейсних дій: заповнення параметрів, запуск моделі, отримання

					ІС12.300БАК.005 ПЗ	Арк.
						57
Зм.	Лист	№ докум.	Підпис	Дата		

оцінки ціни, виведення додаткової аналітичної інформації про тренд попиту. Вивід результатів адаптований для подальшої обробки: таблиці, графіки, пояснення, структура збереження історії.

Інтеграція з джерелами даних реалізована у вигляді окремих парсерів, які обробляють інформацію з Lun.ua і Zillow. Отримані записи автоматично очищуються, нормалізуються та записуються в базу даних. Завдяки структурованій побудові таблиць ці дані одразу доступні для подальших розрахунків. Крім того, реалізоване збереження кожного запиту в історії, що дає змогу аналізувати зміну попиту в динаміці. Це відкриває можливості для формування графіків, виявлення змін на ринку та подальшого навчання моделі на більшій вибірці.

Важливо, що система не обмежується лише розрахунком ціни, додатково реалізоване визначення тренду попиту. Для цього враховується динаміка попередніх прогнозів по районах і типах об'єктів. Завдяки цьому користувач може не лише побачити актуальну вартість, а й оцінити загальну ситуацію на ринку: зростає інтерес до локації чи навпаки – спостерігається падіння. Такий підхід підвищує інформативність і дозволяє приймати рішення на основі не лише точкових значень, а й контексту.

Система успішно працює з кількома джерелами, підтримує обробку сотень записів одночасно, демонструє стабільність при навантаженні та виводить дані без затримок. Це свідчить про технічну готовність до розширення. Поточна реалізація вже охоплює функції, які дозволяють використовувати її в умовах реального застосування, наприклад, агентствами нерухомості, аналітиками чи навіть муніципальними структурами, які займаються просторовим плануванням.

З урахуванням того, що архітектура спроектована модульно, подальше розширення системи виглядає цілком реалістичним. Є можливість додавання нових джерел, автоматичного оновлення прогнозної моделі, підключення додаткових аналітичних модулів, врахування сезонності, зміни ринку в залежності від зовнішніх чинників.

					ІС12.300БАК.005 ПЗ	Арк.
						58
Зм.	Лист	№ докум.	Підпис	Дата		

7 ТЕСТУВАННЯ ТА ВЕРИФІКАЦІЯ

7.1 Підходи до тестування програмного забезпечення

Для перевірки працездатності системи прогнозування попиту застосований підхід, який орієнтується не лише на технічну коректність окремих модулів, а й на реальні сценарії використання. Такий підхід дозволяє побачити, як поводить себе програма під час стандартної взаємодії з користувачем, які помилки можуть виникати і наскільки швидко система на них реагує.

Базовою одиницею перевірки обраний сценарій, тобто конкретну ситуацію, у якій користувач взаємодіє з програмою. Кожен сценарій охоплює повний цикл: від введення або завантаження даних до виводу результату на екран або в базу. Така форма тестування наближена до реального використання й дозволяє побачити не лише функціональність окремих компонентів, а й взаємозв'язки між ними. Наприклад, при запуску прогнозу через CSV-файл перевіряється не тільки робота моделі, а й правильність зчитування файлу, обробка векторів ознак, збереження результатів і відображення таблиці з трендами.

Пріоритет зосереджений на перевірці того, як система поводить себе у звичних і граничних умовах: при повному заповненні форми, при відсутності деяких полів, при неправильному форматі даних або великому обсязі вхідної інформації. Окремі сценарії також передбачають ситуації, коли користувач пропускає обов'язкові етапи або виконує дії в неправильному порядку. Це дозволяє перевірити захищеність системи від типових помилок і впевнитися, що вона не зависає, не повертає некоректні значення і не створює дублікати в базі.

Серед протестованих сценаріїв: одиничний запит із вебінтерфейсу, обробка неповних даних, запуск прогнозу без завантаженого файлу, масова обробка об'єктів, перевірка стабільності роботи з базою та вивід графіка на основі збереженої історії. Усі вони покривають основні можливості системи та дозволяють оцінити, наскільки вона готова до використання у реальних умовах.

Кожен тест розроблений таким чином, щоб охопити всі точки взаємодії: інтерфейс, серверну частину, модель, базу даних, механізми кешування та виводу.

					ІС12.300БАК.005 ПЗ	Арк.
						59
Зм.	Лист	№ докум.	Підпис	Дата		

Це дає змогу не тільки знайти помилку, а й точно визначити її джерело. У результаті навіть складні проблеми розв’язуються швидко, без потреби в повному перезапуску системи.

Акцент на тому, як система реагує на нестандартні ситуації. Наприклад, коли користувач намагається побудувати графік без жодного параметра або лише за містом без вибраного району. У таких випадках система має виводити інформативне повідомлення або будувати узагальнений графік, а не повертати помилку або пусту сторінку. Такі сценарії не лише перевіряють технічну надійність, а й формують довіру до системи з боку користувача.

7.2 Тестування сценаріїв

Для перевірки стабільності роботи системи й оцінки її поведінки в реальних умовах сформований набір практичних сценаріїв. Кожен із них відображає конкретну ситуацію, у якій користувач взаємодіє з програмою, використовуючи один або кілька основних функціоналів. Такий підхід дозволяє побачити не тільки технічну коректність, а й те, наскільки зручно та передбачувано поводить ся система під час роботи.

Сценарій 1 – робота з неповними даними. Цей сценарій моделює ситуацію, коли користувач не вказує певні обов’язкові параметри, наприклад рік побудови або площу. У такому випадку система все одно має формувати прогноз, використовуючи доступні ознаки, і виводити результат без помилок. Тест показує, як саме програма реагує на відсутність інформації: чи замінюються порожні поля значенням "None", чи модель коректно працює із неповними векторами, на рисунку 7.1 зображено результати такого тесту.

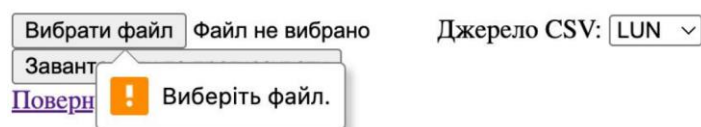
7508	Київ	Голосіївський	None	4		128462.7421875	падає
------	------	---------------	------	---	--	----------------	-------

Рисунок 7.1 – Результат без повних даних

Сценарій 2 – пакетне завантаження CSV-файлу. Перевіряється, як система поводить, коли CSV-файл відсутній або має некоректну структуру. Наприклад, у разі спроби запуску прогнозу без завантаженого файлу очікується чітке повідомлення про помилку. Система не має зависати або повертати пусту сторінку – замість цього користувач бачить повідомлення про відсутність вхідного файлу, що й бачимо на рисунку 7.2.

[Головна](#) | [Пакетне завантаження](#) | [Тренд Київ](#)

Завантажити CSV з оголошеннями



Вибрати файл Файл не вибрано Джерело CSV:

Завантажити

[Поверн](#) ❗ Виберіть файл.

Рисунок 7.2 – Результат при відсутньому файлі

Сценарій 3 – побудова графіка за відсутності повних параметрів. Цей сценарій перевіряє, як поводить система при спробі побудови графіка попиту без наданих параметрів. У першому випадку користувач відкриває вкладку з графіком, не обираючи жодного міста чи району. Система не повинна повертати помилку, зависати або створювати порожній графік із некоректною віссю часу. Очікується, що інтерфейс виведе повідомлення про відсутність даних або запропонує вибрати місто, на рисунку 7.3 бачимо що немає а ні порожнього графіку, а ні повідомлень.

Оберіть місто або регіон для графіка

Місто: Регіон:

Рисунок 7.3 – Результат при порожніх даних графіку

У другому випадку вказується лише місто, без зазначення району. Очікується, що система у відповідь сформує агрегований графік попиту за всіма доступними районами цього міста. Графік будується на основі усереднених

					IC12.300БАК.005 ПЗ	Арк.
						61
Зм.	Лист	№ докум.	Підпис	Дата		

значень прогнозів, отриманих у попередніх запитах, і показує зміни попиту у вигляді лінії тренду, що зображено на рисунку 7.4.

Оберіть місто або регіон для графіка

Місто: Київ Region: — (всі) — Показати

Графік для Київ

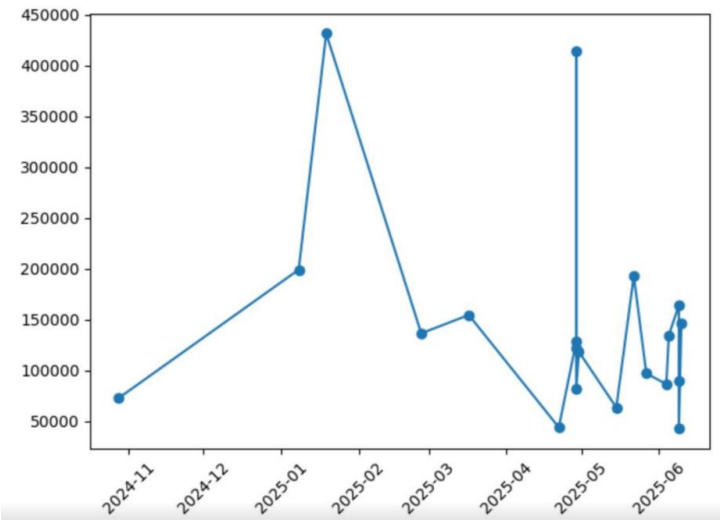


Рисунок 7.4 – Графік при виборі лише міста

Сценарій 4 – перевірка інтеграції з базою даних. Після запуску прогнозу перевіряється, чи створюється новий запис у таблиці property, чи заповнюється request_id. Результат тесту зображено на рисунках 7.5-7.7.

Введіть параметри об'єкта

Місто: Avondale Region: OR
Площа (м²): 76 Кімнат: 3
Рік побудови: 19992
Дата додавання (YYYY-MM-DD): 09.06.2025
Тип об'єкта: flat Джерело: Zillow
Фактична ціна: 110221
Прогнозувати
[Завантажити CSV](#)

Рисунок 7.5 – Введені параметри

7510	zillow	Adelanto	SD	772	11	2005	2025-04-30	flat	777112
------	--------	----------	----	-----	----	------	------------	------	--------

Рисунок 7.6 – Результат в zillow.db

7511	7511	12526835	зростає
------	------	----------	---------

Рисунок 7.7 – Результат в forecast.db

Сценарій 5 – навантаження та продуктивність. Сюди входить перевірка, як система працює під час обробки великої кількості об'єктів. У тестовому режимі використовується CSV-файл із понад 1000 записами. За кожним із них має виконуватися повний цикл обробки – від зчитування до запису у базу й формування таблиці результатів. Протягом цього сценарію відслідковується швидкість обробки, стабільність інтерфейсу, використання оперативної пам'яті та відсутність помилок. Ціль – переконатися, що середній час на обробку одного об'єкта не перевищує 1–2 секунди, а загальна продуктивність не погіршується навіть при навантаженні. Результат зображено на рисунку 7.8.

Результати пакетного прогнозу							
Київ		2 з 1173					
ID	Logo	Region	Площа	Кімнат	Тип	Ціна прогнозу (грн)	Тренд
33229	Київ	Дарницький	33.0	1		51852.92578125	зростає
33230	Київ	Дарницький	77.0	2		122042.8046875	падає
33231	Київ	Печерський	180.0	4		414406.3125	зростає

Рисунок 7.8 – Результат при навантаженні системи

7.3 Аналіз результатів

У результаті проходження обраних тестових сценаріїв вдалося оцінити не лише технічну коректність системи, а й загальну зручність використання. Сценарії виявили стабільну поведінку всіх основних компонентів: вебінтерфейсу, обробки запитів, роботи моделі та взаємодії з базою даних.

У першому сценарії перевірявся випадок, коли користувач залишає поле "площа" порожнім. Прогноз усе одно формується, система не видає помилку, а у

					ІС12.300БАК.005 ПЗ		Арк.
Зм.	Лист	№ докум.	Підпис	Дата			63

виведеній таблиці значення площі відображається як None. Це підтверджує, що модель здатна адаптуватися до часткової відсутності ознак і не потребує повного набору параметрів для виконання прогнозу.

У другому сценарії перевірялась поведінка інтерфейсу у випадку запуску пакетного прогнозу без вибраного файлу. Система спрацьовує коректно, замість запуску процесу вона виводить повідомлення з проханням обрати файл. Жодна помилка або зупинка роботи не відбувається.

У третьому сценарії здійснювалася перевірка коректності збереження даних у базі. Після кожного нового запиту з'являється новий запис у таблиці з прогнозами, з актуальними значеннями ознак, ідентифікатором запиту та отриманим результатом. Це підтверджує повну функціональність інтеграції між інтерфейсом і базовим сховищем.

У п'ятому сценарії перевірялася швидкодія системи при завантаженні великого CSV-файлу. Після імпорту понад тисячі записів прогноз формується майже миттєво – результат з'являється менш ніж за секунду. Програма не зависає, не дає помилок, і коректно обробляє всі об'єкти по черзі, що свідчить про її оптимізовану логіку.

Необхідно наголосити на важливості четвертого сценарію, який стосується побудови графіка. У разі, якщо користувач не обирає жодного параметра, система не виконує жодних дій – інтерфейс залишається незмінним, не з'являється ані помилка, ані порожній графік. Якщо ж зазначити лише місто, наприклад Київ, система автоматично формує загальний графік по цьому місту, агрегуючи прогнози з усіх районів.

Загалом система демонструє надійну роботу в усіх перевірених режимах. Жодного випадку критичної помилки або зупинки не зафіксовано.

Висновки до розділу 7

Тестування системи дозволяє сформуванню чіткого уявлення про її стабільність, зручність взаємодії та готовність до реального використання. Усі протестовані

					ІС12.300БАК.005 ПЗ	Арк.
						64
Зм.	Лист	№ докум.	Підпис	Дата		

сценарії охоплюють основні можливості, від ручного введення до пакетного прогнозування та побудови графіків. Такий підхід забезпечує повноцінну перевірку поведінки програми у звичних і нестандартних умовах.

Результати підтверджують, що система коректно реагує на відсутність деяких параметрів, виводить прогноз навіть за неповного набору ознак, а інтерфейс не допускає запуску процесів без обов’язкових дій користувача – наприклад, без вибору файлу. Дані зберігаються у базі згідно з логікою, без дублювань або втрат. Побудова графіків працює передбачувано: при неповному введенні система або попереджає, або адаптує запит для виводу агрегованого результату.

Навіть при обробці великої кількості записів система зберігає високу продуктивність – час відгуку залишається практично миттєвим. Це особливо важливо для практичного застосування, де йдеться про аналіз сотень або тисяч об’єктів нерухомості одночасно.

У результаті тестування не виявлено критичних помилок або поведінки, яка б заважала користувачам отримувати коректний прогноз. Усі модулі працюють злагоджено, а структура системи дозволяє легко масштабувати її в майбутньому.

ВИСНОВКИ

У дипломному проєкті сформована концепція інформаційної системи прогнозування попиту на ринку житлової нерухомості, що працює на основі аналізу великих даних. У центрі реалізованого підходу – потреба в об’єктивному інструменті оцінки попиту в умовах нестабільного ринку, коли класичні методи вже не забезпечують достатньої точності, швидкості чи адаптивності.

Актуальність задачі визначається зростанням цифровізації ринку, посиленням вимог до прозорості прийняття рішень і потребою в автоматизованій аналітиці з урахуванням просторових та поведінкових змін.

У процесі проєктування розглянуті принципи побудови моделей регресії, обраний метод градієнтного бустингу (XGBoost), сформована структура вхідних ознак і адаптована модель до реальних форматів даних з платформ LUN і Zillow. Також реалізована логіка логарифмування цільової змінної для зменшення впливу викидів та оптимізації навчання.

Проведене тестування підтвердило стабільність роботи моделі при різних вхідних умовах, її стійкість до частково неповних даних, а також високу продуктивність при обробці великих обсягів інформації. Усі перевірені сценарії підтверджують відповідність системи вимогам до сучасного інструмента аналітики: результат повертається швидко, точність прогнозу підтримується в межах допустимого відхилення.

Результати, отримані в процесі проєкту, можуть знайти застосування не лише в межах ринку житлової нерухомості. Сама логіка формування ознак, вибору моделі та оцінки відхилень є універсальною і може бути використана для аналізу об’єктів у інших галузях, наприклад, у логістиці, комерційній нерухомості чи муніципальному плануванні.

					IC12.300БАК.005 ПЗ	Арк.
						66
Зм.	Лист	№ докум.	Підпис	Дата		

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Середні ціни на житло в Україні за регіонами. Flatfy. URL: <https://flatfy.ua>
2. Вторинний ринок житла: дані за 2021-2024 роки. Скільки-скільки?. URL: <https://skilky-skilky.info/vtorynnyy-rynok-zhytla-u-kvitni-na-cherkashchyni-tsina-zrosla-na-43/> (дата звернення: 17.06.2025).
3. James, G., Witten, D., Hastie, T., Tibshirani, R. An Introduction to Statistical Learning. Springer, 2021.
4. Zhang, X., Guhathakurta, S., Li, M. A machine learning approach to predict residential housing price dynamics: A case study of Zillow's Zestimates. Journal of Real Estate Research, 2020. P. 453–479.
5. Toth, D., Nagy, G., Foldesi, P. Real estate price estimation using machine learning techniques. CEUR Workshop Proceedings, 2021. P. 55–64.
6. Офіційний сайт застосунку LUN. URL: <https://www.lun.ua>
7. Офіційний сайт застосунку Zillow. URL: <https://www.zillow.com/>
8. Kelleher, J. D., Mac Namee, B., & D'Arcy, A. Fundamentals of Machine Learning for Predictive Data Analytics. MIT Press, 2015
9. Aurélien Géron. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media, 2019.
10. Python документация. URL: <https://docs.python.org/uk/3.13/tutorial/index.html>
11. Flask documentation. URL: <https://flask.palletsprojects.com/en/stable/>
12. Scrapy documentation. URL: <https://docs.scrapy.org/en/latest/>
13. SQLAlchemy documentation. URL: <https://docs.sqlalchemy.org/en/20/>
14. XGboost documentation. URL: <https://xgboost.readthedocs.io/en/stable/>
15. 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.
16. scikit-learn: Machine Learning in Python. <https://scikit-learn.org>

17. Zhang, X., Guhathakurta, S., & Li, M. A machine learning approach to predict residential housing price dynamics: A case study of Zillow's Zestimates. Journal of Real Estate Research, 2020, P. 453–479.

18. Demand Forecasting Methods: ML for Supply Chain Forecasting. Jelvix. URL: <https://jelvix.com/blog/demand-forecasting-methods-machine-learning> (date of access: 17.06.2025).

19. Toth, D., Nagy, G., & Foldesi, P. Real estate price estimation using machine learning techniques: A case study from Budapest. CEUR Workshop Proceedings, 2021. P. 55–64.

20. Aggarwal, C. C. Data Mining: The Textbook. Springer, 2015.

21. James, G., Witten, D., Hastie, T., Tibshirani, R. An Introduction to Statistical Learning. Springer, 2021.

22. Optimized XGBoost Model with Small Dataset for Predicting Relative Density of Ti-6Al-4V Parts Manufactured by Selective Laser Melting. MDPI. URL: <https://www.mdpi.com/1996-1944/15/15/5298> (date of access: 17.06.2025).