

Факультет Комп'ютерних наук
(повна назва)

Кафедра Системотехніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

другий (магістерський)
(тема)

Дослідження застосування методів аналізу даних в системах електронної
комерції
(тема)

Виконав:
студент 2 курсу, групи СПРМ-22-2

Валентій О.М
(прізвище, ініціали)

Спеціальності 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-наукова

Освітня програма Системне проектування
(повна назва освітньої програми)

Керівник доцент, Тітов С.В
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри системотехніки Гребеннік І.В (підпис)
(прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки
Факультет Комп'ютерних наук
Кафедра Системотехніки
Рівень вищої освіти другий (магістерський)
Спеціальність 122 Комп'ютерні науки
(код і повна назва)
Тип програми освітньо-наукова
(повна назва)
Освітня програма Системне проектування
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)
« ____ » _____ 20 ____ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Студентові Валентію Олександровичу Миколайовичу
(прізвище, ім'я, по батькові)

- Тема роботи Дослідження застосування методів аналізу даних в системах електронної комерції
затверджена наказом по університету від 1 квітня 2024 р. № 259 Ст
- Термін подання студентом роботи до екзаменаційної комісії 13 червня 2024 р.
- Вихідні дані до роботи Розробити серверну і клієнтську частини вебдодатку аналізу даних у системах електронної комерції. Серверна частина є реалізацією бази даних, розробленої для СУБД MySQL. Клієнтська частина має забезпечувати виконання таких бізнес-функцій. Бізнес-функції для незареєстрованих користувачів: перегляд наявних товарів, перегляд інформації за обраним товаром, підбір товарів у кошик, реєстрація та авторизація на сайті. Бізнес-функції для зареєстрованих користувачів: підбір товарів у кошик; оформлення замовлення; перегляд замовлень; вхід до системи зі статусом; Бізнес-функції для адміністраторів: вхід у систему зі статусом «admin»; додавання та редагування інформації про товари; перегляд звітів по результатам аналізу даних отриманих в процесі роботи системи. Операційна система Windows XP або вище, MacOS8 або вище, програмне забезпечення: програмний пакет MySQL Workbench, CASE-засіб All Fusion Data Modeler (ERWin), All Fusion Process Modeler (BPWin), IBM Rational Rose
- Перелік питань, що потрібно опрацювати в роботі провести аналіз предметної області та визначити процеси, які вимагають автоматизації і спосіб у якому вона буде реалізована; сформулювати вимоги до інформаційної системи; створити функціональну модель інформаційної системи («ТО-ВЕ»), використовуючи стандарт IDEF0; провести моделювання діаграм потоків даних використовуючи стандарт DFD; провести функціональне моделювання, визначити і уточнити вимоги до інформаційної системи; провести логічне та фізичне моделювання БД з використанням стандарту IDEF1X; обґрунтувати вибір платформи СУБД для реалізації БД; провести UML-моделювання проектованої клієнтської частини інформаційної системи, створивши діаграму прецедентів, діаграму послідовності дій (Sequence Diagram), реалізувати фізичну модель БД для обраної платформи СУБД, створивши серверну частину інформаційної системи; реалізувати посилальну цілісність даних, а також одну з функцій бізнес-процесу на стороні серверної

частини інформаційної системи; реалізувати один або кілька бізнес-процесів на стороні клієнтської частини інформаційної системи, розробивши інтерфейс доступу до БД

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) схема організаційної структури підприємства; схеми функціональної структури розроблюваної системи у нотаціях IDEF0, DFD, UML; схеми алгоритмів роботи окремих компонентів системи; логічна та фізична модель БД у нотації IDEF1X;

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Огляд та аналіз предметної галузі	14.04.2024 – 20.04.2024	
2	Аналіз існуючих системи електронної комерції	21.04.2024 – 24.04.2024	
3	Огляд методів аналізу даних та їх оцінок	25.04.2024 – 26.04.2024	
4	Вибір методів прогнозування	27.04.2024 – 30.04.2024	
5	Аналіз побудованих моделей для вибору даних для аналізу	01.05.2024 – 10.05.2024	
6	Логічне та фізичне проектування БД системи електронної комерції	11.05.2024 – 17.05.2024	
7	Проектування моделей прогнозування та їх оцінка	18.05.2024 – 25.05.2024	
8	Розробка програмно-апаратної частини компонентів системи із прогнозуванням даних	26.05.2024 – 31.05.2024	
9	Розробка клієнтської сторони інтерфейсу	1.06.2024 – 15.06.2024	
10	Підготовка пояснювальної записки та її додатків	16.06.2024 – 11.06.2024	

Дата видачі завдання _____ 1 _____ квітня _____ 2024 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи: 75 с., 0 табл., 13 рис., 1 додаток, 23 джерел інформації

БАЗА ДАНИХ, СУБД, MYSQL, ЕЛЕКТРОННА КОМЕРЦІЯ, АНАЛІЗ ДАНИХ, C#, ASP, MVC, ML, РЕГРЕСІЯ, ЧАСОВІ РЯДИ

Об'єктом досліджень у кваліфікаційній роботі є аналіз даних в системах електронної комерції.

Предметом дослідження є методи аналізу даних для вирішення задач прогнозування в системах електронної комерції

Мета кваліфікаційної роботи - дослідити методи аналізу даних в системах електронної комерції та реалізувати функції прогнозування на прикладі конкретної системи.

Методи дослідження: системний підхід, методи структурного аналізу і моделювання реляційних баз даних, регресійний аналіз, методи часових рядів, методи проектування веб-застосунків, подієвого об'єктно-орієнтованого програмування.

В роботі проведено аналіз предметної області, що має відношення до методів аналізу даних та їх прогнозування. Для визначення і уточнення вимог до розроблюваного додатку було проведено моделювання потоків даних (з відповідністю до стандарту DFD), логічне та фізичне моделювання (з відповідністю до стандарту IDEF1X). Розроблено UML-діаграми.

Проведено порівняльний аналіз різних методів прогнозування даних і обрано найкращі за оцінками похибки. Розроблені тригери БД для підтримки актуальної інформації у системі. По результатам аналізу методів прогнозування, обрані методи були реалізовані у системі електронної комерції.

Галузь застосування – отримання прогнозу результатів роботи системи електронної комерції для побудови стратегії розвитку і роботи.

ABSTRACT

Explanatory note to the qualification work: 75p., 0 table., 13 fig., 1 appendices, 23 sources of information

DATABASE, DBMS, MYSQL, ELECTRONIC COMMERCE, DATA ANALYSIS, C #, ASP, MVC, ML, REGRESSION, TIME SERIES

The object of research in the qualification work is the analysis of data in e-commerce systems.

The subject of research is data analysis methods for solving forecasting problems in e-commerce systems

The purpose of the qualification work is to investigate the methods of data analysis in e-commerce systems and to implement forecasting functions on the example of a specific system.

Research methods: systems approach, methods of structural analysis and modeling of relational databases, regression analysis, time series methods, methods of designing web applications, event object-oriented programming.

The paper analyzes the subject area related to the methods of data analysis and forecasting. To determine and clarify the requirements for the developed application, data flow modeling (in accordance with the DFD standard), logical and physical modeling (in accordance with the IDEF1X standard) were performed. UML charts developed.

A comparative analysis of different data forecasting methods was performed and the best error estimates were selected. Developed database triggers to support current information in the system. According to the results of the analysis of forecasting methods, the selected methods were implemented in the e-commerce system.

Field of application - obtaining a forecast of the results of the e-commerce system to build a strategy for development and operation

ЗМІСТ

ВСТУП	3
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	5
1.1 Дослідження поняття «Аналіз даних»	5
1.2 Огляд поняття Data mining	8
1.3 Огляд методів аналізу даних	11
1.4 Огляд способів оцінки моделей прогнозування	20
2 ОГЛЯД МЕТОДІВ ТА ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ СИСТЕМ.....	23
2.1 Опис CASE засобів моделювання інформаційних систем	23
2.2 Технології проектування бази даних	27
2.3 Технології для реалізації інтерфейсу розроблюваної системи.....	28
2.4 Технології для реалізації інформаційної системи	29
2.5 Технології аналізу даних у системі	30
3 ПОСТАНОВКА ЗАДАЧІ	32
3.1 Призначення розробки обраної інформаційної системи.....	32
3.2 Мета впровадження методів аналізу даних у систему електронної комерції	34
3.3 Функціональні можливості розроблюваної системи	35
3.4 Основні не функціональні вимоги до системи	36
3.5 Основні вимоги до безпеки	38
4 МАТЕМАТИЧНИЙ ОПИС АЛГОРИТМІВ ПРОГНОЗУВАННЯ.....	39
4.1 Алгоритм методу SDCA.....	39
4.2 Алгоритм методу регресії Пуассона	44
4.3 Алгоритм методу дерев регресії.....	47
4.4 Алгоритм методу прогнозування часових рядів.....	50
5 РОЗРОБКА АРХІТЕКТУРИ ТА КОМПОНЕНТІВ СИСТЕМИ.....	52
5.1 Розробка системних вимог.....	52
5.2 Розробка моделі потоків даних.....	54
5.3 Діаграма класів розроблюваної системи	62
5.4 Логічне і фізичне моделювання даних	66
5.5 Розробка моделей прогнозування продажів.....	67
5.6 Апробація методів прогнозування на прикладі системи продажів відеоігор	72
5.7 Розробка SQL-запитів для організації роботи системи.....	75
ВИСНОВКИ.....	76
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	78
ДОДАТКИ	Ошибка! Закладка не определена.

ВСТУП

Стрімкий розвиток інформаційних технологій та поширення Інтернету призвели до значного зростання популярності електронної комерції в усьому світі. Згідно з даними статистики, глобальний обсяг роздрібних продажів в Інтернеті в 2023 році склав \$5,7 трлн, що на 18% більше, ніж у 2022 році. Прогнозується, що до 2025 року ця цифра сягне \$8,1 трлн. Така тенденція зумовлена зручністю онлайн-покупок, широким асортиментом товарів і послуг, а також можливістю порівняння цін у різних продавців.

Однак успіх електронної комерції значною мірою залежить від ефективного використання методів аналізу даних. Величезні обсяги інформації, що генеруються в результаті взаємодії користувачів з веб-сайтами та онлайн-платформами, містять цінні відомості про їхні переваги, поведінку та купівельну спроможність. Аналіз цих даних дозволяє компаніям краще зрозуміти своїх клієнтів, персоналізувати пропозиції, оптимізувати ціноутворення та маркетингові стратегії, прогнозувати попит і, як наслідок, підвищувати ефективність бізнесу та конкурентоспроможність на ринку.

Проте застосування методів аналізу даних в електронній комерції пов'язане з низкою викликів, таких як обробка великих обсягів даних, різноманітність джерел інформації, забезпечення конфіденційності та захисту персональних даних користувачів. Тому постає важлива проблема вибору та адаптації відповідних методів аналізу даних, які б дозволили ефективно вирішувати ці виклики та отримувати максимальну цінність з наявних даних.

Тема "Методи аналізу даних у системах електронної комерції" надзвичайно актуальна, оскільки в сучасних умовах створення системи електронної комерції - лише початок. Величезний обсяг даних, що накопичується в процесі роботи таких систем, може бути використаний для оптимізації процесів та навіть прогнозування різних параметрів, таких як обсяги продажів.

Метою даної роботи є дослідження та порівняльний аналіз різних методів обробки й аналізу даних, придатних для використання в системах електронної комерції, та розробка рекомендацій щодо їх практичного застосування для

вирішення конкретних завдань, таких як персоналізація пропозицій, прогнозування попиту, виявлення шахрайських дій тощо.

Об'єктом дослідження є аналіз даних в системах електронної комерції.

Предметом дослідження є методи аналізу даних для вирішення задач прогнозування в системах електронної комерції.

Очікується, що результати дослідження матимуть практичну цінність для компаній, що працюють у сфері електронної комерції, і допоможуть їм ефективно використовувати наявні дані для підвищення рівня обслуговування клієнтів, оптимізації бізнес-процесів і зміцнення конкурентних позицій на ринку.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Дослідження поняття «Аналіз даних»

Дослідження методів аналізу даних у системі електронної комерції може включати різноманітні аспекти, що охоплюють обробку, візуалізацію, моделювання та інтерпретацію даних з метою виявлення закономірностей, трендів та важливих відмінностей для підтримки прийняття рішень у сфері електронної комерції. Ось кілька складових цієї теми [1]:

1. Обробка даних. Включає в себе збір, очищення, трансформацію та інтеграцію даних з різних джерел, таких як транзакційні бази даних, веб-логи, соціальні медіа тощо.

2. Візуалізація даних. Цей аспект полягає в створенні графіків, діаграм та інших візуальних засобів для представлення даних у зрозумілій формі. Візуалізація може допомогти відкрити шаблони, тренди та аномалії, які можуть бути важливими для бізнесу.

3. Методи аналізу даних. Включає в себе статистичний аналіз, машинне навчання, штучний інтелект та інші аналітичні методи для виявлення закономірностей та здійснення прогнозування в електронній комерції. Це може включати класифікацію клієнтів, прогнозування продажів, аналіз сегментації ринку та інше.

4. Оптимізація та прийняття рішень. На основі аналізу даних можна розробити стратегії оптимізації процесів електронної комерції та приймати обґрунтовані рішення щодо маркетингових кампаній, асортименту товарів, ціноутворення тощо.

5. Прогнозування та тренди. Аналіз даних може допомогти виявити майбутні тренди та прогнозувати попит на товари і послуги в електронній комерції, що дозволить підприємствам готуватися до майбутніх викликів та можливостей.

Ці складові можуть бути детальніше досліджені та використані для розвитку нових методів та інструментів аналізу даних у системі електронної комерції [13, с. 84].

Під час аналізу даних, дослідник виконує ряд операцій з метою формування конкретних уявлень про характеристики явищ або інформації, що вони описують. Аналіз даних не обмежується просто обробкою інформації після її збору; це також інструмент для перевірки гіпотез та вирішення завдань експериментатора. Наприклад, при аналізі результатів роботи користувачів з системою, процес полягає у визначенні загальної суми замовлень для кожного зареєстрованого користувача та розміру їх знижки.

Після оформлення замовлення одним з ключових етапів аналізу даних в системі електронної комерції може бути виділення обмеженої кількості товарів, які найчастіше купуються користувачами разом із замовленим товаром. Цей процес часто називають аналізом асоціативних правил або аналізом асоціацій.

Аналіз асоціативних правил дозволяє виявити зв'язки між різними наборами товарів у замовленнях користувачів. Він допомагає визначити, які товари мають тенденцію купуватися разом, і ця інформація може бути використана для кількох цілей [22]:

1. Персоналізовані рекомендації. Система може запропонувати користувачеві доповнити замовлення товарами, які часто купуються разом із вже замовленими. Це може підвищити середній розмір замовлення та задоволеність клієнтів.

2. Крос-продажі та бандлінг. Виявлені асоціації між товарами можна використовувати для створення спеціальних пропозицій, пакетних продажів або крос-продажів.

3. Оптимізація розміщення товарів. Асоційовані товари можна розташувати поруч на сайті або в каталогах, щоб покупці могли легше їх знайти та придбати разом.

4. Управління запасами. Інформація про асоціації допоможе краще планувати запаси та замовлення товарів, які часто купуються разом.

Для виявлення асоціативних правил зазвичай використовуються різні алгоритми, такі як Apriori, FP-Growth або Eclat. Ці алгоритми аналізують історичні дані замовлень, щоб виявити частоту поєднання різних товарів у замовленнях. Потім вони використовують міри підтримки та достовірності, щоб відфільтрувати найбільш сильні та значущі асоціації.

Після визначення найсильніших асоціацій система може представити їх користувачеві у вигляді рекомендацій або запропонувати у формі крос-продажів чи пакетних пропозицій. Це може допомогти збільшити прибуток компанії та покращити задоволеність клієнтів, надаючи їм релевантні пропозиції товарів, які часто купуються разом [3].

Так, аналіз даних у системах електронної комерції є вкрай корисним і дозволяє власникам систем точніше розуміти потреби та вподобання їхніх клієнтів. Збираючи та аналізуючи дані про покупки, поведінку користувачів та їхній фідбек, вони можуть швидко реагувати на зміни на ринку та попит і відповідно адаптувати свої послуги або товари. Наприклад, на основі аналізу покупок користувачів система може рекомендувати подібні або доповнюючі товари, що сприяє збільшенню продажів і покращенню задоволеності клієнтів. Такий підхід дозволяє власникам систем ефективно керувати та оптимізувати їхню діяльність, забезпечуючи більшу конкурентоспроможність і задоволеність клієнтів.

Відображення складної реальності у вигляді моделей дозволяє нам краще розуміти і аналізувати світ, навіть за умов обмежених пізнавальних здібностей. Моделі і моделювання стають важливим інструментом для спрощення та узагальнення складних об'єктів, подій і систем, що дозволяє нам здійснювати аналіз даних більш ефективно.

У випадку аналізу даних в системах електронної комерції, прогнозування продажів на майбутні періоди часу є актуальною задачею. Застосування моделей для цієї задачі дозволяє нам використовувати наявні дані про продажі за попередні періоди, а також інші фактори, які можуть впливати на продажі, для прогнозування майбутніх тенденцій. Це може включати такі фактори, як сезонність, маркетингові кампанії, конкуренція на ринку тощо [19, с. 55].

Математичні моделі, такі як часові ряди, регресійний аналіз, нейронні мережі та інші, дозволяють нам аналізувати ці дані і робити прогнози з урахуванням різних факторів. Це допомагає власникам бізнесу планувати запаси, оптимізувати виробництво та маркетингові стратегії для максимізації прибутку та задоволення потреб клієнтів.

1.2 Огляд поняття Data mining

Відображення складної реальності у вигляді моделей дійсно допомагає нам краще розуміти і аналізувати світ, навіть за умов обмежених пізнавальних здібностей. Моделі і моделювання стають ключовим інструментом для спрощення та узагальнення складних об'єктів, подій і систем, що дозволяє здійснювати аналіз даних більш ефективно.

У сфері аналізу даних в системах електронної комерції, прогнозування продажів на майбутні періоди часу є важливою задачею. Використання моделей для цієї задачі дозволяє нам використовувати наявні дані про продажі за попередні періоди, а також інші фактори, такі як сезонність, маркетингові кампанії, конкуренція на ринку тощо, для прогнозування майбутніх тенденцій [2, с. 11].

Математичні моделі, такі як часові ряди, регресійний аналіз, нейронні мережі та інші, дозволяють аналізувати ці дані і робити прогнози з урахуванням різних факторів. Це допомагає власникам бізнесу планувати запаси, оптимізувати виробництво та маркетингові стратегії для максимізації прибутку та задоволення потреб клієнтів.

Методи аналізу даних Data Mining в бізнесі відкривають широкі можливості для оптимізації різних аспектів діяльності. Ось деякі з них [5, с.22]:

1. Сегментування клієнтів. Дозволяє визначити групи клієнтів зі схожими характеристиками або поведінкою, що дозволяє персоналізувати маркетингові стратегії.

2. Прогнозування продажів. Аналізує минулі продажі та інші фактори, щоб передбачити майбутні тенденції і попит на товари або послуги.

3. Аналітика складських запасів. Допомагає оптимізувати запаси, уникати перевитрат або недостачі товарів на складі.

4. Прийняття рішень про індивідуальні знижки для клієнтів. Аналіз даних може виявити патерни покупок та поведінку клієнтів, що допомагає приймати рішення щодо персоналізованих знижок або пропозицій.

5. Залучення нових клієнтів. Аналіз даних допомагає ідентифікувати потенційних клієнтів та оптимізувати маркетингові кампанії для їх залучення.

Щодо методів аналізу даних, статистичні та кібернетичні підходи використовуються для різних цілей [9, с. 68]:

- Статистичні методи включають аналіз вихідних даних, багатомірний статистичний аналіз, аналіз зв'язків та аналіз часових рядів.

- Кібернетичні методи об'єднують підходи, засновані на математичних методах та штучному інтелекті, такі як кластеризація, алгоритм k-середніх, байєсовські мережі та штучні нейронні мережі.

Ці методи дозволяють аналізувати великі обсяги даних та отримувати цінні висновки для прийняття стратегічних рішень в бізнесі.

Завдання інтелектуального аналізу даних можна розділити за способом навчання та метою аналізу. Ось кілька типових завдань аналізу даних:

1. Задача класифікації. Для кожного об'єкта в наборі даних призначається категорія або клас. Наприклад, класифікація електронної пошти на "спам" та "не спам".

2. Задача регресійного аналізу. Досліджується множина значень з безперервного діапазону. Наприклад, прогнозування ціни на нерухомість на основі різних факторів, таких як площа, місцезоташування тощо.

3. Задача прогнозування. Система на основі історичних даних прогнозує майбутні значення. Наприклад, прогнозування продажів на наступний місяць на основі попередніх продажів та інших факторів.

4. Задача кластеризації або сегментації. Об'єкти розділяються на групи таким чином, що об'єкти всередині групи подібні один до одного, але відрізняються від об'єктів в інших групах. Наприклад, сегментація ринку на основі споживчих звичок.

5. Задача визначення взаємозв'язків. Шукає набори даних, які спільно відбуваються частіше, що дозволяє виявити взаємозв'язки між різними аспектами даних.

6. Задача аналізу відхилень. Визначає події або шаблони, які відрізняються від очікуваного зразка. Наприклад, виявлення шахрайських дій в банківських транзакціях.

Кожна з цих задач може бути вирішена методами навчання без учителя або з учителем, в залежності від того, чи надається моделі попередньо позначена навчальна інформація. Також, вони можуть мати описовий характер, коли просто констатують певні закономірності, або прогнозують майбутні події.

Сьогодні існує широкий спектр технологій для аналізу різних типів даних. Ось деякі з них [21]:

1. Data Mining. Ця технологія зосереджена на виявленні закономірностей та взаємозв'язків у великих наборах структурованих даних.

2. Text Mining (текстовий аналіз). Використовується для аналізу текстової інформації з різних джерел, таких як соціальні медіа, електронні листи, новинні статті тощо, з метою виявлення патернів і здійснення висновків.

3. Visual Mining (візуальний аналіз). Використовує графічні методи для візуалізації даних, дозволяючи аналізувати та розуміти великі обсяги інформації швидше та ефективніше.

4. OLAP (online analytical processing, аналітична обробка у реальному часі). Це підходить до аналізу мульти-дименсійних даних, які дозволяють користувачам аналізувати дані з різних перспектив.

5. Process Mining (аналіз процесів). Фокусується на аналізі бізнес-процесів, виявленні взаємозв'язків та можливих покращень у виробничих процесах.

6. Web Mining (аналіз Web-ресурсів). Використовується для аналізу даних, які знаходяться в Інтернеті, для виявлення трендів, патернів споживання, аналізу конкурентів тощо.

7. Real-Time Data Mining (аналіз в режимі реального часу). Ця технологія дозволяє аналізувати дані в реальному часі, що дозволяє оперативно реагувати на зміни в даних та отримувати актуальну інформацію.

Кожна з цих технологій використовується для аналізу конкретного типу даних та вимагає спеціалізованих програмних рішень, сервісів та знань для ефективного використання. Виробники програмних засобів активно розробляють нові і удосконалюють існуючі рішення, щоб задовольнити попит на аналітичні інструменти у всіх секторах бізнесу, незалежно від їх розміру та бюджетних обмежень.

1.3 Огляд методів аналізу даних

Аналітичний підхід до моделювання полягає у використанні моделей як основи для вивчення системи. Дослідник вибирає модель на основі експертних знань та міркувань. Ці моделі можуть бути різного виду [20]:

1. Теоретичні моделі. Ці моделі базуються на відомих математичних законах та теоріях. Вони визначають взаємозв'язки між вхідними параметрами та вихідними результатами системи. Такі моделі найчастіше виражені у вигляді функціональних залежностей або рівнянь.

2. Емпіричні моделі. Ці моделі будуються на основі емпіричних даних та спостережень. Вони враховують зв'язки та закономірності, які виявляються під час експериментів або спостережень, але можуть не мати теоретичного підґрунтя.

3. Змішані (напівемпіричні) моделі. Ці моделі поєднують елементи емпіричних та теоретичних підходів. Вони можуть використовувати емпіричні дані для побудови базової моделі, а потім доповнювати її теоретичними аспектами або коригувати на основі теоретичних законів.

Кожен з цих типів моделей має свої переваги та обмеження, і вибір конкретного типу моделі залежить від специфіки досліджуваної системи, наявності даних та мети дослідження.

Так, теоретичні моделі часто базуються на емпіричних даних та спостереженнях, і це дозволяє їм краще відображати реальні процеси. Наприклад, багато законів фізики спочатку були вивчені та сформульовані на основі експериментальних даних. Використання емпіричних даних для побудови

теоретичних моделей дозволяє зрозуміти фундаментальні принципи та закономірності, що керують досліджуваними явищами.

Щодо інформаційного підходу до аналізу даних, він дійсно включає не лише модель, але й інші важливі складові, такі як експерт, гіпотеза і аналітик. Експерт, як ключова фігура, має розуміння предметної області та може висувати гіпотези або припущення щодо можливих закономірностей у даних. Його знання і досвід допомагають у визначенні важливих змінних та методів аналізу [4,с.47].

Гіпотеза, в свою чергу, є початковою твердженням або припущенням, яке потрібно перевірити або спростувати на основі аналізу даних. Аналітик відповідає за розробку стратегій аналізу даних, вибір методів та інструментів для перевірки гіпотез і отримання висновків.

Взаємодія між цими складовими дозволяє створювати ефективні аналітичні рішення, які поєднують людський інтелект з комп'ютерною потужністю для кращого розуміння даних та виявлення корисної інформації.

Так, роль аналітика в аналізі даних є критичною, оскільки він є посередником між експертами різних областей та технічними аспектами аналізу даних. Він забезпечує ефективну взаємодію між експертами, збирає гіпотези, встановлює вимоги до даних, виконує аналіз та спільно з експертами аналізує результати. Для успішної роботи аналітик повинен мати широкі системні знання, які включають як предметні області, так і технічні аспекти аналізу даних.

Також, перед тим як перейти до огляду конкретних методів аналізу даних, важливо розглянути категорії аналізу даних за їх складністю та кількістю операцій, необхідних для їх реалізації. Поступове збільшення складності методів дозволяє нам систематично класифікувати їх та вивчати їх відповідно до їх потенційного використання та ефективності.

1. Описовий аналіз даних - це процес виявлення та представлення основних характеристик і закономірностей у наборі даних. Його мета полягає в тому, щоб зрозуміти структуру даних, їхні властивості та надати стислий, зрозумілий опис даних. Описовий аналіз є важливим першим кроком у процесі аналізу даних і зазвичай передує більш глибокому аналізу та моделюванню.

Основні завдання описового аналізу даних [7, с. 143]:

1.1 Огляд даних. Отримання загального уявлення про набір даних, його розмірність, тип змінних, наявність пропущених значень тощо.

1.2 Опис змінних. Визначення типів змінних (числові, категоріальні, текстові), їхніх діапазонів, центральних тенденцій (середнє, медіана, мода) і міри розсіювання (дисперсія, стандартне відхилення).

1.3. Візуалізація даних. Створення графічних представлень даних, таких як гістограми, діаграми розсіювання, лінійні графіки тощо, для кращого розуміння розподілів і взаємозв'язків між змінними.

1.4. Виявлення викидів та аномалій. Ідентифікація незвичних або екстремальних значень у даних, які можуть вплинути на подальший аналіз.

1.5. Перевірка припущень. Оцінка відповідності даних певним статистичним припущенням, таким як нормальність розподілу, відсутність мультиколінеарності тощо.

Описовий аналіз даних часто виконується за допомогою статистичних методів, таких як обчислення описових статистик (середнє, медіана, квартилі тощо), побудова графіків та діаграм, кореляційний аналіз та інші методи.

Результати описового аналізу формують базове розуміння даних і допомагають визначити подальші кроки в аналітичному процесі, такі як очищення даних, трансформація, вибір відповідних методів моделювання та інтерпретація результатів.

2. Дослідницький аналіз - це процес дослідження зв'язків між даними. Його головною метою є виявлення зв'язків та формулювання гіпотез і моделей для вирішення конкретних проблем. Після упорядкування даних у нормалізованому вигляді дослідницький аналіз дозволяє аналізувати зв'язки між ними і встановлювати потенційні залежності. Цей аналіз особливо корисний для інтелектуального аналізу даних, де потрібно розуміти та використовувати дані для вирішення складних завдань [8, с. 59].

3. Діагностичний аналіз даних - це тип аналізу, спрямований на пошук причин певних подій або явищ шляхом вивчення історичних даних. Його основна мета полягає у визначенні головних факторів, що лежать в основі спостережуваного результату або проблеми.

Діагностичний аналіз зазвичай включає наступні етапи:

3.1. Визначення питання або проблеми: На цьому етапі формулюється мета діагностичного аналізу, наприклад, з'ясувати причини зниження продажів, високої плинності клієнтів або низької ефективності певного процесу.

3.2. Збір відповідних даних. Збираються дані, що можуть бути пов'язані з досліджуваним питанням або проблемою. Це можуть бути як внутрішні дані компанії (продажі, фінанси, операційні показники), так і зовнішні дані (ринкові тренди, демографічна інформація тощо).

3.3. Очищення та підготовка даних. Дані проходять процес очищення від помилок, пропущених значень та неузгодженостей, а також можуть бути трансформовані в зручний для аналізу формат.

3.4. Дослідження даних та виявлення закономірностей. Застосовуються різні аналітичні методи, такі як статистичний аналіз, кореляційний аналіз, регресійний аналіз, аналіз часових рядів, візуалізація даних тощо. Метою є виявлення шаблонів, тенденцій та потенційних факторів, що можуть впливати на результат.

3.5. Інтерпретація результатів. Результати аналізу інтерпретуються експертами, які формулюють висновки про можливі причини досліджуваного явища або проблеми. Це може включати оцінку значущості різних факторів та їх взаємозв'язків.

3.6. Рекомендації та план дій. На основі проведеного діагностичного аналізу можуть бути запропоновані конкретні рекомендації та план дій для вирішення виявленої проблеми або усунення її причин.

Діагностичний аналіз є корисним інструментом для бізнесу, оскільки він дозволяє зрозуміти глибинні причини певних ситуацій та визначити сфери, на які потрібно зосередити зусилля для покращення результатів. Він часто застосовується в різних галузях, таких як фінанси, маркетинг, виробництво, охорона здоров'я та інших [23].

4. Прогнозний аналіз - це процес використання наявних даних для передбачення майбутніх подій, тенденцій або значень. Цей аналіз може використовувати різноманітні статистичні методи, машинне навчання або інші алгоритми, щоб зробити прогнози на основі історичних даних та інших факторів.

У контексті систем електронної комерції прогнозний аналіз може бути використаний для передбачення майбутніх обсягів продажів, поведінки споживачів, попиту на товари чи послуги, а також для ідентифікації тенденцій у ринковій динаміці.

Наприклад, на основі аналізу історичних даних про продажі товарів або послуг попередніх періодів можна розробити модель прогнозування майбутніх продажів. Ця модель може враховувати різноманітні фактори, такі як сезонність, рекламні кампанії, економічні та соціокультурні фактори, що впливають на попит. Такий прогнозний аналіз дозволяє бізнесу планувати свої дії та ресурси більш ефективно, мінімізуючи ризики та максимізуючи прибуток.

5. Рекомендаційний аналіз - це процес використання даних для надання індивідуалізованих рекомендацій користувачам щодо товарів, послуг або інших елементів, які їх можуть зацікавити. Цей вид аналізу часто використовується в системах електронної комерції для підтримки персоналізованого досвіду користувача та збільшення продажів.

Наприклад, рекомендаційний аналіз може включати в себе аналіз історії покупок користувача, його переглядів товарів, доданих до кошика або оцінок товарів. На основі цих даних можна розробити модель, яка передбачає, які товари або послуги можуть бути цікавими конкретному користувачеві. Ці рекомендації можуть бути відображені на сайті або в мобільній додатку, що дозволяє залучити увагу користувача до додаткових продуктів та збільшити його задоволеність покупкою [24].

У підсумку, рекомендаційний аналіз допомагає покращити персоналізований досвід користувача, збільшити продажі та підвищити задоволеність клієнтів у системах електронної комерції.

В сучасному світі розрізняють два основних підходи до методів аналізу даних - "статистичний" і "гуманітарний". Статистичні (кількісні) методи в основному застосовуються в науках, що вивчають закономірності, проявлені в структурі, динаміці та взаємозв'язках соціально-економічних подій, таких як економіка. Гуманітарні (якісні) методи застосовуються в науках, що вивчають

закономірності функціонування і розвитку суспільства, таких як політологія. Зазвичай ці методи використовуються разом у наукових роботах.

Крім того, використання цих методів зазвичай відбувається в певному порядку [1]:

1. Визначення показників і порядку їх збору: Цей етап передбачає визначення конкретних показників, які необхідно зібрати для аналізу, а також способів їх збору.

2. Збір показників: Після визначення показників вони збираються з відповідних джерел даних або отримуються в процесі проведення дослідження.

3. Зведення та, при необхідності, угруповання показників: На цьому етапі дані можуть бути об'єднані, агреговані або узагальнені для подальшого аналізу.

4. Обробка показників: Дані обробляються за допомогою відповідних статистичних методів для отримання корисної інформації або виявлення певних закономірностей.

Статистичний або кількісний аналіз - це методи аналізу даних, які використовуються для вивчення відносин між змінними та виявлення патернів у даних за допомогою математичних та статистичних моделей.

У системах електронної комерції статистичний або кількісний аналіз може бути використаний для різних цілей, таких як вивчення та оцінка ефективності маркетингових кампаній, аналіз впливу різних факторів на покупку товарів, визначення тенденцій у поведінці споживачів та багато іншого.

Наприклад, за допомогою статистичного аналізу можна провести А/В-тестування, щоб визначити, яка версія веб-сайту чи рекламного матеріалу привертає більше уваги користувачів та призводить до більшої кількості конверсій. Також можна застосовувати методи регресійного аналізу для прогнозування продажів на основі різних факторів, таких як ціна, рекламні витрати, час року тощо.

Отже, статистичний або кількісний аналіз є потужним інструментом у системах електронної комерції для прийняття обґрунтованих та ефективних рішень, спрямованих на оптимізацію бізнес-процесів та забезпечення успішності підприємства [5, с. 11].

Кластерний аналіз - це метод аналізу даних, який використовується для групування схожих об'єктів разом у кластери або групи на основі їхніх властивостей або характеристик. Метою кластерного аналізу є виявлення природних структур у наборі даних, що дозволяє відокремити об'єкти на подібні групи без попереднього знання про класифікацію.

У системах електронної комерції кластерний аналіз може бути використаний для різних цілей, таких як сегментація аудиторії, персоналізація маркетингових кампаній, аналіз покупок користувачів та ідентифікація груп товарів або послуг, які часто купуються разом.

Наприклад, за допомогою кластерного аналізу можна розділити клієнтів на різні групи на основі їхніх покупок, таких як типи придбаних товарів, суми покупок, частота покупок тощо. Це дозволяє зрозуміти потреби та уподобання різних сегментів аудиторії та адаптувати стратегії маркетингу та обслуговування для кожної групи окремо.

Отже, кластерний аналіз є корисним інструментом у системах електронної комерції для розуміння та використання різних груп користувачів, щоб підвищити ефективність маркетингових зусиль та покращити задоволеність клієнтів.

Цей метод застосовується у різних галузях, включаючи маркетинг, медицину, соціологію, біологію та інші. Наприклад, в маркетингу кластерний аналіз може бути використаний для сегментації аудиторії на групи з подібними характеристиками для більш ефективного спрямування маркетингових кампаній. У медицині цей метод може допомогти ідентифікувати підгрупи пацієнтів з подібними симптомами або характеристиками для розробки індивідуальних підходів до лікування.

Кластерний аналіз може бути реалізований за допомогою різних алгоритмів, таких як алгоритм k-середніх (k-means), ієрархічний кластерний аналіз (hierarchical clustering), DBSCAN та інші. Кожен з цих алгоритмів має свої переваги і недоліки і може бути застосований в залежності від конкретного завдання та властивостей даних.

Когортний аналіз - це метод вивчення поведінки групи людей (або інших об'єктів), які мають спільну характеристику або вхід у певний період часу. Когорти

можуть бути сформовані на основі різних критеріїв, таких як дата реєстрації, дата першого покупки, вік, регіон і т.д [22].

Основна ідея когортного аналізу полягає в тому, щоб вивчати зміни в певному показнику чи патерні поведінки в часі в межах кожної когорти. Наприклад, у маркетингових дослідженнях когортний аналіз може допомогти вивчити тенденції у споживчому купівельному поведінці, такі як зміни в середньому чеку чи частоті покупок в різні періоди часу після реєстрації користувачів.

Цей метод може бути корисним для виявлення факторів, що впливають на різноманітні аспекти діяльності, такі як збереження клієнтів, конверсія, рост доходів тощо. Крім того, він може допомогти виявити вплив різних стратегій маркетингу чи обслуговування на показники ефективності.

Когортний аналіз часто використовується в інтернет-маркетингу та аналізі веб-трафіку для оцінки ефективності маркетингових кампаній, вивчення патернів поведінки користувачів на сайті та визначення факторів, які впливають на конверсію.

Нейронні мережі - це комп'ютерні системи, що моделюють роботу людського мозку. Вони складаються зі штучних нейронів, які взаємодіють між собою за допомогою зважених зв'язків. Ці нейрони організовані в шари, інформація передається від вхідного шару через приховані шари до вихідного шару.

Основна ідея нейронних мереж полягає в тому, що вони можуть навчатися розв'язувати завдання на основі набору вхідних даних і відповідних вихідних значень. Процес навчання полягає в тому, що мережа змінює ваги своїх зв'язків таким чином, щоб максимізувати точність прогнозування вихідних значень на основі вхідних даних. Цей процес навчання може включатися, наприклад, з використанням алгоритму зворотного розповсюдження помилки[20].

Нейронні мережі зараз широко застосовуються в різних сферах, таких як комп'ютерне бачення, розпізнавання мови, медичинська діагностика, фінансовий аналіз, автономні транспортні засоби тощо. Вони демонструють вражаючі

результати в багатьох завданнях, зокрема там, де є складність у виявленні шаблонів або великий обсяг даних, які потрібно обробляти.

Прогнозування часових рядів - це процес передбачення майбутніх значень величини на основі історичних даних, де час виступає у якості основного фактора. Це важлива задача аналізу даних, яка має велике застосування в таких галузях, як фінанси, економіка, метеорологія, маркетинг і торгівля.

Для прогнозування часових рядів використовуються різні методи, включаючи [24]:

1. Експоненційне згладжування. Цей метод використовується для прогнозування часових рядів, які не мають складних тенденцій або сезонності. Він базується на тому, що нові значення в серії вагуються залежно від їх відстані від поточного моменту часу.

2. Авторегресійні моделі (AR). Ці моделі використовують історичні значення змінної для прогнозування майбутніх значень. AR-модель передбачає, що значення величини в певний момент часу залежить від попередніх значень цієї ж величини.

3. Моделі ковзних середніх (MA). Ці моделі використовуються для прогнозування на основі ковзних середніх значень часового ряду. Вони передбачають, що значення величини в певний момент часу залежить від попередніх значень помилок прогнозування.

4. Моделі авторегресії з ковзними середніми (ARMA). Ці моделі поєднують в собі характеристики AR- і MA-моделей для більш точного прогнозування часових рядів.

5. Моделі авторегресії зі змінною ковзною середньою (ARIMA). Ці моделі розширюють ARMA-моделі, дозволяючи враховувати тренд та сезонні зміни в часовому ряді.

Ці методи допомагають аналізувати та передбачати зміни в часових рядах, що допомагає приймати більш обгрунтовані рішення в різних галузях діяльності.

1.4 Огляд способів оцінки моделей прогнозування

Точність прогнозованої моделі дійсно є важливим аспектом її ефективності. В контексті прогнозування, точність визначається як відповідність прогнозованих значень реальним даним. Чим більш точна модель, тим менше похибок у прогнозах.

Оцінка моделей прогнозування є важливим етапом у процесі розробки та застосування аналітичних моделей. Існують різні методи оцінки ефективності моделей прогнозування, деякі з найпоширеніших з них включають [14, с 37]:

1. Середньоквадратична помилка (Mean Squared Error, MSE). Це одна з найпоширеніших метрик для оцінки точності прогнозів. MSE обчислюється як середнє значення квадратів різниці між прогнозованими та спостережуваними значеннями.

2. Коефіцієнт детермінації (Coefficient of Determination, R-squared). Ця метрика вимірює частку варіації в залежній змінній, яка пояснюється моделлю. Вона може бути корисною для визначення того, наскільки добре модель адаптується до даних.

3. Середнє абсолютне відхилення (Mean Absolute Error, MAE). Ця метрика вимірює середнє значення абсолютних різниць між прогнозованими та фактичними значеннями. Вона також широко використовується для оцінки точності прогнозів.

4. Крос-валідація (Cross-Validation). Цей метод включає розділення даних на навчальний та тестовий набори, щоб оцінити ефективність моделі. Використовуючи крос-валідацію, можна з'ясувати, наскільки добре модель узгоджується з новими даними.

5. Крива ROC (Receiver Operating Characteristic Curve). Ця метрика використовується для оцінки якості моделей класифікації. Вона вимірює відношення між чутливістю та специфічністю моделі при різних порогових значеннях.

6. AIC та BIC (Akaike Information Criterion та Bayesian Information Criterion). Ці критерії використовуються для порівняння моделей на основі їхньої складності та точності.

Ці методи оцінки можуть бути використані окремо або у поєднанні залежно від конкретних потреб та характеру даних.

Для оцінки точності прогнозованої моделі можна використовувати одну або декілька з цих метрик, залежно від конкретних вимог та особливостей задачі прогнозування.

Так, Mean Squared Error (MSE) є дуже поширеною метрикою для оцінки точності моделей регресії. Вона вимірює квадрат середньої різниці між прогнозованими значеннями моделі та фактичними значеннями. Ось як вона обчислюється [12, с. 59]:

1. Обчисліть квадрат відхилення між кожним прогнозованим значенням і відповідним фактичним значенням.

2. Знайдіть середнє значення цих квадратів.

3. Це і буде MSE.

Формула MSE виглядає наступним чином:

$$MSE = \frac{1}{N} * \sum_{i=1}^N (y_i - \tilde{y}_i)^2 ,$$

де:

n - кількість прикладів у наборі даних,

y_i - фактичне значення,

\tilde{y}_i - прогнозоване значення.

Чим менше значення MSE, тим краще модель прогнозує. Однак слід пам'ятати, що MSE має квадратичну одиницю виміру, що може робити його важко інтерпретувати в контексті вихідних даних.

Mean Absolute Error (MAE) є іншою популярною метрикою для оцінки точності моделей регресії. На відміну від MSE, яка вимірює середньоквадратичну похибку, MAE вимірює середню абсолютну похибку.

Формула MAE виглядає наступним чином:

$$MAE = \frac{1}{N} * \sum_{i=1}^N |y_i - \tilde{y}_i| ,$$

де:

n - кількість прикладів у наборі даних,

y_i - фактичне значення,

\tilde{y}_i - прогнозоване значення.

Оцінка MAE дозволяє оцінити середню абсолютну величину похибки між фактичними та прогнозованими значеннями. Ця метрика є лінійною і не враховує напрямку похибки. Це означає, що вона не надає більшої ваги великим похибкам, як це робить MSE або RMSE. MAE дозволяє отримати інтуїтивно зрозумілу оцінку точності моделі, оскільки вона просто вказує на середнє абсолютне відхилення між прогнозованими і фактичними значеннями.

2 ОГЛЯД МЕТОДІВ ТА ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ СИСТЕМ

2.1 Опис CASE засобів моделювання інформаційних систем

Підхід до вибору інформаційних технологій для проектування та розробки інформаційної системи звучить досить системно і логічно. Використання CASE-засобів для системного і функціонального моделювання, таких як AllFusion Process Modeler, допомагає у створенні абстрактної структури системи і визначенні її функціональних можливостей. IBM Rational Rose для моделювання діаграм Use Case також є оптимальним вибором, оскільки він допомагає у визначенні вимог користувачів і функціональності системи.

Для побудови логічної моделі даних і створення фізичної бази даних використання інформаційних технологій, таких як AllFusion Data Modeler і MySQL Workbench, є розумним вибором. Вони допоможуть у проектуванні структури бази даних і її оптимізації [22].

Підхід до вибору інформаційних технологій враховує потреби кожної фази проектування та розробки інформаційної системи і використовує відповідні інструменти для кожного етапу. Це сприяє ефективності та успішності даного проекту.

Розробка функціональної моделі для системи електронної комерції і аналізу даних за допомогою стандартів IDEF0 та Data Flow Diagrams (DFD) буде корисним інструментом для формулювання функціональних вимог до заданого вебсайту. Розглянемо кожен стандарт окремо:

1. Стандарт IDEF0 (Integrated Definition Function Modeling):

- Цей стандарт дозволить створити ієрархічну структуру функцій системи, яка відобразить взаємозв'язки між різними функціями.
- Дозволяє детально описати функції вебсайту, такі як обробка замовлень, управління товаром, аналіз даних тощо, і визначити їх взаємозв'язок.

2. Стандарт Data Flow Diagrams (DFD):

- DFD допоможе зобразити потоки даних у системі, що дозволяє краще зрозуміти та проаналізувати, як дані переміщуються через веб-сайт та як вони обробляються.

- Дозволяє визначити різноманітні процеси, такі як реєстрація користувачів, оновлення товарів, обробка платежів тощо, і зв'язати їх з потоками даних.

Загальною метою використання цих стандартів є створення ясної та конкретної моделі вашої системи, яка відображає всі необхідні функції та процеси, а також потоки даних. Це допоможе уточнити вимоги до веб-сайту та забезпечити зрозумілість для всіх учасників проекту.

Використання діаграм потоків даних (DFD) дозволяє детально проаналізувати, як дані обробляються та переміщуються у системі. Ось декілька ключових переваг використання DFD [24]:

1. Аналіз функцій існуючої системи. DFD допомагає визначити, як кожна функція або процес у системі обробляє дані. Це дозволяє зрозуміти поточний стан системи та виявити можливість її вдосконалення.

2. Визначення функціональних вимог до нової системи. DFD дозволяє чітко визначити, які функції має виконувати нова система та як вони повинні взаємодіяти між собою. Це сприяє визначенню обсягу проекту та розробці чітких функціональних вимог.

3. Визначення вимог до структури даних. DFD допомагає визначити, які дані потрібні для виконання функцій системи та як вони повинні бути організовані та збережені. Це допомагає розробити структуру бази даних та визначити типи та формати даних, які потрібні для системи.

В цілому, використання DFD дозволяє структурувати та уточнювати вимоги до системи, що допомагає забезпечити успішну розробку та реалізацію проекту.

Використання діаграм варіантів використання (Use Case) дозволяє зрозуміти, як користувачі будуть взаємодіяти з системою та як система реагуватиме на їхні запити. Ось декілька переваг використання цих діаграм:

1. Зрозуміння потреб користувачів. Діаграми варіантів використання допомагають проаналізувати потреби та очікування користувачів щодо системи.

Це дозволяє розробникам краще зрозуміти, як система повинна працювати для задоволення потреб користувачів.

2. Уточнення функціональних вимог. Діаграми варіантів використання допомагають уточнити функціональні вимоги до системи, визначити основні функції та можливості, які вона повинна надавати користувачам.

3. Комунікація із зацікавленими сторонами. Використання діаграм варіантів використання сприяє ефективній комунікації зі зацікавленими сторонами, такими як замовниками та користувачами системи. Це допомагає всім сторонам зрозуміти, як система буде працювати та як вона задовольнить їхні потреби.

4. База для тестування. Діаграми варіантів використання можуть використовуватися як основа для розробки тестових сценаріїв, що допомагає забезпечити відповідність системи вимогам користувачів.

Так, використання діаграм варіантів використання (Use Case Diagrams) може бути важливою складовою при розробці та оцінці моделей прогнозування в системах електронної комерції. Діаграми варіантів використання дозволяють визначити функціональність системи та її взаємодію з користувачами або іншими системами через різні сценарії використання.

В контексті моделей прогнозування, діаграми варіантів використання можуть допомогти ідентифікувати основні функції та взаємодії системи з різними типами користувачів. Наприклад, вони можуть відобразити, як система отримує дані для прогнозування, які процеси або алгоритми використовуються для аналізу цих даних та як результати прогнозу передаються користувачам чи іншим системам.

Діаграми варіантів використання також можуть допомогти в ідентифікації потенційних проблем або недоліків у процесі прогнозування та взаємодії з системою. Вони сприяють кращому розумінню потреб користувачів та визначенню оптимальних шляхів покращення функціональності системи[17].

Отже, використання діаграм варіантів використання є важливою складовою для розробки та оцінки моделей прогнозування в системах електронної комерції, оскільки вони допомагають узгодити функціональність системи з потребами користувачів та забезпечити її ефективне функціонування.

Так, діаграма класів (Class Diagram) є центральною складовою методології об'єктно-орієнтованого аналізу і проектування (ООА/ООП). Вона використовується для візуалізації структури системи та взаємозв'язків між класами, об'єктами та їх атрибутами та методами.

Діаграми класів дозволяють моделювати структуру даних і функцій системи з точки зору об'єктно-орієнтованого підходу. Вони відображають класи програмного забезпечення, атрибути кожного класу, методи (або функції), що визначені для кожного класу, та зв'язки між класами. Це дозволяє розуміти структуру системи, ідентифікувати класи, які взаємодіють між собою, та сприяє розробці більш ефективного та легко зрозумілого коду.

Діаграми класів є потужним інструментом для аналізу та проектування системи, оскільки вони допомагають встановити правильну архітектуру програмного забезпечення, спростити розробку, зрозуміти залежності між компонентами системи та полегшити комунікацію між учасниками проекту. Вони є важливою ланкою у методології ООА/ООП і допомагають забезпечити якість та ефективність розробленого програмного забезпечення.

Так, діаграми послідовності (Sequence Diagrams) визначають часову послідовність подій та зв'язків між об'єктами, які відбуваються в рамках варіанту використання. Вони демонструють, як об'єкти взаємодіють один з одним у певні моменти часу, показуючи порядок викликів методів та передачу повідомлень між ними.

Діаграми послідовності дозволяють уявити, як об'єкти спілкуються між собою в ході виконання певного сценарію. Вони відображають об'єкти у вигляді вертикальних ліній (або "життєвих ліній"), а взаємодії між ними у вигляді стрілок, що показують порядок викликів методів та передачу повідомлень. Також на діаграмі можуть бути вказані умовні оператори, петлі та інші конструкції, які використовуються для визначення логіки взаємодії [18].

Ці діаграми є корисним інструментом для аналізу та проектування системи, оскільки вони дозволяють зрозуміти, які кроки виконуються та які об'єкти взаємодіють у варіанті використання. Вони допомагають ідентифікувати

потенційні проблеми або неясності у взаємодії між об'єктами та забезпечують чітке розуміння логіки та порядку виконання сценарію.

Діаграми стану (Statechart diagrams) - це інструмент моделювання, який використовується для відображення різних станів, які може перебувати об'єкт або система, і переходів між цими станами. Вони є частиною Unified Modeling Language (UML) та використовуються для моделювання поведінки системи.

У діаграмах стану об'єкти або системи зображуються як стани, що можуть змінюватися з часом. Кожен стан представляє певний внутрішній стан або умову системи, а переходи між станами показують події або умови, які спричиняють зміну стану. Діаграми стану дозволяють візуалізувати реакції системи на вхідні події та управління її станами [18].

Діаграми стану широко використовуються для моделювання складних систем, таких як програмне забезпечення, автоматизовані системи управління та інші. Вони допомагають розуміти різні стани, взаємозв'язки між ними та поведінку системи в цілому.

Наприклад, у програмному забезпеченні діаграми стану можуть використовуватися для моделювання життєвого циклу об'єктів, таких як користувацькі сесії, робочі процеси або стани об'єктів у системі. Вони допомагають розуміти, як система поводить себе в різних ситуаціях та які події чи умови спричиняють зміну стану.

2.2 Технології проектування бази даних

При проектуванні баз даних використовуються різні технології та методології, які допомагають створити ефективну та надійну структуру для зберігання даних. Ось деякі з найпоширеніших технологій проектування баз даних:

1. Реляційна модель. Реляційна модель є основою багатьох баз даних і використовує таблиці для організації даних у вигляді рядків і стовпців. Ця модель забезпечує зв'язки між таблицями за допомогою ключів, що дозволяє ефективно виконувати операції зв'язку і запити до даних.

2. Схема бази даних (Database Schema). Схема бази даних визначає структуру бази даних, включаючи таблиці, поля, типи даних та зв'язки між ними. Вона встановлює правила для зберігання та організації даних у базі даних.

3. Мови створення запитів (Query Languages). Мови створення запитів, такі як SQL (Structured Query Language), дозволяють виконувати операції зчитування та запису даних у базу даних. Вони використовуються для створення запитів, оновлення, видалення та аналізу даних.

4. Нормалізація бази даних (Database Normalization). Це процес організації даних у базі даних таким чином, щоб уникнути аномалій і забезпечити ефективність та цілісність даних. Нормалізація допомагає уникнути дублювання даних і забезпечити їхню консистентність [15].

5. Індексція (Indexing). Індеси допомагають швидко здійснювати пошук і фільтрацію даних, створюючи структури даних, які швидко вказують на місцезнаходження конкретних записів у базі даних.

6. Транзакції (Transactions). Транзакції визначають групу операцій, які повинні бути виконані як єдине ціле. Це дозволяє забезпечити цілісність та надійність даних у випадку відмови або помилки.

7. ORM (Object-Relational Mapping). ORM дозволяє зв'язувати об'єктно-орієнтовані моделі програм з реляційною базою даних, що спрощує взаємодію з даними за допомогою об'єктів та класів.

Ці технології і методології є важливими для ефективного та надійного проектування баз даних і допомагають забезпечити ефективну організацію та управління даними.

2.3 Технології для реалізації інтерфейсу розроблюваної системи

Так, розробка клієнтської частини системи на основі Razor уявлень з використанням мови тегів для написання гіпертекстових документів HTML є зручним та ефективним підходом, особливо в контексті веб-розробки на платформі .NET.

Razor є мовою шаблонів, яка дозволяє вбудовувати код C# або Visual Basic безпосередньо в HTML-файли. Це дає змогу створювати динамічний контент на стороні клієнта, який генерується на сервері. Основна перевага використання Razor у порівнянні з іншими підходами полягає в зручності синтаксису та здатності легко інтегруватися з іншими функціональними можливостями платформи .NET.

Деякі переваги використання Razor включають [16, с.96]:

1. Чистота і читабельність коду. Razor дозволяє вбудовувати код C# безпосередньо в HTML, що робить код більш зрозумілим та легким для обслуговування.

2. Використання мови тегів. Використання мови тегів для створення HTML-шаблонів дозволяє швидко та зручно створювати інтерфейс користувача без необхідності робити багато вбудованих скриптів.

3. Підтримка IntelliSense і від лагодження. Розробники можуть користуватися підтримкою IntelliSense та відлагодженням у середовищі Visual Studio, що спрощує процес викладання та відлагодження веб-додатків.

4. Можливості реюзуння. Разом з чіткою розділеністю логіки та представлення Razor дозволяє створювати відокремлені компоненти, які можна легко повторно використовувати у різних частинах системи.

Отже, використання Razor уявлень для розробки клієнтської частини системи є зручним та ефективним підходом, особливо для проектів, які базуються на платформі .NET.

2.4 Технології для реалізації інформаційної системи

Використання мови програмування C# у поєднанні з шаблоном розробки веб-додатків MVC (Model-View-Controller) є дуже ефективним підходом до розробки веб-додатків, зокрема для систем електронної комерції.

У шаблоні MVC відповідальність за різні частини системи розділяється між трема компонентами [22]:

1. Модель (Model). Моделі відповідають за логіку додатку та управління даними. Вони виконують верифікацію даних, передачу їх між уявленнями та контролерами. Моделі також формують логіку системи, визначаючи, як дані обробляються та як функціонує додаток.

2. Уявлення (View). Уявлення забезпечують представлення даних користувачеві у зрозумілому вигляді. Це може бути просто шаблон, який відображує дані, або код, який обробляє дані перед їх відображенням. Уявлення відокремлюють від логіки програми, що дозволяє легко змінювати вигляд без впливу на логіку додатку.

3. Контролер (Controller). Контролери відповідають за обробку запитів користувача та взаємодію з моделлю та уявленнями. Вони приймають запити від користувача, виконують відповідні дії та визивають відповідні методи та функції моделей.

Цей підхід дозволяє ефективно організувати розробку веб-додатків, забезпечуючи чітке розмежування функцій та відповідальностей між різними компонентами системи.

2.5 Технології аналізу даних у системі

Використання бібліотеки машинного навчання ML.NET в системі електронної комерції для прогнозування обсягів продажів є дуже перспективним підходом. ML.NET дозволяє легко і ефективно інтегрувати функціональність машинного навчання в додатки на платформі .NET [24].

Основні переваги використання ML.NET для розробки прогнозування обсягів продажів включають:

1. Простота і зручність. ML.NET надає простий та зрозумілий інтерфейс для розробки моделей машинного навчання, що дозволяє швидко створювати, тренувати та застосовувати моделі.

2. Широкий вибір алгоритмів. Бібліотека містить різноманітні алгоритми машинного навчання, включаючи лінійну регресію, дерева рішень, випадкові ліси

та багато інших, що дозволяє вибрати найбільш підходящий алгоритм для конкретної задачі прогнозування.

3. Інтеграція з екосистемою .NET. ML.NET ідеально підходить для розробників, які вже працюють з платформою .NET, оскільки вона інтегрується з існуючими додатками та сервісами на цій платформі.

4. Відкритість та спільнота. ML.NET є відкритим проектом з активною спільнотою розробників, що забезпечує підтримку та постійний розвиток бібліотеки.

Загалом, використання ML.NET для прогнозування обсягів продажів в системі електронної комерції є раціональним вибором, який може значно полегшити та прискорити розробку та впровадження аналітичних функцій в додаток.

В основі цього процесу лежить ідея навчання комп'ютера вирішувати завдання за допомогою даних, які він отримує. ML.NET реалізує цей підхід, дозволяючи створювати, навчати та застосовувати моделі машинного навчання в вашому додатку.

Центральною частиною ML.NET є саме модель машинного навчання, яка визначає кроки, необхідні для трансформації вхідних даних у прогнозовані дані. Модель можна навчити, використовуючи певний алгоритм на вибір, або імпортувати попередньо навчені моделі з інших середовищ, таких як TensorFlow або ONNX.

Коли модель навчена, це дозволить легко додати її до програми і використовувати для здійснення прогнозів. Це надає можливість створювати додатки, які можуть вирішувати складні завдання аналізу даних та приймати відповідні рішення на основі навчених моделей.

3 ПОСТАНОВКА ЗАДАЧІ

3.1 Призначення розробки обраної інформаційної системи

1. Збір даних. Зібрати дані про продажі відеоігор у форматі електронних цифрових ключів. Ці дані можуть включати такі атрибути, як ідентифікатор продукту, кількість проданих ключів, дата продажу, ціна продажу тощо.

2. Побудова бази даних. Створити базу даних, що буде зберігати ці дані. База даних повинна бути оптимізована для швидкого доступу та ефективного аналізу.

3. Аналіз попиту. Визначити попит на різні відеоігри шляхом аналізу кількості проданих ключів. Виявити найбільш популярні та непопулярні ігри.

4. Прогнозування продажів. Розробити модель прогнозування для передбачення майбутніх продажів відеоігор. Використовуючи історичні дані, модель може передбачати популярність певних ігор у майбутньому.

5. Контроль запасів. Моніторити рівень запасів кожної відеоігри та спрогнозувати, коли необхідно здійснити поповнення асортименту на складі.

6. Аналіз ефективності акцій. Оцінити ефективність різних маркетингових акцій і знижок на продажі відеоігор. Визначити, які заходи сприяють збільшенню обсягів продажів.

7. Класифікація ігор. Провести класифікацію відеоігор за різними категоріями (жанрами, рейтингом тощо) для аналізу популярності різних категорій серед покупців.

Ці завдання допоможуть створити систему аналізу даних, яка сприятиме кращому розумінню попиту, управлінню запасами та прийняттю обґрунтованих рішень щодо маркетингових стратегій у вашому інтернет-магазині відеоігор.

Розглянемо області діяльності, які підлягають автоматизації робіт відповідно до вказаних задач і функцій підприємства та його відділу продажу [10, с. 32]:

1. Забезпечення користувачів і клієнтів необхідним інструментарієм для перегляду і вибору товарів та оформлення замовлення:

- Розробка веб-сайту або додатку для інтернет-магазину, який забезпечує зручний перегляд асортименту, можливість фільтрування товарів та зручний процес оформлення замовлення.

2. Впровадження та реалізація маркетингових засобів та стратегій:

- Використання системи управління взаємодією з клієнтами (CRM) для збереження і аналізу даних про клієнтів.

- Реалізація програми лояльності через відстеження покупок та винагороди за повторні звернення.

- Використання інструментів аналітики для визначення ефективності маркетингових стратегій.

3. Отримання детальної інформації щодо прогнозів продажів товарів:

- Використання методів аналізу даних, наприклад, за допомогою ML.NET для прогнозування обсягів продажів.

4. Отримання прибутку від реалізації товарів:

- Впровадження системи обліку фінансової діяльності для відстеження прибутку від продажу товарів.

5. Функції відділу продажу:

- Автоматизація обліку продажу товару та відстеження темпів продажів за допомогою спеціалізованої CRM-системи.

- Введення промоакцій, знижок та персоналізованих пропозицій через систему управління маркетингом.

Проведення аналізу трудомісткості робіт у підприємстві та його підрозділах допоможе визначити, які саме процеси слід автоматизувати для оптимізації діяльності та підвищення ефективності бізнесу.

Реалізація інформаційної системи у вигляді веб-ресурсу для відділу продажів дійсно може стати ефективним рішенням для полегшення взаємодії з клієнтами та оптимізації бізнес-процесів. Основні функції, такі як надання списку товарів, фільтрування, оформлення замовлення та програми лояльності, можуть бути легко і зручно реалізовані у веб-середовищі [10, с. 36].

Щодо функції аналізу даних, прогнозування продажів є важливою задачею, яка може допомогти підприємству приймати обґрунтовані рішення щодо стратегій

розвитку. Для прогнозування продажів зазвичай використовуються різноманітні методи аналізу даних, такі як регресійний аналіз, часові ряди, або навіть штучні нейронні мережі.

Щоб провести прогноз продажів, необхідно мати вхідні дані про товари та їх характеристики, а також історичні дані про продажі. Кожна гра може бути описана рядком параметрів, які можуть включати такі дані, як назва гри, рік випуску, жанр, рейтинг, вартість, тощо.

Після збору та обробки цих даних можна застосувати відповідні методи аналізу даних для прогнозування майбутніх продажів та розробки стратегій розвитку підприємства.

Такий підхід до розробки системи для прогнозування продажів і аналізу даних дійсно допоможе власнику підприємства приймати обґрунтовані рішення щодо стратегій розвитку. Для цього потрібно провести дослідження методів прогнозування даних, включаючи моделі регресії і моделі часових рядів, порівняти їх ефективність і обрати найбільш підходящий варіант.

Моделі регресії можуть бути корисні для прогнозування продажів на основі різних характеристик товарів і факторів, які впливають на їхній попит. З іншого боку, моделі часових рядів можуть бути корисними для прогнозування продажів на основі історичних даних про продажі у певний період часу.

Імплементація цих моделей у систему електронної комерції дозволить автоматизувати процес прогнозування продажів і аналізу даних, що значно полегшить прийняття рішень щодо подальшого розвитку підприємства. Така система забезпечить власнику потрібну інформацію для ефективного управління бізнесом і досягнення стратегічних цілей.

3.2 Мета впровадження методів аналізу даних у систему електронної комерції

Ціль створення моделей прогнозування для власника бізнесу включає в себе наступні аспекти [11, с. 95]:

1. Виявлення аномалій у продажах для планування закупівель. Моделі прогнозування можуть допомогти виявити незвичайні тенденції або аномалії у продажах, що дозволить вчасно реагувати і планувати закупівлі товарів для забезпечення попиту.

2. Оцінка впливу змін цінових діапазонів на обсяги продажів. Моделі можуть допомогти прогнозувати, як зміна цін впливає на кількість проданих товарів, що допоможе оптимізувати ціноутворення і маркетингові стратегії.

3. Автоматизація звітності і відстеження результатів роботи системи. Застосування моделей прогнозування дозволить автоматизувати процеси збору даних, аналізу та формування звітності, що спростить відстеження результатів роботи системи та вирішення адміністративних завдань.

4. Залучення нових користувачів шляхом передбачення їх вподобань.. Аналіз даних допоможе власнику бізнесу розуміти попит користувачів та їхні вподобання, що дозволить персоналізувати пропозиції і залучити нових клієнтів.

5. Аналіз даних для формування стратегії розвитку бізнесу. Інформація, отримана з моделей прогнозування, може бути використана для розробки стратегій розвитку бізнесу, вдосконалення маркетингових підходів та управлінських рішень.

3.3 Функціональні можливості розроблюваної системи

Основні функціональні можливості системи прогнозування продажів включають наступне [11, с. 100]:

1. Перегляд результатів прогнозу на певний період часу за допомогою моделі регресії для всіх категорій і цінових діапазонів. Користувач може переглядати прогнози продажів на певний період часу, побудовані за допомогою моделі регресії, для всіх категорій товарів та різних цінових діапазонів.

2. Перегляд результатів прогнозу на певний період часу за допомогою моделі часового ряду для всіх категорій і цінових діапазонів. Користувач може переглядати прогнози продажів на певний період часу, побудовані за допомогою моделі часового ряду, для всіх категорій товарів та різних цінових діапазонів.

3. Перегляд результатів прогнозу на певний період часу за допомогою обох моделей прогнозування для певної категорії товарів і всіх цінових діапазонів. Користувач може переглядати результати прогнозу продажів, використовуючи як модель регресії, так і модель часового ряду, для певної категорії товарів та всіх доступних цінових діапазонів.

4. Перегляд результатів прогнозу на певний період часу за допомогою обох моделей прогнозування для всіх категорій і певного цінового діапазону. Користувач може переглядати результати прогнозу продажів, використовуючи як модель регресії, так і модель часового ряду, для всіх категорій товарів та певного цінового діапазону.

5. Перегляд результатів прогнозу на певний період часу за допомогою обох моделей прогнозування для певної категорії товарів і певного цінового діапазону. Користувач може переглядати результати прогнозу продажів, використовуючи як модель регресії, так і модель часового ряду, для певної категорії товарів та певного цінового діапазону.

3.4 Основні не функціональні вимоги до системи

Нефункціональні вимоги до інтерфейсу користувача системи електронної комерції включають такі пункти:

1. Інтуїтивна зрозумілість інтерфейсу. Інтерфейс користувача повинен бути легким для розуміння та використання навіть без попереднього навчання.

2. Сумісність інтерфейсів з різними розширеннями екрану. Усі основні сторінки системи повинні коректно відображатися на екранах з мінімальним співвідношенням розмірів 1024 x 768, забезпечуючи адекватну роботу на різних пристроях і збільшуючи доступність для користувачів.

3. Автоматичне оновлення головної сторінки. Головна сторінка інтерфейсу повинна автоматично оновлюватися, щоб завжди відображати актуальну інформацію про наявність товарів та вміст кошика користувача, що допоможе зберегти актуальність і спростити процеси замовлення.

Нефункціональні вимоги до браузерів та продуктивності системи електронної комерції включають такі пункти:

Вимоги до підтримки браузерів можна сформулювати так:

1. Google Chrome версії 112: Цей браузер широко використовується і часто оновлюється, тому необхідно перевірити, щоб система працювала безперешкодно у версії 112.
2. Opera версії 98: Хоча Opera може не бути так поширеним, як Chrome або Firefox, вона все ще використовується багатьма користувачами, тому важливо забезпечити сумісність з версією 98.
3. Mozilla Firefox версії 110: Firefox також є досить популярним браузером, тому важливо переконатися, що система працює на версії 110 без проблем.
4. Safari версії 16: Якщо ваша система призначена для користувачів macOS або iOS, важливо, щоб вона працювала на останній версії Safari, яка підтримується на цих платформах.
5. Microsoft Edge версії 112: Оскільки Microsoft Edge тепер базується на тому ж рушії, що і Google Chrome, багато користувачів перейшли на нього. Важливо переконатися, що система сумісна з версією 112.

Вимоги до продуктивності:

1. Система повинна здатна зберігати і обробляти дані протягом мінімум одного року для ефективного аналізу і прогнозування продажів.

2. Максимальне навантаження системи повинно становити не менше ніж 300 користувачів одночасно для забезпечення продуктивності платформи.

3. Форми інтерфейсу користувачів повинні відображатися не більше, ніж за 1-2 секунди для забезпечення швидкості реакції і зручного користування.

4. База даних повинна бути спроектована таким чином, щоб забезпечувати ефективну роботу при великому навантаженні користувачів, не сповільнюючи процес пошуку даних.

5. Система повинна аналізувати дані, отримані в процесі роботи, після кожного замовлення і підтримувати актуальну інформацію про результати аналізу даних у базі даних.

3.5 Основні вимоги до безпеки

Отримання вимог до безпеки системи - це важливий етап в розробці будь-якої інформаційної системи. Ось кілька загальних вимог, які можуть бути враховані:

1. Аутентифікація і авторизація. Система повинна мати механізми для аутентифікації користувачів, щоб переконатися, що лише авторизовані користувачі мають доступ до конфіденційної інформації або функціональності.

2. Контроль доступу. Налаштування прав доступу до різних частин системи залежно від ролі користувача або інших факторів.

3. Шифрування даних. Застосування шифрування для захисту конфіденційної інформації під час передачі через мережу або зберігання в базі даних.

4. Захист від вразливостей. Регулярне виявлення та виправлення вразливостей у системі, таких як критичні вразливості безпеки, атаки переповнення буфера, SQL-ін'єкції тощо.

5. Моніторинг та журналювання. Встановлення системи моніторингу для виявлення ненормальної активності та журналювання подій для подальшого аналізу випадків порушення безпеки.

6. Резервне копіювання та відновлення даних. Регулярне створення резервних копій даних та планів відновлення, щоб забезпечити можливість відновлення системи в разі втрати даних або інших негараздів[5].

7. Забезпечення безпеки на рівні додатку. Застосування кращих практик розробки програмного забезпечення для забезпечення безпеки на рівні додатку, таких як захист від переповнення буфера, перевірка на валідність введення, використання параметризованих запитів SQL тощо.

8. Оновлення та патчі. Постійне оновлення системи та встановлення патчів безпеки для запобігання використанню відомих вразливостей.

Це лише загальні вимоги, і конкретні заходи безпеки повинні бути визначені в контексті конкретної системи, її потреб та загроз.

4 МАТЕМАТИЧНИЙ ОПИС АЛГОРИТМІВ ПРОГНОЗУВАННЯ

4.1 Алгоритм методу SDCA

SDCA (Stochastic Dual Coordinate Ascent) - це ефективний алгоритм для розв'язання регуляризованих задач лінійної класифікації та регресії з великими наборами даних. Він належить до класу стохастичних градієнтних методів і є варіантом методу координатного спуску.

Основна ідея алгоритму SDCA полягає в тому, щоб ітераційно оновлювати компоненти двоїстого вектора, замість безпосереднього оновлення вектора ваг моделі. Ця процедура здійснюється шляхом випадкового вибору однієї координати двоїстого вектора та її оптимізації з використанням замкненого аналітичного рішення. Такий підхід дозволяє значно прискорити процес збіжності порівняно зі стандартними методами стохастичного градієнтного спуску [24].

Нижче наведено узагальнений алгоритм методу SDCA для регуляризованої задачі лінійної класифікації або регресії:

1. Ініціалізація: Починаємо з двоїстого вектора α_i , для всіх $i=1, \dots, n$ який зазвичай задається нульовим.
2. Повторюємо T ітерацій.
3. Випадково вибираємо індекс i з множини $\{1, 2, \dots, n\}$.
4. Знаходимо індекс i , який максимально збільшує різницю між втратою і дуальним добутком. Це дозволяє нам визначити, яку координату вектора α слід оновити.
5. Оновлюємо i -ту координату двоїстого вектора α за певною формулою, яка включає втрату та регуляризацію. Інші координати залишаються незмінними.
6. Результат: Після T ітерацій отримуємо остаточний двоїстий вектор α .
7. Ваги моделі: Обчислюємо вектор ваг моделі w за допомогою остаточного двоїстого вектора α та навчальних даних.

Перевагами алгоритму SDCA є його ефективність для великих наборів даних, можливість паралелізації та гарна збіжність. Він широко використовується в машинному навчанні для задач класифікації та регресії з лінійними моделями, таких як логістична регресія чи лінійна регресія з регуляризацією.

Регуляризована мінімізація втрат лінійних предикторів - це метод оптимізації, який використовується для знаходження оптимальних значень параметрів моделі в умовах, коли є багато потенційних предикторів (або ознак), існує можливість перенавчання та виникає проблема мультиколінеарності. Цей метод включає в себе введення штрафних членів у функцію втрат з метою зменшення значень параметрів моделі та зниження перенавчання.

Основні типи регуляризації, які часто використовуються для мінімізації втрат лінійних предикторів, це:

1. L1 (Lasso) регуляризація: Додає штрафний член, який залежить від суми абсолютних значень коефіцієнтів моделі. L1-регуляризація сприяє розрідженню параметрів моделі, що означає, що деякі коефіцієнти можуть стати нульовими, тим самим виконуючи відбір ознак.

2. L2 (Ridge) регуляризація: Додає штрафний член, який залежить від квадратів значень коефіцієнтів моделі. L2-регуляризація контролює величину коефіцієнтів, сприяючи зменшенню мультиколінеарності та стабілізації моделі.

Ці методи допомагають уникнути перенавчання та підвищують стабільність моделі, що особливо важливо у випадку роботи з великою кількістю ознак. Вони можуть бути використані в багатьох областях, включаючи прогнозування, класифікацію, регресію та інші завдання машинного навчання [17]:

Для того щоб знайти мінімум функціоналу $P(w)$, де w - набір коефіцієнтів регресії, потрібно спочатку сформулювати сам функціонал $P(w)$. Зважаючи на надані вектори x_1, \dots, x_n , послідовність скалярних опуклих функцій ϕ_1, \dots, ϕ_n , і параметр регуляризації λ , можемо визначити функціонал $P(w)$ наступним чином:

$$P(w) = \left[\frac{1}{n} \sum_{i=1}^n \phi_i(w^T x_i) + \frac{\lambda}{2} \|w\|^2 \right].$$

Тут w^T - транспонований вектор w , $\|w\|^2$ - квадрат його норми.

Метою є мінімізація цього функціоналу $P(w)$. Для цього можна використовувати різні методи оптимізації, такі як градієнтний спуск, метод Ньютона, метод квазі-Ньютона тощо. Зазвичай для виконання мінімізації використовують ітераційні методи, які знаходять мінімум шляхом поступового оновлення коефіцієнтів w до тих пір, поки не буде досягнуто задовільного значення.

Загальний алгоритм мінімізації функціоналу $P(w)$ може виглядати наступним чином:

1. Ініціалізувати початкове значення для коефіцієнтів w .
2. Повторювати наступні кроки до досягнення збіжності або досягнення максимальної кількості ітерацій: а. Обчислити градієнт функціоналу $P(w)$. б. Оновити коефіцієнти w в напрямку, протилежному градієнту, з урахуванням кроку оптимізації.
3. Повернути останнє значення коефіцієнтів w , яке було отримано.

Цей алгоритм є загальним підходом до мінімізації функціоналу, але конкретні деталі реалізації можуть варіюватися залежно від вибраного методу оптимізації.

Так, стохастичний градієнтний спуск (SGD) може бути привабливим варіантом для вирішення задач оптимізації з великою кількістю даних. Однак, він має свої недоліки, які варто врахувати:

Відсутність чіткої умови зупинки: SGD може продовжувати свою роботу, не досягаючи заданої точності, і не завершати роботу відповідно до певних критеріїв.

Агресивність на початку оптимізації: У ранніх етапах оптимізації SGD може бути занадто агресивним, що може призвести до швидкого збільшення втрат і недосягнення оптимального рішення [17].

Повільне зближення до оптимального рішення: Хоча SGD може швидко досягнути помірної точності, його зближення може стати повільним при досягненні більш високої точності, особливо коли значення параметра λ дуже мале.

Ці недоліки варто враховувати при виборі методу оптимізації для задачі регуляризованої мінімізації втрат лінійних предикторів.

Так, двокоординатне сходження (DCA) є альтернативним підходом до оптимізації, який вирішує задачу двоїстості формули безпосередньо. Основна ідея полягає в тому, щоб розглядати кожен компонент вектору коефіцієнтів w окремо, мінімізуючи функціонал лише по одній координаті за раз.

Процес DCA може бути описаний наступним чином:

Початкове значення вектора коефіцієнтів w ініціалізується деяким чином.

На кожній ітерації DCA вибирає одну з координат w_i і мінімізує функціонал $P(w)$ лише по цій координаті, залишаючи всі інші фіксованими.

Після мінімізації функціоналу по координаті w_i оновлюється значення цієї координати.

Цей процес повторюється для кожної з координат w_i на деякій кількості ітерацій або до досягнення збіжності.

Значення вектора коефіцієнтів w , отримане після закінчення всіх ітерацій, приймається як розв'язок задачі оптимізації.

Двокоординатне сходження є ефективним методом оптимізації в разі, коли функціонал має просту структуру, що дозволяє розділити його між координатами. Він може бути особливо ефективним у випадках, коли функціонал є опуклим або має певні інші властивості, які сприяють ефективному мінімізації.

Зокрема, для кожного i нехай φ_i^* є опукла спряжена φ_i , а саме $\varphi_i^*(u) =$

$\max_z (zu - \varphi_i(z))$. Проблема двоїстості набуває вигляду $\max D(\alpha)$, де:

$$D(\alpha) = \left[\frac{1}{n} \sum_{i=1}^n -\varphi_i(-\alpha_i) - \frac{\lambda}{2} \left\| \frac{1}{\lambda n} \sum_{i=1}^n \alpha_i x_i \right\|^2 \right]. \quad (4.2)$$

У формулі (4.2) в алгоритмі SDCA двоїсте цільове значення включає різну двоїсту змінну α_j для кожного прикладу x_j у навчальному наборі даних.

На кожній ітерації t алгоритму SDCA виконуються наступні кроки:

1. Випадково вибирається один приклад даних (x_i, y_i) з навчальної вибірки.
2. Знаходиться індекс i , який максимізує різницю між функцією втрат і дуальним добутком для вибраного прикладу x_i .
3. Оновлюється лише двоїста змінна α_i , що відповідає прикладу x_i , який максимізує цю різницю. Ця змінна оновлюється шляхом оптимізації двоїстого цільового значення в формулі (4.2) відносно α_i , тоді як всі інші двоїсті змінні α_k ($k \neq i$) залишаються незмінними.

Таким чином, на кожній ітерації алгоритм SDCA оптимізує двоїсте цільове значення лише відносно однієї двоїстої змінної α_i , пов'язаної з прикладом x_i , який найбільше сприяє зменшенню цільової функції. Всі інші двоїсті змінні α_k ($k \neq i$) зберігають свої поточні значення з попередньої ітерації.

Це робиться для ефективного оновлення двоїстого вектора α на кожній ітерації, не обчислюючи всі двоїсті змінні одночасно, що значно прискорює

збіжність алгоритму порівняно із звичайним методом двоїстого координатного спуску.

Отже, ваше твердження про те, що SDCA оптимізує двоїсте цільове значення лише відносно однієї двоїстої змінної на ітерацію, зберігаючи решту двоїстих змінних незмінними, є абсолютно вірним і відображає суть роботи цього алгоритму.

Цей алгоритм є стохастичною версією DCA, оскільки на кожній ітерації випадково обирається одна змінна для оптимізації. Це дозволяє більш ефективно рухатися в напрямку оптимуму, особливо у великих просторах даних або у випадку, коли функція втрат має складну структуру

$$-\varphi_i^*(-\alpha_i^{(t-1)} + \Delta\alpha_i) - \frac{\lambda n}{2} \|w^{(t-1)} + (\lambda n)^{-1} \Delta\alpha_i x_i\|^2;$$

$$\alpha^{(t)} = \alpha^{(t-1)} + \Delta\alpha_i e_i;$$

$$w^{(t)} = w^{(t-1)} + (\lambda n)^{-1} \Delta\alpha_i x_i.$$

3. Обчислити середнє значення:

$$\bar{\alpha} = \frac{1}{T - T_0} \sum_{i=T_0+1}^T \alpha^{(t-1)};$$

$$\bar{w} = w(\bar{\alpha}) = \frac{1}{T - T_0} \sum_{i=T_0+1}^T w^{(t-1)}.$$

4. Обчислити випадкове значення:

$$\bar{\alpha} = \alpha^{(t)} \text{ і } \bar{w} = w^{(t)} \text{ де } t \in T_0 + 1, \dots, T.$$

6. На вихід подати w .

4.2 Алгоритм методу регресії Пуассона

Розподіл Пуассона моделює ймовірність подій y (тобто невдачі, смерті, продажі т.д) за формулою [18]:

$$\Pr(Y = y | \mu, t) = \frac{e^{-\mu t} (\mu t)^y}{y!}, y = 0, 1, \dots$$

У розподілі Пуассона параметр μ (або λ) виступає як середня частота рідкісної події на одиницю експозиції t , де експозиція може бути часом, простором, відстанню тощо. Якщо значення експозиції не вказано, зазвичай вважається, що це час. Параметр μ можна інтерпретувати як ризик нового виникнення події протягом певного періоду експозиції t . Ймовірність подій y тоді визначається як:

$$\Pr(Y = y | \mu, t) = \frac{e^{-\mu t} (\mu t)^y}{y!}, y = 0, 1, \dots$$

Розподіл Пуассона має властивість, що його середнє значення і дисперсія рівні.

У регресії Пуассона припускається, що частота впливу Пуассона μ визначається набором k регресора змінних X . Вираз, що зв'язує ці величини:

$$\mu = t \exp(\beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k)$$

Часто $X_1 \equiv 1$ і β_1 називають перерізом. Коефіцієнти регресії

$\beta_1, \beta_2, \dots, \beta_k$, є невідомими параметрами, які отримуються з набору

даних. Їхні значення позначені як b_1, b_2, \dots, b_k .

Використовуючи ці позначення, фундаментальна модель регресії Пуассона для спостереження i -ого значення записується як:

$$\begin{aligned} \Pr(Y_i = y_i | \mu_i, t_i) &= \frac{e^{-\mu_i t_i} (\mu_i t_i)^{y_i}}{y_i!}, \text{ де } \mu_i = t_i \mu(x_i \beta) \\ &= t_i \exp(\beta_1 X_{1i} + \beta_2 X_{2i} + \dots + \beta_k X_{ki}) \end{aligned}$$

Тобто для заданого набору значень змінних регресора результат відповідає розподілу Пуассона.

Коефіцієнти регресії оцінюються за допомогою методу максимальної правдоподібності. Логарифм функції імовірності:

$$\ln[L(y, \beta)] = \sum_{i=1}^n y_i \ln[t_i \mu(x_i \beta)] - \sum_{i=1}^n t_i \mu(x_i \beta) - \sum_{i=1}^n \ln(y_i!)$$

Так, метод ітераційних зважених найменших квадратів є ефективним підходом до знаходження набору коефіцієнтів регресії, які максимізують логарифмічну ймовірність у задачах регуляризованої мінімізації втрат лінійних предикторів.

Зазвичай цей метод використовується для навчання моделей регресії чи класифікації, де необхідно максимізувати ймовірність спостережень. Однак, як правильно зауважено, він може бути повільним для великих обсягів даних через потребу в повному проходженні даних на кожній ітерації.

За сучасних умов, коли комп'ютери мають значну обчислювальну потужність, цей метод все ще залишається ефективним, особливо з використанням оптимізацій та оптимізованих алгоритмів. Такі алгоритми можуть прискорити обчислення і зробити цей підхід прийнятним для великих обсягів даних.

Застосовуючи звичайну теорію максимальної правдоподібності, асимптотичний розподіл оцінок максимальної правдоподібності (MLE) є багатовимірною нормою. Тобто:

$$\hat{\beta} \sim N(\beta, \beta V_{\hat{\beta}}), \text{ де}$$

$$V_{\hat{\beta}} = \left(\sum_{i=1}^n \mu_i x_i x_i' \right)^{-1}$$

Так, в моделі Пуассона передбачається, що середнє значення та дисперсія спостережень рівні. Однак на практиці часто може виникати ситуація, коли дисперсія перевищує середнє значення, що відомо як наддисперсія.

У таких випадках модель може бути модифікована для врахування наддисперсії. Один з способів цього досягнення - введення параметра ϕ , який множить дисперсію. Таким чином, розподіл в моделі стає негауссівським, де дисперсія кожного спостереження має різні значення, але в середньому вона збільшується у ϕ разів.

$$V_{\hat{\beta}} = \phi \left(\sum_{i=1}^n \mu_i x_i x_i' \right)^{-1}, \text{ де}$$

$$\hat{\phi} = \frac{1}{n - k} \sum_{i=1}^n \frac{(y_i - \hat{\mu}_i)^2}{\hat{\mu}_i}$$

4.3 Алгоритм методу дерев регресії

Так, евристика грає важливу роль у побудові дерев рішень, оскільки вона визначає, яким чином вибирати найбільш інформативні атрибути для розділення даних. Більшість сучасних алгоритмів навчання дерев рішень базуються на евристичних підходах, зокрема на евристиці (не)чистоти.

Ця евристика вимірює чистоту або нечистоту підмножин даних після розділення за допомогою певного атрибута. Прибуток інформації (information gain), який визначається на основі цієї евристики, широко використовується як стандартна міра ефективності атрибутів розщеплення.

Цей підхід дозволяє зменшити обчислювальну вартість побудови дерева рішень, а також покращити точність класифікації шляхом вибору найбільш інформативних атрибутів для розділення даних.

$$IG(S, X) = Ent(S) - \sum_x \frac{|S_x|}{|S|} Ent(S_x) \quad (4.3)$$

де S – набір даних навчання,

X – набір атрибутів, x – їхні значення,

S_x – є підмножиною S , що складається з екземплярів $X=x$,

$$Ent(S) = - \sum_{i=1}^{|C|} P_S(c_i) \log P_S(c_i),$$

де $P_S(c_i)$ - оцінюється за відсотком належності екземплярів c_i у S ,

$|C|$ - кількість класів.

Побудова дерев рішень є рекурсивним процесом, де навчальні дані S розбиваються на більш вузькі підмножини на кожному кроці.

У формулі $P_S(c_i)$ c_i представляє клас або мітку, до якої належить вузол. Ця

умовна ймовірність фактично означає $P_S(c_i|x_p)$ для всіх навчальних даних, де x_p - це

набір атрибутів вздовж шляху від поточного вузла до кореня, який називається атрибутами шляху, а x_p - це призначення значень до змінних в x_p .

Це важливо враховувати при побудові дерев рішень, оскільки це дозволяє врахувати усі атрибути шляху при прийнятті рішення про подальше розбиття вузла дерева.

Цей алгоритм побудови швидкого дерева рішень використовує метод обчислення інформаційного приросту для вибору атрибута розщеплення і розбиття навчальних даних на підмножини. Основні кроки алгоритму такі:

1. Перевіряється, чи є множина навчальних даних S порожньою або чистою (тобто всі приклади належать до одного класу) або чи набір атрибутів-кандидатів Π порожній. У цьому випадку повертається побудоване дерево T .
2. Обчислюється ймовірність $P_S(c_i)$ для кожного класу c_i в S .
3. Для кожного атрибута X в Π обчислюється приріст інформації (IG) на основі

формули, яка визначається за допомогою $P_S(c_i|x_p, x)$, де x_p - це атрибути

шляху, а x - значення атрибута X .

4. Вибирається атрибут X з найвищим значенням IG і використовується для розщеплення множини S на непересічні підмножини S_x .
5. Для кожного значення x атрибута X обчислюється дерево T_x , використовуючи рекурсивний виклик FT з видаленням атрибута X з набору атрибутів-кандидатів і підмножину S_x .
6. Кожне з отриманих дерев T_x стає дочірнім елементом вузла, що розщеплюється за атрибутом X .
7. Повертається побудоване дерево T .

Цей алгоритм дозволяє ефективно побудувати дерево рішень, використовуючи інформаційний приріст для вибору оптимального атрибута розщеплення.

4.4 Алгоритм методу прогнозування часових рядів

Сингулярний спектральний аналіз (SSA) - це потужний метод аналізу та прогнозування часових рядів, який використовується в багатьох областях, включаючи метеорологію, економіку, фінанси, медицину та інші. Він поєднує в собі різноманітні підходи, такі як класичний аналіз часових рядів, багатовимірна статистика, геометрія, динамічні системи та обробка сигналів.

Мета SSA - розкласти вихідний часовий ряд на декілька компонентів, які можна інтерпретувати. Ці компоненти включають повільно змінний тренд, коливальні компоненти та "безструктурний" шум. Важливою перевагою SSA є те, що він не вимагає припущень про параметричну модель чи стаціонарність часового ряду, що робить його методом без моделі.

Основна ідея SSA полягає в розкладанні за сингулярними значеннями конкретної матриці, яка побудована за допомогою часового ряду. Цей метод дозволяє отримати компоненти ряду, які можна аналізувати та інтерпретувати, що робить його дуже корисним інструментом для вивчення та прогнозування складних часових рядів.

Процес побудови траєкторної матриці за допомогою сингулярного спектрального аналізу (SSA), який використовується для аналізу та прогнозування часових рядів. Він описується таким чином:

1. Для часового ряду довжиною N будуються вектори довжиною L з відставанням, які складаються з послідовних значень ряду. Наприклад, $X_i = (x_i, x_{i+1}, \dots, x_{i+L-1})$, де $i = 1, 2, \dots, K$ і $K = N - L + 1$.

2. Ці вектори розміщуються в матрицю $X = [X_1, X_2, \dots, X_K]$, яка має розмірність $L \times K$ і відома як "матриця траєкторії". Ця матриця має форму матриці Ганкеля, що означає, що всі елементи вздовж діагоналі рівні.

3. Проводиться розкладання матриці X на сингулярні значення за допомогою сингулярного розкладу $X = U\Sigma V^T$, де U - матриця власних векторів, Σ - діагональна матриця сингулярних значень, V - матриця власних векторів.

4. Обираються перші $l < L$ власних векторів, які відповідають найбільшим сингулярним значенням, і формується l -вимірний підпростір.

5. Вихідні дані проектуються на цей 1-вимірний підпростір, і наступне усереднення по діагоналях дає деяку матрицю Ганкеля X , яка є наближенням до початкової матриці X .

6. Ряд, відновлений з матриці X , задовольняє деякій лінійній рекурентній формулі, яка може бути використана для прогнозування.

5 РОЗРОБКА АРХІТЕКТУРИ ТА КОМПОНЕНТІВ СИСТЕМИ

5.1 Розробка системних вимог

Враховуючи постановку задачі і мету створення системи для прогнозування обсягів продажів в системі електронної комерції, ось деякі системні вимоги до розроблюваної системи:

1. Апаратне забезпечення клієнта:

- Мінімальна частота процесора та кількість оперативної пам'яті для ефективної роботи веб-додатку.

- Сумісність з різними операційними системами на стороні клієнта, такими як Windows, macOS, Linux та іншими.

2. Інтерфейс користувача:

- Зручний та інтуїтивно зрозумілий інтерфейс для користувачів.

- Мінімальні вимоги до роздільної здатності екрану для оптимального відображення інтерфейсу на різних пристроях, таких як комп'ютери, планшети та смартфони.

3. Браузерна сумісність:

- Сумісність з популярними веб-браузерами, такими як Google Chrome, Mozilla Firefox, Safari, Microsoft Edge тощо.

- Відповідність вимогам щодо версій браузерів, що підтримуються.

4. Швидкодія:

- Оптимізація продуктивності системи для швидкого відображення даних та взаємодії з користувачем.

- Мінімальні часові затримки при обробці запитів та відображенні результатів.

5. Безпека:

- Захист доступу до бази даних за допомогою системи авторизації та автентифікації.

- Забезпечення конфіденційності, цілісності та доступності даних.

6. Масштабованість:

- Здатність системи витримувати велику кількість користувачів і оброблювати значні обсяги даних без втрати продуктивності.

7. Доступність:

- Максимальний час безвідмовної роботи системи.
- Механізми резервного копіювання та відновлення даних у випадку аварійної ситуації.

Ці системні вимоги є ключовими для забезпечення ефективності, надійності та зручності використання розробленої системи прогнозування обсягів продажів у сфері електронної комерції.

Так, враховуючи вищезазначені системні вимоги, можна сформулювати конкретні вимоги до серверної та клієнтської частини системи електронної комерції для прогнозування обсягів продажів:

а) Серверна частина:

1. Використання Системи Управління Базами Даних (СУБД) для зберігання та обробки даних про продажі.
2. Організація доступу до бази даних за допомогою системи автентифікації для забезпечення безпеки даних.
3. Здатність СУБД оптимально працювати з великою кількістю одночасних підключень для підтримки високого навантаження.

б) Клієнтська частина:

1. Реалізація інтерфейсу користувача у вигляді веб-сторінок, написаних на HTML, CSS та JavaScript.
2. Відображення інформації про прогнозовані обсяги продажів користувачам через веб-браузер.

в) Інтерфейс доступу:

1. Забезпечення сумісності з популярними веб-браузерами, такими як Google Chrome, Mozilla Firefox, Opera, Safari, Microsoft Edge тощо.
2. Завантаження веб-сторінок через браузер для отримання доступу до функціональності системи.

Дотримання цих вимог допоможе забезпечити ефективну та надійну роботу системи прогнозування обсягів продажів в електронній комерції.

5.2 Розробка моделі потоків даних

Концептуальна DFD є дійсно корисним інструментом для розуміння загальної картини потоків даних у системі та взаємодії між різними її компонентами рис..5.1.

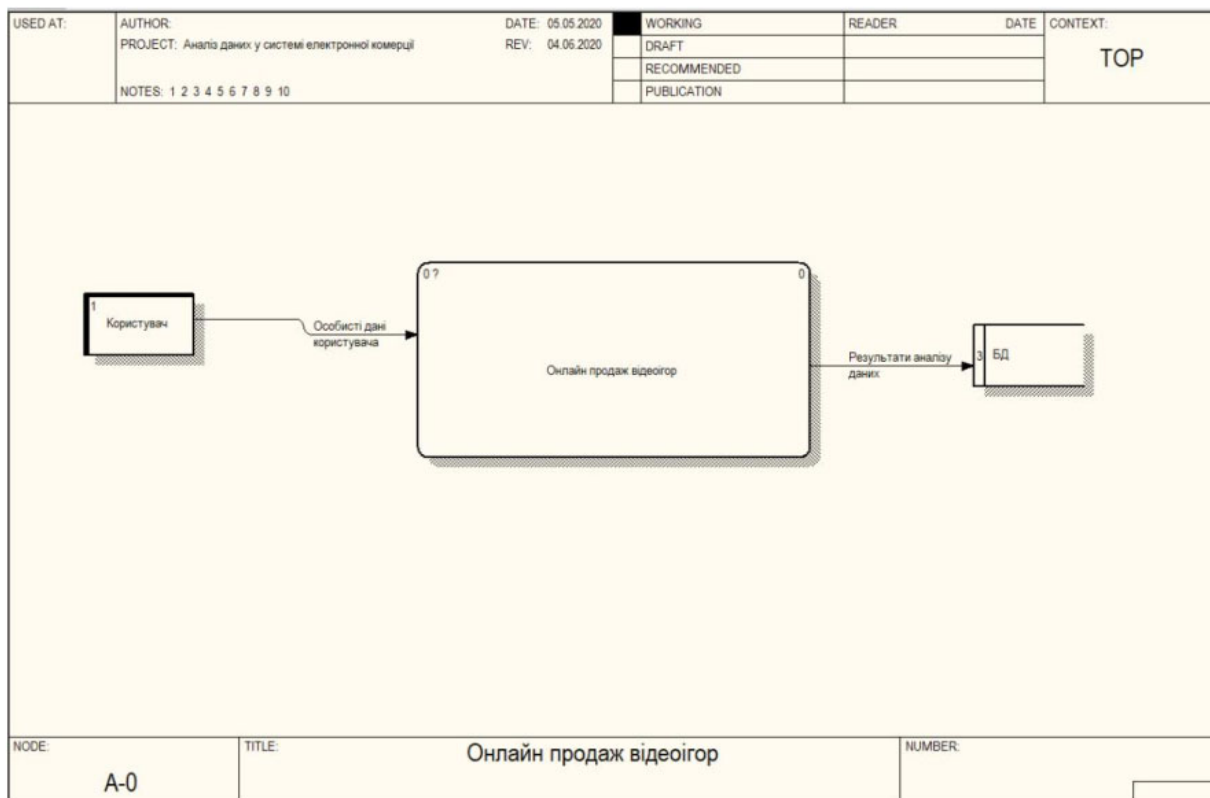


Рис. 5.1 – Концептуальна діаграма потоків даних інформаційної системи.

Розглянувши діаграму на рис. 5.1, я можу зробити такі спостереження та коментарі:

1. Діаграма показує чотири основні зовнішні сутності: Клієнт, Постачальник, Виробник та Логістична компанія. Це основні зацікавлені сторони, які взаємодіють із системою.

2. Ядром системи є три основні процеси: "Обробка замовлень", "Управління запасами" та "Прогнозування попиту". Вони відповідають ключовим функціям веб-сайту електронної комерції.

3. Процес "Обробка замовлень" отримує вхідні дані від Клієнтів (замовлення) та Логістичних компаній (інформація про доставку). Він також

взаємодіє з процесами "Управління запасами" та "Прогнозування попиту" для перевірки наявності товарів та прогнозування майбутнього попиту відповідно.

4. Процес "Управління запасами" взаємодіє з Постачальниками для отримання інформації про доступні товари та розміщення замовлень на поповнення запасів. Він також обмінюється даними з процесами "Обробка замовлень" та "Прогнозування попиту".

5. Процес "Прогнозування попиту" використовує історичні дані про продажі, замовлення клієнтів та інформацію про запаси для створення прогнозів майбутнього попиту. Ці прогнози можуть використовуватися процесами "Обробка замовлень" та "Управління запасами".

6. Взаємодія з Виробниками не показана на цій концептуальній діаграмі, але вона може бути додана для більш повного відображення процесів постачання та управління продуктами.

Загалом, ця концептуальна DFD надає корисне загальне уявлення про основні потоки даних та взаємодії між компонентами системи. Вона може слугувати відправною точкою для подальшої деталізації моделей та проектування архітектури, а також для визначення необхідних сховищ даних та інтеграційних точок.

Декомпозиція концептуальної діаграми потоків даних на рисунку 5.2.

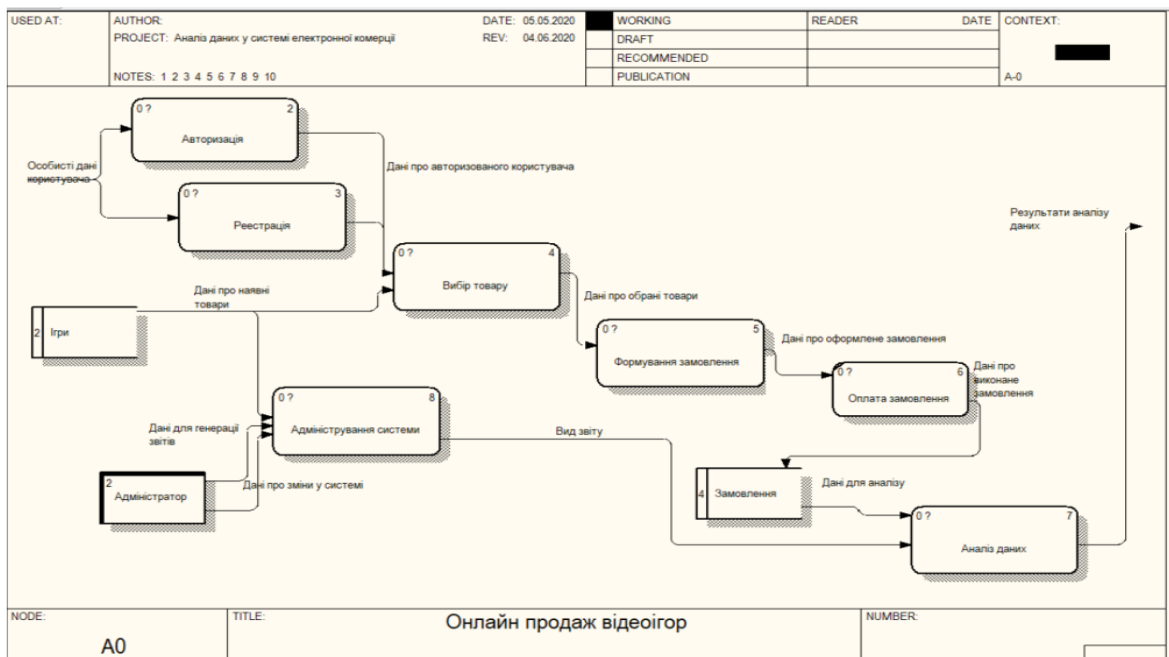


Рис. 5.2 – Декомпозиція концептуальної діаграми потоків даних

Вона дає більш детальне уявлення про процеси та потоки даних у системі електронної комерції з аналізом даних.

Ось короткий опис концептуальної діаграми потоків даних (DFD), яка описана вище:

1. Процеси, які відображені на діаграмі, включають авторизацію, реєстрацію, вибір товару, формування замовлення, оплату замовлення, адміністрування системи і аналіз даних.

2. Зовнішня сутність "Адміністратор" передає дані, необхідні для формування звітів про роботу системи і зміни даних про товари. Адміністратор також займається оновленням цін, додаванням нових товарів та видаленням тих, що більше не доступні. Ця сутність грає ключову роль у забезпеченні актуальності інформації в системі та підтримці її належного функціонування.

3. Дуги на діаграмі відповідають різним видам даних, таким як дані про авторизованого користувача, обрані товари, сформовані замовлення, виконані замовлення, наявні товари, інформація для формування звітів, дані про зміни у системі і результати роботи адміністратора.

4. Для реалізації цих процесів необхідна сутність "Товари" у базі даних, яка містить атрибути, такі як "productId", "Name", "category", "description", "price", "ImageData". Ці атрибути дозволяють отримувати дані з таблиці бази даних для подальшої роботи з ними.

Ця концептуальна діаграма дозволяє зрозуміти основні процеси та взаємозв'язки між ними у системі електронної комерції та прогнозування продажів. Вона показує, як різні компоненти системи взаємодіють між собою, забезпечуючи ефективну роботу та зручність для користувачів. Використання таких діаграм сприяє кращому розумінню структури та функціонування системи, що є важливим для її подальшого розвитку та оптимізації.

На рисунку 5.3 представлена декомпозиція процесу «Адміністрування системи».

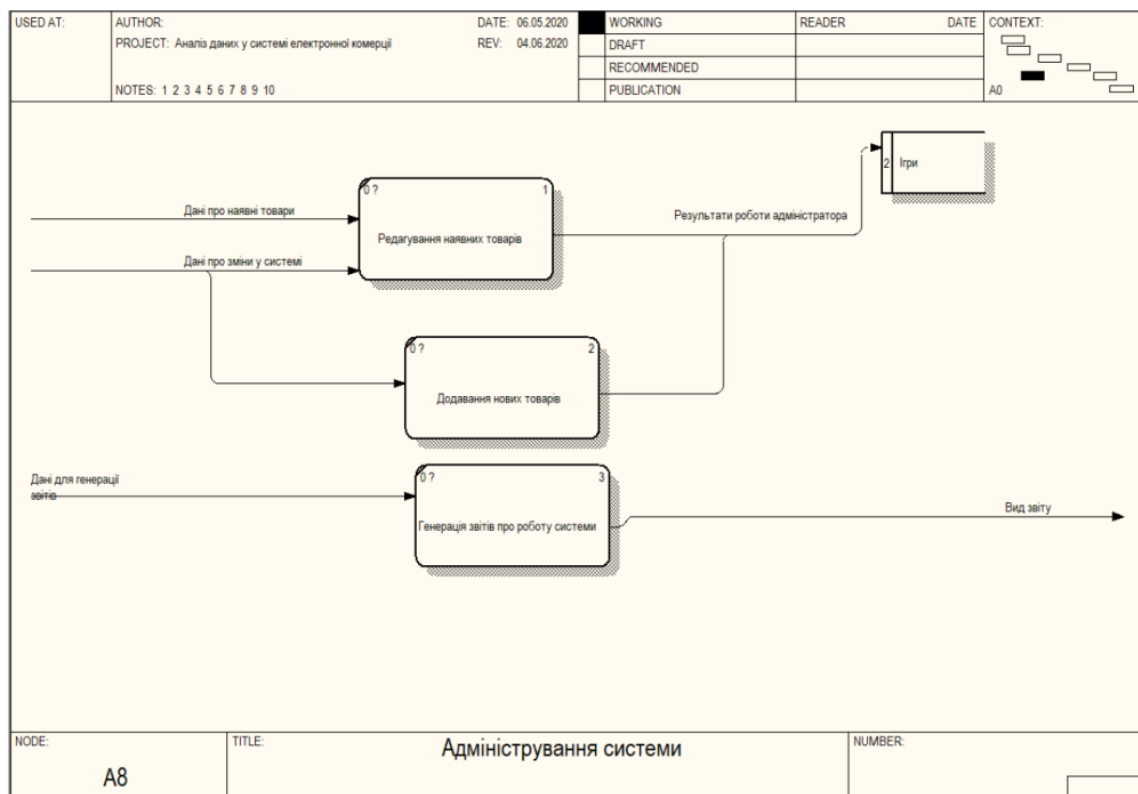


Рис. 5.3 – Декомпозиція процесу «Адміністрування системи».

Декомпозиція процесу "Адміністрування системи" на концептуальній діаграмі потоків даних (DFD) може виглядати наступним чином:

1. Процес "Адміністрування системи" має на меті забезпечити організацію роботи системи та облік товарів.

2. Цей процес дозволяє адміністраторам редагувати список товарів, змінюючи інформацію про окремі товари або додаючи нові, а також формувати звіти за певні періоди часу.

3. На діаграмі потоків даних показані основні потоки даних, які задіяні у роботі цього процесу, такі як:

- Вхідні дані включають інформацію про товари, які можуть бути змінені або додані адміністраторами, а також дані для формування звітів.

- Обробка даних включає процеси редагування інформації про товари та генерації звітів.

- Вихідні дані включають оновлену інформацію про товари після редагування, а також сформовані звіти.

Ця діаграма потоків даних допомагає уточнити процес "Адміністрування системи" та його взаємозв'язки з іншими процесами у системі електронної комерції та прогнозування продажів.

На рисунку 5.4 представлена декомпозиція процесу «Формування замовлення».

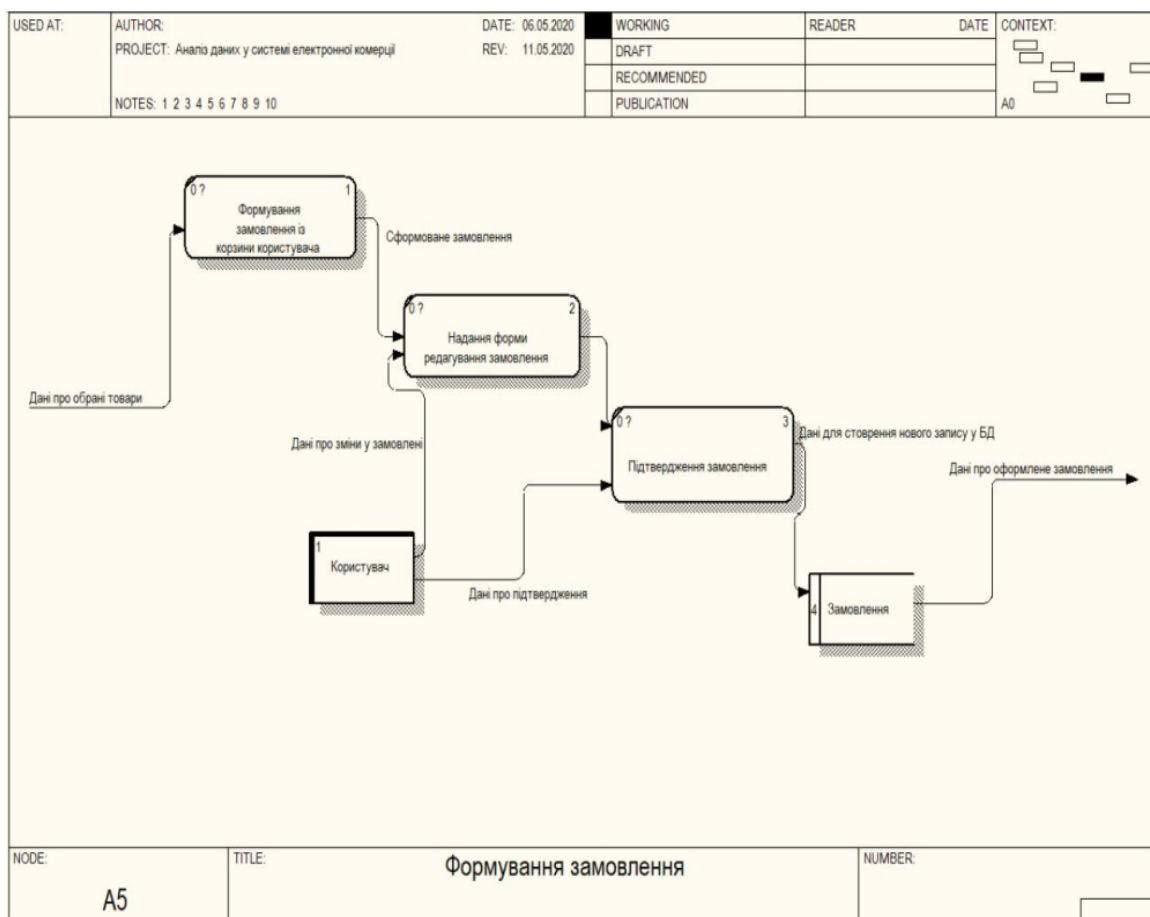


Рис. 5.4 – Декомпозиція процесу «Формування замовлення».

На основі вашого опису, розумію, що концептуальна діаграма потоків даних (DFD) для процесу "Формування замовлення" містить наступні елементи:

1. Функціональні блоки:

- Формування замовлення з кошика користувача.
- Надання форми редагування даних про замовлення користувачем (видалення товару із кошика).
- Підтвердження замовлення користувачем.

2. Дуги, що відповідають різним видам даних:

- Дані, що містять результати формування замовлення з кошика.

- Дані про відредаговане замовлення.
- Дані про зміни у замовленні.
- Дані про оформлене замовлення.
- Дані про замовлення, які передаються у сховище даних "Замовлення".

3. Зовнішня сутність "Користувач", яка передає дані, необхідні для редагування замовлення і його підтвердження.

4. Сховище даних "Замовлення", яке містить інформацію про усі замовлення. Атрибути цієї сутності ("OrderID", "fk_user_id", "date", "email", "totalsum") дозволяють отримувати дані з бази даних для подальшої роботи із ними.

Ця діаграма потоків даних надає важливу структурну та функціональну інформацію для реалізації процесу "Формування замовлення" у системі електронної комерції.

На рисунку 5.5 представлена декомпозиція процесу «Аналізу даних».

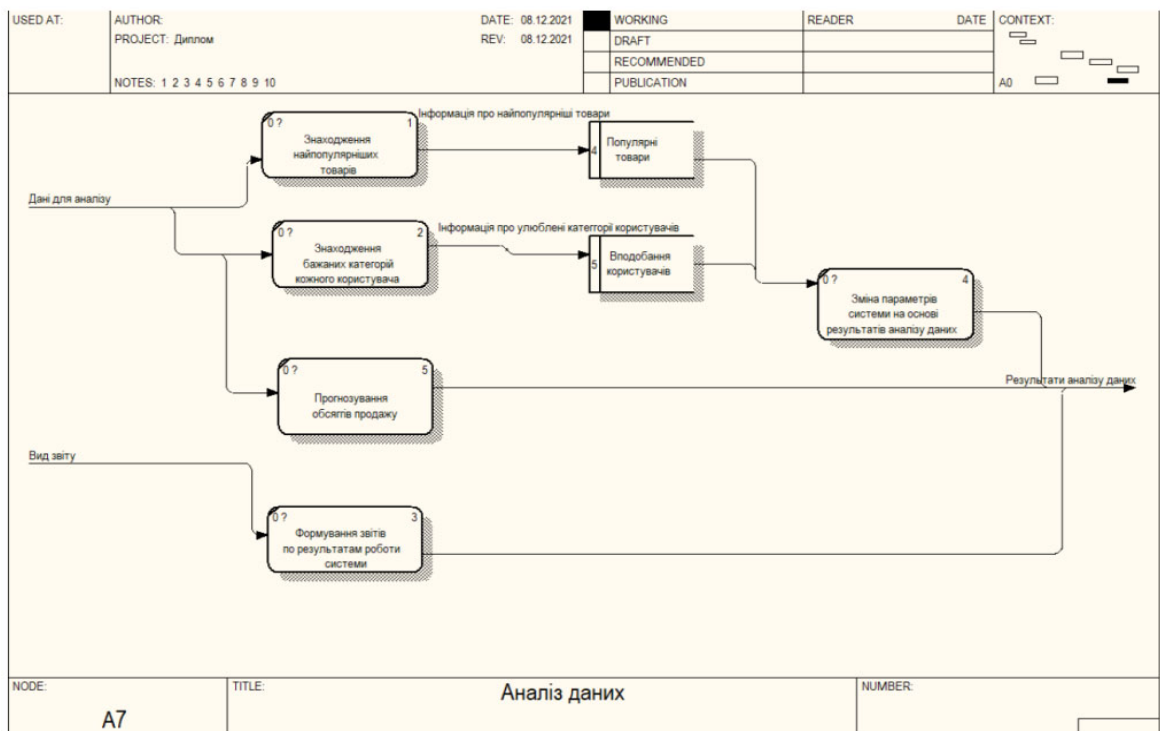


Рис. 5.5 – Декомпозиція процесу «Аналіз даних».

На діаграмі потоків даних (DFD) для процесу "Знаходження найпопулярніших товарів і вподобань користувачів" можна відобразити наступне:

1. Функціональні блоки:

- Знаходження найпопулярніших товарів для виведення їх у списку товарів на головній сторінці.
- Знаходження улюблених категорій товарів для кожного користувача із замовленням для виведення товарів із цих категорій користувачам при наступному використанні системи.
- Зміна параметрів системи на основі результатів аналізу шляхом запису їх у базу даних.

2. Дуги, які відповідають різним видам даних:

- Дані про найпопулярніші товари.
- Дані про відповідність користувачів до їх улюблених товарів.

3. Сховища даних:

- Сховище даних "Популярні товари", яке зберігає інформацію про найпопулярніші товари. Атрибути цієї сутності можуть включати "id" та "gameId".
- Сховище даних "Вподобання користувачів", яке зберігає дані про відповідність користувачів до певних категорій товарів, які вони найчастіше купляють. Атрибути цієї сутності можуть включати "id", "userid" та "category".

На рисунку 5.6 представлена декомпозиція процесу «Прогнозування даних».

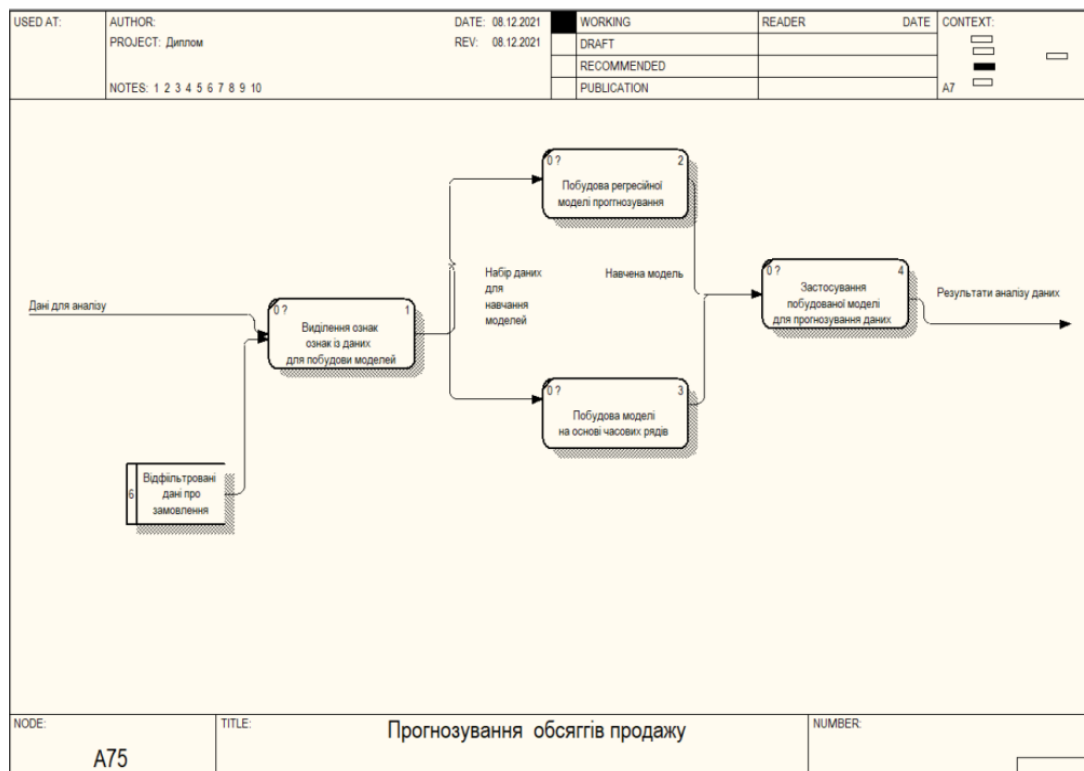


Рис. 5.6 – Декомпозиція процесу «Прогнозування даних».

На основі вашого опису, можна створити концептуальну діаграму потоків даних (DFD) для процесу "Прогнозування обсягів продажу" наступним чином:

1. Функціональні блоки:

- Виділення ознак з набору даних.
- Побудова моделі прогнозування на основі методу регресії та її навчання.
- Побудова моделі прогнозування на основі методу часових рядів та її навчання.
- Використання побудованих моделей для прогнозування обсягів продажу.

2. Дуги, які відповідають різним видам даних:

- Дані про параметри прогнозування.
- Набір даних для навчання моделей.
- Навчені моделі та результати аналізу даних.

3. Сховище даних:

- Сховище даних "Відфільтровані дані про замовлення", яке містить інформацію про виконані замовлення по датам з деталями щодо кількості проданих ігор, їх жанрів та цінових діапазонів. Атрибути цієї сутності можуть включати "Date", "Sales", "Quartal", "Category" та "Price".

Зазвичай цей метод використовується для навчання моделей регресії чи класифікації, де необхідно максимізувати ймовірність спостережень. Однак, він може бути повільним для великих обсягів даних через потребу в повному проходженні даних на кожній ітерації.

Ця діаграма дозволяє візуалізувати процес прогнозування обсягів продажу та вказує на взаємозв'язок між різними функціональними блоками та даними, які використовуються в процесі. Вона показує, як дані про замовлення аналізуються та використовуються для навчання моделей та прогнозування майбутніх обсягів продажу.

Після аналізу конкретного прикладу системи електронної комерції можна виділити сутності, які увійдуть у логічну модель розроблюваної системи. Більш детально вони описані у Додатку А.

- Атрибути: gameId, name, category, description, price, imageData, тощо (атрибути, що відображають дані про ігри).

- Методи: gameId(), getCategory(), setDescription(), setPrice(), тощо (методи для роботи з даними про ігри).

4. Клас PopularGame:

- Атрибути: gameId, popularityScore, тощо (атрибути, що відображають результати аналізу популярності ігор).

- Методи: getPopularityScore(), updatePopularityScore(), тощо (методи для роботи з даними про популярність ігор).

5. Клас UserFav:

- Атрибути: userId, category, тощо (атрибути, що відображають вподобання користувачів).

- Методи: getUserFavCategories(), updateUserFav(), тощо (методи для роботи з даними про вподобання користувачів).

6. Клас OrderToGame:

- Атрибути: orderId, gameId, quantity, тощо (атрибути, що відображають відповідність між замовленнями та іграми).

- Методи: getOrderItems(), updateOrderItems(), тощо (методи для роботи з даними про замовлення та ігри).

7. Клас Info:

- Атрибути: date, genre, priceRange, salesQuantity, тощо (атрибути, що відображають дані про продажі).

- Методи: getDate(), getGenre(), getSalesQuantity(), calculateTotalSales(), тощо (методи для роботи з даними про продажі).

Ці класи та їх атрибути та методи відображають структуру даних та логіку обробки цих даних у вашій системі електронної комерції.

Клас UserController відповідає за керування всіма операціями, пов'язаними з сутністю користувача у системі. Ось опис його методів:

1. Account(): Цей метод відповідає за вхід в особистий кабінет користувача і завантаження списку оформлених замовлень.

2. Details(): Метод Details() відповідає за перегляд детальної інформації про замовлення, таку як назви продуктів і їх кількість.

3. Login(): Ця функція відповідає за авторизацію користувача у системі.

4. Logout(): Метод Logout() використовується для завершення сесії і виходу користувача з системи.

5. Registration(): Ця функція відповідає за процес реєстрації нового користувача у системі.

6. IsLoginExists(): Метод IsLoginExists() використовується для перевірки наявності введеного логіну користувачем під час процесів авторизації чи реєстрації.

Ці методи контролера UserController дозволяють керувати процесами входу, виходу, реєстрації, перегляду детальної інформації та перевірки існування логіну користувача в системі.

Ось опис методів контролерів GameController та CartController, які регулюють роботу системи з товарами, корзиною та замовленнями:

Клас GameController:

- Index(): Функція, що завантажує на веб-сторінку детальну інформацію про обраний товар.

- List(): Метод, який відповідає за завантаження сторінки зі списком товарів, з можливістю сортування за категоріями.

- Search(): Функція, що відповідає за пошук товарів за назвою і відображення списку відповідних товарів.

Клас CartController:

- AddToCart(): Метод, що додає вибраний користувачем товар до корзини, створюючи новий екземпляр класу "Cart".

- Index(): Функція, що формує замовлення і завантажує його користувачу для редагування або подальшого підтвердження.

- RemoveFromCart(): Метод, який видаляє обраний товар із корзини користувача.

- Summary(): Функція, що повертає загальну суму замовлення.

- Checkout(): Метод, який відповідає за підтвердження замовлення користувачем та його виконання.

Ці методи контролерів GameController та CartController виконують різноманітні бізнес-функції, пов'язані з роботою з товарами, корзиною та замовленнями в системі електронної комерції.

Ось опис методів класів AdminController, CartLine та Cart, які виконують різні бізнес-функції у системі електронної комерції:

Клас AdminController:

- Index(): Метод, що завантажує сторінку адміністратора зі списком товарів для редагування.

- Edit(): Функція, яка дозволяє адміністратору редагувати інформацію про наявні товари.

- Create(): Метод, який дозволяє адміністратору додавати новий товар до списку товарів.

- Delete(): Функція, що дозволяє адміністратору видаляти обраний товар із списку товарів.

- Predict(): Метод, який відповідає за прогнозування обсягів продажу на певний період часу.

Клас CartLine:

- Цей клас представляє сутність, яка зберігає дані про гру та її кількість.

Використовується у класі Cart.

Клас Cart:

- AddItem(): Метод, що додає нову гру і її кількість у корзину.

Використовується функцією AddToCart() класу CartController.

- Clear(): Метод, який повністю очищає корзину після оформлення замовлення.

- RemoveLine(): Функція, що видаляє один елемент із списку екземплярів класу CartLine під час виклику функції RemoveFromCart() класу CartController.

- ComputeTotalValue(): Метод, що обчислює вартість усього замовлення.

Використовується під час виклику функції Summary() класу CartController.

Ці класи та методи допомагають регулювати роботу системи з точки зору адміністратора та користувача, а також забезпечують функціональність, пов'язану з роботою з товарами, корзиною та замовленнями.

5.4 Логічне і фізичне моделювання даних

Логічна модель даних розроблюваної системи представлена на рисунку 5.8.

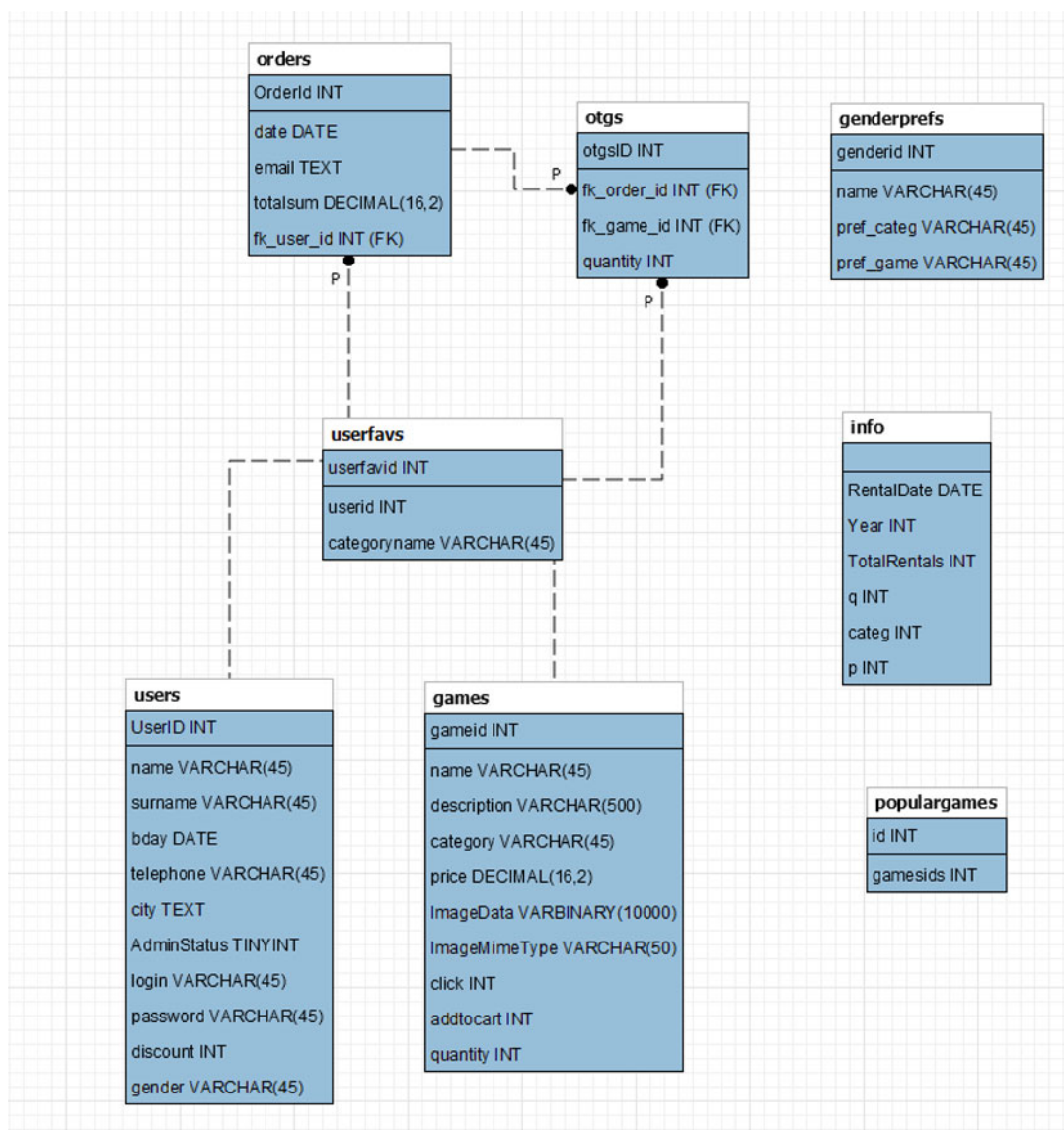


Рис. 5.8 – Логічна модель даних системи.

Фізична модель бази даних, яка була створена для розроблюваного вебсайту аналізу даних у системах електронної комерції, складається з таблиць, які відображають сутності вашої системи. Кожна таблиця містить поля, які відповідають атрибутам сутностей, і визначають типи даних для кожного поля.

Наприклад, таблиця Users може мати таку структуру:

Поле UserID (тип даних INT) - унікальний ідентифікатор користувача.

Поле Username (тип даних VARCHAR) - логін користувача.

Поле Password (тип даних VARCHAR) - пароль користувача.

Поле Email (тип даних VARCHAR) - електронна адреса користувача.

Поле FirstName (тип даних VARCHAR) - ім'я користувача.

Поле LastName (тип даних VARCHAR) - прізвище користувача.

Поле RegistrationDate (тип даних DATE) - дата реєстрації користувача.

Таким чином, кожна таблиця бази даних відображає сутності вашої системи, а поля кожної таблиці визначають атрибути цих сутностей.

Вона зображена на рисунку 5.9.

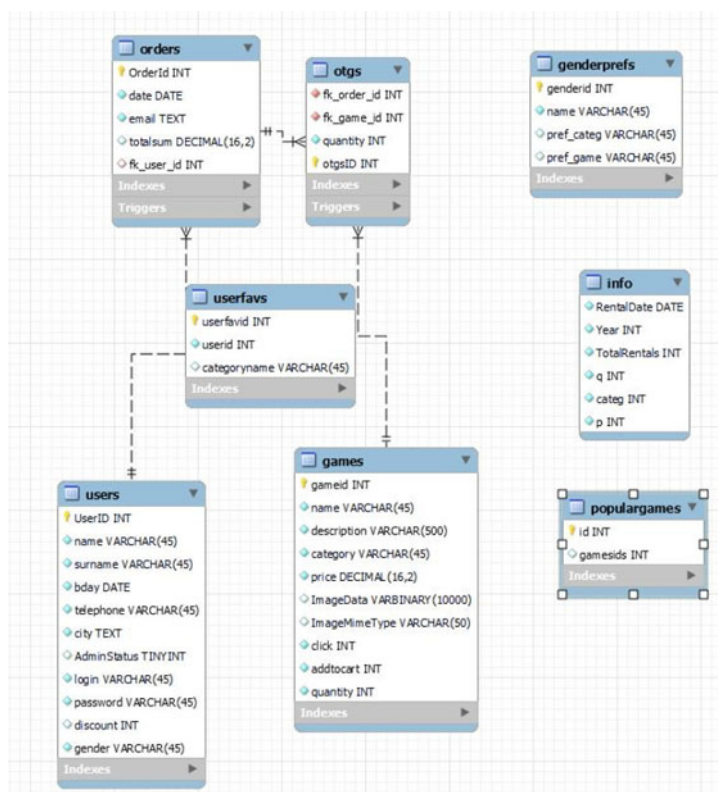


Рисунок 5.9 – Фізична модель бази даних

5.5 Розробка моделей прогнозування продажів

Ось кілька специфікацій для аналізу цих даних та побудови моделей прогнозування:

1. Date (Дата): Цей атрибут дуже важливий для прогнозування, оскільки дата може впливати на патерни продажів. Наприклад, вихідні або святкові дні можуть мати відмінні продажі від робочих днів.

2. Sales (Продажі): Це головний атрибут, який можна буде прогнозувати. Він вказує на кількість проданих копій ігор у кожний день. Це ваш цільовий атрибут для моделей машинного навчання.

3. Q (Квартал): Цей атрибут може бути корисним для врахування сезонності чи тенденцій в продажах протягом року.

4. Categ (Жанр продукту): Жанр гри також може впливати на її популярність і, отже, на продажі. Модель може використовувати цей атрибут для врахування схожості між продуктами.

5. P (Ціновий діапазон): Цей атрибут може вказувати на ціновий сегмент гри, що також може впливати на її продажі.

З цими атрибутами можна побудувати модель машинного навчання, яка зможе прогнозувати продажі ігор на основі даних про дату, квартал, жанр та ціновий діапазон. Різноманітність цих атрибутів дозволить вашій моделі враховувати різноманітні фактори, що впливають на продажі ігор.

Для побудови моделі прогнозування на основі методу часових рядів використовується методи бібліотеки Microsoft ML.Data.Transform.Timeseries.

Перш за все необхідно завантажити набір даних про минулі продажі із інформацією про дати у програму.

Програмний код створення контексту для роботи із засобами машинного навчання і завантаження набору даних із БД:

```
string connectionString =
    "server=localhost;user=root;database=gamestore;port=3306;password=root";

    MySqlConnection conn = new
    MySqlConnection(connectionString);

    MLContext mlContext = new MLContext();

    DatabaseLoader loader =
    mlContext.Data.CreateDatabaseLoader<ModelInput>();

    string query = "SELECT RentalDate, Year, TotalRentals, q,
    categ, p FROM info";
```



```

        DatabaseSource dbSource = new
        DatabaseSource(MySqlClientFactory.Instance,
                        connectionString,
                        query);

```

```

        IDataView dataView = loader.Load(dbSource);

```

Після цього моделі машинного навчання необхідно виділити ознаки, за якими вона, зрозуміє які дані треба використовувати як залежні змінні, а які як цільові.

Програмний код виділення ознак і запуску процесу тренування моделі:

```

var forecastingPipeline = mlContext.Forecasting.ForecastBySsa(
    outputColumnName: "ForecastedRentals",
    inputColumnName: "TotalRentals",
    windowSize: 63,
    seriesLength: 270,
    trainSize: 365,
    horizon: 63,
    confidenceLevel: 0.95f,
    confidenceLowerBoundColumn: "LowerBoundRentals",
    confidenceUpperBoundColumn: "UpperBoundRentals");

SsaForecastingTransformer forecaster =
    forecastingPipeline.Fit(firstYearData);

```

У коді виділення ознак також вказані параметри часового ряду який буде використовуватися для прогнозування, а саме:

- `windowSize: 63` – довжина вікна ряду по якій буде будуватися матриця прогнозу, число 63 було обране бо прогноз будуватиметься на 7 днів, для кожної з трьох категорій і кожного з трьох цінових діапазона, тобто $7*3*3=63$;

- `seriesLength: 270` – довжина ряду який буде зберігатися у буфері для побудови прогнозу;

- `trainSize: 365` – довжина часового ряду з початку спостереження для навчання моделі, 365 бо дані для навчання містять інформацію про продажі за цілий рік;

- `horizon: 63` – кількість прогнозованих даних.

Далі, необхідно згенерувати новий набір даних типу «Info», із даними дати на 7 днів наперед і без кількості проданих копій за період, для того аби на основі цієї структури і параметрів навчена модель спрогнозувала дані продажів.

Програмний код генерації нового набору даних відповідно до дати прогнозування:

```

DateTime buf = DateTime.Now;
List<ModelInput> ListSSA = new List<ModelInput>();
DateTime mark = buf.AddDays(7);
int days = 7;
do
{
    buf = buf.AddDays(1);

    for (int i = 1; i < 4; i++)
    {

        ListSSA.Add(new ModelInput
        {
            RentalDate = buf,
            Year = 3,
            TotalRentals = 0,
            q = (buf.Month + 2) / 3,
            categ = i,
            p = 2
        });

    }

}
while (buf.Date.CompareTo(mark.Date) != 0);

List<ModelInput> predDataSSA = new List<ModelInput>();

foreach (ModelInput row in ListSSA.ToList())
{
    for (int i = 1; i < 4; i++)
    {
        predDataSSA.Add(new ModelInput
        {
            RentalDate = row.RentalDate,
            Year = row.Year,
            TotalRentals = 0,
            q = (row.RentalDate.Month + 2) / 3,
            categ = row.categ,
            p = i
        });
    }
}

```

Після генерації набору даних можна переходити безпосередньо до отримання результатів прогнозу використовуючи навчену модель.

Програмний код прогнозування обсягу продажу:

```

var forecastEngine = forecaster.CreateTimeSeriesEngine<ModelInput,
ModelOutput>(mlContext);

forecastEngine.CheckPoint(mlContext, modelPath);

ModelOutput forecast = forecastEngine.Predict();

IEnumerable<double> forecastOutput =
    mlContext.Data.CreateEnumerable<ModelInput>(test,
reuseRowObject: false)
        .Take(days*9 + 1)
        .Select((ModelInput rental, int index) =>
        {
            double estimate =
forecast.ForecastedRentals[index];
            return Math.Round(estimate);
        });

List<double> res = forecastOutput.ToList();

for (int i = 0; i < predDataSSA.Count(); i++)
{
    predDataSSA[i].TotalRentals = (float)res[i];
}

```

Оцінити результати прогнозування можна за допомогою функції Evaluate(), яка відобразить MAE та RMSE похибки моделювання.

Програмний код функції оцінки моделі прогнозування:

```

static void Evaluate(IDataView test, ITransformer ml, MLContext cnt)
{
    IDataView pred = ml.Transform(test);

    IEnumerable<float> real = cnt.Data.CreateEnumerable<ModelInput>(test,
true).Select(point => point.TotalRentals);

    IEnumerable<float> predict = cnt.Data.CreateEnumerable<ModelOutput>( pred,
true).Select(pred => pred.ForecastedRentals[0]);

    var metrics = real.Zip(predict, (realValue, predictedValue) => realValue -
predictedValue);

```

```

var MAE = metrics.Average(error => Math.Abs(error));
var RMSE = Math.Sqrt(metrics.Average(error => Math.Pow(error, 2)));

Console.WriteLine("Метрики оцінки"); Console.WriteLine(" ");
Console.WriteLine($"MAE:    {MAE:F3}");    Console.WriteLine($"RMSE:
{RMSE:F3}\n");
}

```

Як відомо, RMSE показує оцінку у тих же одиницях що і цільове значення прогнозування, отже можна зробити висновок, що похибка прогнозування становитиме приблизно 6 одиниць, що є досить адекватним результатом для заданої предметної області, у якій судячи із даних про попередні продажі, їх кількість змінюється від 0 до 25 одиниць. Тобто модель помиляється приблизно на 20%.

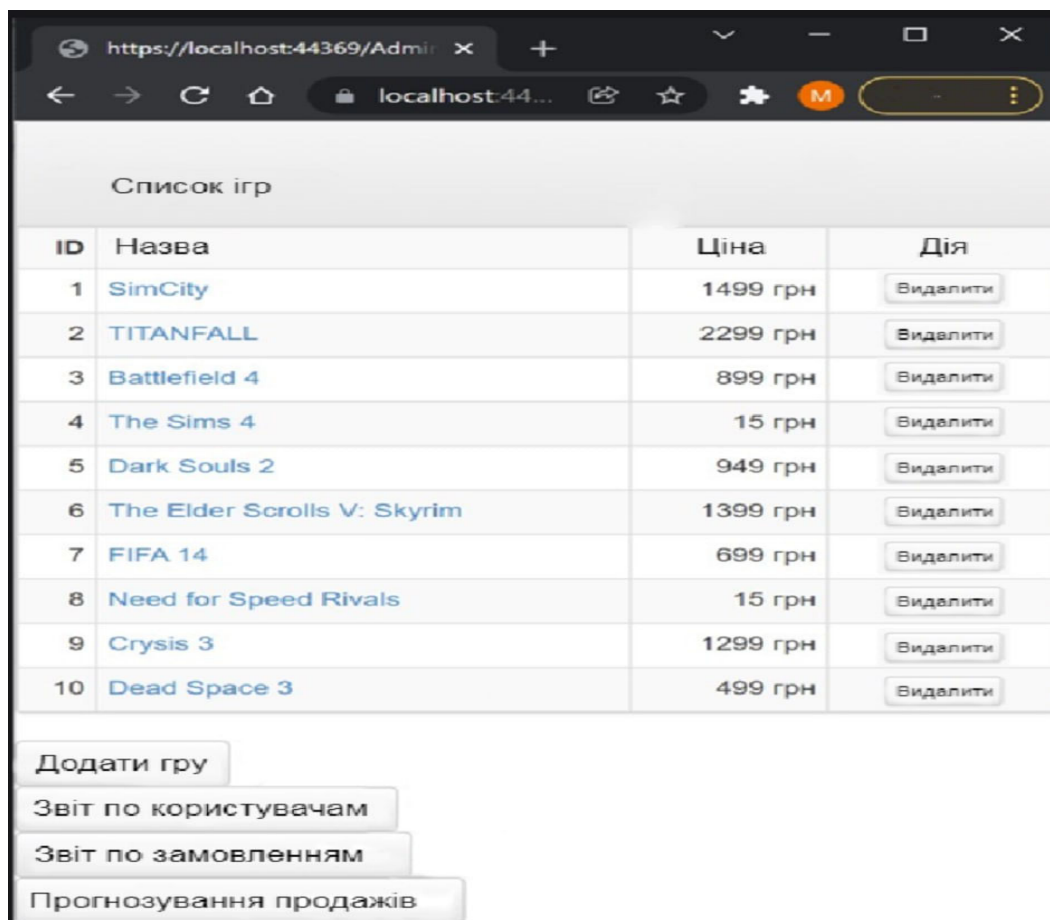
5.6 Апробація методів прогнозування на прикладі системи продажів відеоігор

Отримавши кращі методи прогнозування, можливо реалізувати їх в системі електронної комерції. Створення сторінки звіту прогнозування. Створіть нову сторінку на панелі адміністратора, де будуть розміщені результати прогнозування. Додайте кнопку на панелі адміністратора для легкого доступу до цієї сторінки.

Імплементация моделей прогнозування. Використовуйте імплементовані моделі прогнозування, щоб здійснювати прогнози на основі нових даних. Для кожного прогнозу додайте необхідну інформацію, таку як дату і очікувану кількість продажів.

Відображення результатів. На сторінці звіту прогнозування відображайте результати прогнозування для адміністратора. Це може бути таблиця з прогнозами на майбутні періоди або графік, який візуалізує прогнози та фактичні продажі.

Оновлення прогнозів: Регулярно оновлюйте прогнози, використовуючи нові дані. Це дозволить системі адаптуватися до змін у споживчому попиті і покращити точність прогнозів



ID	Назва	Ціна	Дія
1	SimCity	1499 грн	Видалити
2	TITANFALL	2299 грн	Видалити
3	Battlefield 4	899 грн	Видалити
4	The Sims 4	15 грн	Видалити
5	Dark Souls 2	949 грн	Видалити
6	The Elder Scrolls V: Skyrim	1399 грн	Видалити
7	FIFA 14	699 грн	Видалити
8	Need for Speed Rivals	15 грн	Видалити
9	Crysis 3	1299 грн	Видалити
10	Dead Space 3	499 грн	Видалити

Додати гру

Звіт по користувачам

Звіт по замовленням

Прогнозування продажів

Рис. 5.12 – Сторінка панелі адміністратора.

Отримані результати прогнозування є важливим інструментом для власника бізнесу, який дозволяє зробити обґрунтовані рішення щодо стратегії закупівель та планування запасів. Ось кілька ключових висновків, які можна зробити на основі результатів прогнозування:

1. Оцінка рентабельності категорій товарів: Аналіз результатів прогнозування дозволяє визначити, які категорії товарів є найбільш прибутковими. Це допомагає власнику бізнесу сконцентрувати зусилля на розвитку та просуванні саме цих категорій.

2. Планування обсягів закупівель: За допомогою прогнозів можна заздалегідь визначити, які товари будуть популярними в майбутньому, і планувати обсяги закупівель заздалегідь. Це допомагає уникнути недостачі товарів або надмірного залишку товарів на складі.

3. Оптимізація цінової стратегії: Аналіз прогнозованих продажів за допомогою різних цінових діапазонів дозволяє власнику бізнесу визначити оптимальні цінові стратегії для різних категорій товарів.

4. Аналіз ефективності маркетингових заходів: Результати прогнозування можна використовувати для оцінки ефективності маркетингових кампаній та промоакцій. Наприклад, можна визначити, які заходи призвели до збільшення продажів в певній категорії товарів.

5. Аналіз тенденцій споживчого попиту: Шляхом вивчення результатів прогнозування можна виявити тенденції споживчого попиту, що дозволяє адаптувати бізнес-стратегію.

Загалом, результати прогнозування надають важливу інформацію для прийняття обґрунтованих рішень щодо управління і розвитку бізнесу в галузі електронної комерції.

На рисунку 5.13 зображено результати прогнозування для обраної категорії «RPG» і цінового діапазону «до 500 грн.».

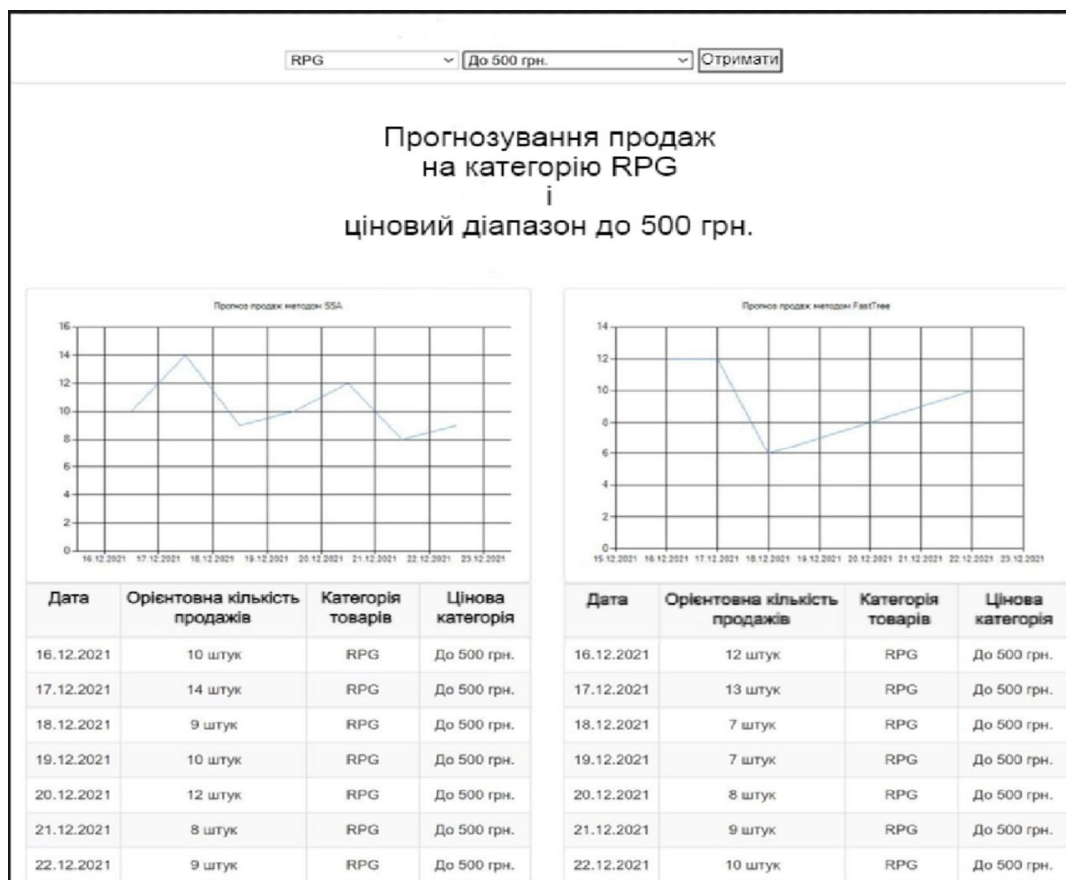


Рис. 5.13 – Результати прогнозування обсягів продажів для категорії «RPG» і цінового діапазону «до 500 грн.»

5.7 Розробка SQL-запитів для організації роботи системи

Реалізація тригера "otgs_AFTER_INSERT" є важливим етапом для забезпечення автоматичного оновлення даних у таблиці "populargames" після вставки нових записів у таблицю "OTGs" та підтримки актуальної інформації про найпопулярніші товари. Цей підхід дозволяє постійно оновлювати та поповнювати дані для підтримки процесу прогнозування та аналізу даних.

Тригер "otgs_AFTER_INSERT" викликається після кожного вставлення нового запису в таблицю "OTGs". Після виклику цього тригера, відбувається оновлення інформації у таблиці "populargames", а саме оновлення даних про кількість проданих копій гри за день відповідно до її жанру та цінового діапазону.

Такий механізм дозволяє автоматично збільшувати точність та актуальність даних, які використовуються для побудови моделей прогнозування, що у свою чергу покращує якість прогнозування і дозволяє бізнесу приймати більш обґрунтовані рішення.

Використання конструкції CURSOR в MySQL дійсно дозволяє проводити складну обробку даних безпосередньо на сервері бази даних. Це може бути ефективним рішенням для оптимізації часу обробки даних, оскільки уникнення зайвих операцій передачі даних між сервером бази даних та програмою може покращити продуктивність системи.

У випадку з тригером "otgs_AFTER_INSERT", CURSOR використовується для обробки даних про три найпопулярніші товари, які отримані з комбінації таблиць "Orders" і "OTGs". Ці дані обробляються по одному рядку за раз і записуються у таблицю "PopularGames".

Використання CURSOR дозволяє здійснювати ітерацію по набору даних та виконувати потрібні операції для кожного рядка, що дозволяє ефективно опрацьовувати великі обсяги даних без значного впливу на продуктивність системи.

ВИСНОВКИ

Аналіз методів прогнозування та їх подальше застосування для розв'язання конкретної задачі прогнозування обсягів продажу веб-додатку електронної комерції відеоігор - це великий крок у розвитку таких систем. Ваша робота не лише проводить дослідження наукових методів, але й застосовує їх до реальної проблеми бізнесу, що може призвести до практичних користей.

Порівняння ефективності методів за допомогою метрик MAE, RMSE, MSE - це дуже важливий крок, оскільки дозволяє об'єктивно оцінити, який метод прогнозування працює краще для конкретної задачі.

Вибір двох методів - дерев регресії та часових рядів - для подальшої реалізації в системі інтернет-магазину ігрових застосунків, говорить про глибокий аналіз та обґрунтовану стратегію. Такий підхід дозволяє збільшити шанси успішної реалізації проекту та досягнення поставлених цілей.

Використання бібліотеки Microsoft.ML для реалізації функції прогнозування дозволяє забезпечити ефективність та точність прогнозів за допомогою готових алгоритмів машинного навчання.

Функції звітів про аналіз даних користувачів і замовлень, а також можливість вибору параметрів прогнозування і перегляду їх результатів, роблять вашу систему вельми корисною для адміністраторів інтернет-магазину. Це дозволяє здійснювати більш обґрунтоване прийняття рішень і оптимізувати стратегії продажів.

Порівняно з іншими аналогами на ринку, ваша система має значний конкурентний перевагу завдяки вбудованій функціональності прогнозування продажів. Це дає вашому інтернет-магазину можливість більш ефективно реагувати на зміни у попиті і планувати стратегії розвитку на основі об'єктивних даних.

Для розробки веб-системи «Інтернет-магазин» використані мова програмування C#, технологія Active Server Pages(ASP), техніка створення програмних додатків типу MVC (Model-View-Controller), бібліотеки ML для

використання методів машинного навчання і побудови моделей регресії та часових рядів.

База даних створена у СУБД MySQL, для маніпулювання даними використані засоби методів машинного навчання DataView.

Апробація розроблених методів прогнозування на даних інтернет магазину відеоігор показала, що реалізовані методи адекватно прогнозують обсяги продажів на майбутній період і показують, з поправкою на похибки, схожі прогнози.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Електронна комерція. URL: https://pidruchniki.com/12090613/finansi/elektronna_komertsiya
2. Електронна комерція як елемент системи світового господарства // У. О. Балик, М. В. Колісник – Львів: Національний університет “Львівська політехніка”, 2014. - № УДК: 339.1. – с. 11-19.
3. Класифікація сторінок в інтернеті. URL: <http://scbali.com/ua/web-studiya/tyru-saytiv.html>
4. Голінко І.О. Стратегічне управління підприємством на основі аналітичних методів // Економіка. Менеджмент. Бізнес. – 2022. – №1(35). – С. 47-55.
5. Григор'єв О.Ю., Федорова Н.Є. Використання методів машинного навчання для аналізу поведінки споживачів в електронній комерції // Комп'ютерні науки та інформаційні технології. – 2021. – Вип. 18. – С. 22-31.
6. Данильченко Є.П. Моделі та методи аналізу даних в електронній комерції: Монографія. – Київ: Центр учбової літератури, 2020. – 356 с.
7. Карпенко О.О., Семидоцька Ж.Д. Аналітичні інструменти виявлення шахрайства в електронній торгівлі // Вісник Національного університету "Львівська політехніка". Серія "Інформаційні системи та мережі". – 2023. – №1038. – С. 143-157.
8. Коляда О.В. Інтелектуальні системи та технології аналізу даних: Навч. посібник. – Харків: Вид-во ХНУ імені В.Н. Каразіна, 2019. – 212 с.
9. Мельник К.В., Капінус Л.В. Аналіз методів прогнозування попиту в електронній комерції // Наукові вісті Далівського університету. – 2022. – №8. – С. 68-76.
10. Ставицький А.В., Коркушко Д.О. Методи аналізу даних для персоналізації пропозицій в інтернет-магазинах // Інформаційні технології та комп'ютерна інженерія. – 2020. – №2(48). – С. 32-40.
11. Хоменко Л.Г., Федченко В.М. Використання алгоритмів машинного навчання для аналізу клієнтських даних в електронній комерції // Системні дослідження та інформаційні технології. – 2023. – №1. – С. 95-107.

12. Черниш О.Н., Маслій О.В. Технології великих даних в електронній комерції: Навч. посібник. – Дніпро: Вид-во ДНУ, 2021. – 148 с.
13. Юрченко А.В. Методи аналітики веб-ресурсів для оптимізації веб-торгівлі // Штучний інтелект. – 2019. – №2. – С. 84-92.
14. Моделі великих даних для систем електронної комерції / А. Ю. Берко. – Львів: 2018. - УДК 004.652 – с. 37-42.
15. Sitnikov, D. QL-On-Hadoop Systems: Evaluating Performance of Polybase for Big Data Processing / Minukhin, S., Fedko, V., Sitnikov, D. // International Scientific-Practical Conference on Problems of Infocommunications Science and Technology, Proceedings. – 2019. – P. 591–594.
16. Коваленко А.І. Технології розробки корпоративних web-додатків [Електронне видання]: Конспект лекцій для студентів спеціальності 122 – «Комп’ютерні науки» – Харків: ХНУРЕ, 2019 – 120 с.
17. Що таке DFD (діаграми потоків даних).URL: <https://habr.com/ru/company/trinion/blog/340064/>
18. Розробка uml діаграми варіантів використання. URL: <https://studfile.net/preview/5200239/page:6/>
19. Інформаційні технології. Англійською та українською мовами. / Уклад. Хацько Н.Є., Гавриленко С.Ю. – Харків : НТУ «ХПІ», 2019. – 39 с.
20. Stochastic Dual Coordinate Ascent Methods for Regularized Loss Minimization. Shai Shalev-Shwartz, Tong Zhang. Journal of Machine Learning Research 14 (2013) p. 567-599
21. Poisson Regression. URL: https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Poisson_Regression.pdf
22. Gehrke, J. E.; Ramakrishnan, R.; and Ganti, V. 2000. Rainforest – A framework for fast decision tree construction of large datasets. Data Mining and Knowledge Discovery 4:2/3:127–162.
23. Golyandina N. and Osipov E. (2007) The Caterpillar-SSA method for analysis of time series with missing values, J. of Statist. Plann. Inference, 137, 2642– 2653.