МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

ТКАЧИК ОЛЕКСАНДР АНДРІЙОВИЧ

На правах рукопису

УДК 004.652

МЕТОДИ ТА ЗАСОБИ КЛАСТЕРИЗАЦІЇ РІЗНОТИПОВИХ ДАНИХ

122 – Комп’ютерні науки

Дисертація на здобуття наукового ступеня
доктора філософії

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

_____/О. А. Ткачик/

Науковий керівник

Бойко Наталія Іванівна

кандидат економічних наук, доцент

Львів – 2023

АНОТАЦІЯ

Ткачик О. А. Методи та засоби кластеризації різнотипових даних. – Кваліфікаційна наукова праця на правах рукопису. Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 122 “Комп’ютерні науки”. – Національний університет «Львівська політехніка», Львів, 2023.

Зміст анотації. Дисертаційна робота призначена для розробки методів та інструментів штучного інтелекту з метою створення користувацьких профілів на онлайн-платформі нерухомості. Ця ініціатива в перспективі дозволить оптимізувати взаємодію менеджерів системи з користувачами та підвищить задоволеність клієнтів завдяки наданню більш точних пропозицій.

У *першому* розділі проаналізовано алгоритми опрацювання різнотипових даних, таких як як K-середніх, DBSCAN, ієрархічна кластеризація, та нечітка кластеризація.

Виявлено ряд обмежень, які включають питання масштабування, інтерпретації, універсальності та адаптивності до змінюваних умов та структур даних. Також було виявлено, що не всі системи, що зараз є на ринку, використовують інформацію із неструктурованих чи напівструктурованих джерел даних для покращення своїх служб.

Це відкриває можливості для подальшого розширення та удосконалення. Базуючись на цих спостереженнях, можна стверджувати, що потрібно проводити подальші дослідження з метою розробки більш ефективних та гнучких методів кластеризації для різнотипових даних. Це не тільки може покращити здатність систем до створення більш точних та інформативних профілів користувачів, але й привести до відкриття нових можливостей у використанні цих даних для бізнес-аналітики, прогнозування та інших цілей.

У *другому* розділі проаналізовано методи кластеризації даних. Проведено формальне визначення профілів користувачів, що включає демографічні, поведінкові, психографічні, мотиваційні та знаннєво-інформаційні критерії. Кожен із цих критеріїв дозволяє сформувати із наявного датасету підмножини

даних, сформовані за певними ознаками. Підмножини даних, у свою чергу, складаються із різнотипових даних, таких як числові, текстові, ординальні, категоріальні та інших. Описано процес класифікації профілів користувачів на основі атрибутів користувачів. Проведено визначення рівня задоволеності користувача із допомогою метрик CSS, NPS, CES та CSI. Кожна метрика дозволяє визначити наскільки клієнти задоволені якістю послуг, спілкуванням з брокерами та агентами нерухомості, процесом транзакції, рівнем професіоналізму, цінами, умовами контракту та загальним досвідом взаємодії з компанією або агентством нерухомості і таким чином дозволяє зрозуміти потреби користувача більш точно.

У *третьому* розділі розроблено алгоритм підготовки даних. Проведено аналіз та порівняння методів обробки пропущених значень. Проведено аналіз методів виявлення та видалення дублікатів, проаналізовано та порівняно різні підходи до виявлення та усунення викидів. Проаналізовано роботу методів зменшення розмірності даних та виділення нових ознак. Застосовано статистичний метод перцентилів для розрахунку початкових центроїдів. Розроблено метод кластеризації різнотипових даних, який дозволяє працювати з потоковими даними на основі поділу на пакети.

В *четвертому* розділі дисертаційного дослідження була розроблена архітектура інформаційної системи для здійснення автоматизованого профілювання користувачів на основі різнотипових даних клієнтів.

Для зменшення вартості розгортання такої системи, при побудові використовується безсерверний підхід на основі стороннього сервісу Google Cloud Functions.

Впровадження запропонованої архітектури призвело до зниження кінцевої вартості системи в 4 рази, порівняно зі стандартною архітектурою, що використовує дроплети.

Основні результати дисертації опубліковано у 7 наукових працях, зокрема: **три** статті – у наукових фахових періодичних виданнях України; **одна** - у

закордонному фаховому періодичному виданні; **три** публікації матеріалів міжнародних науково-технічних конференцій.

Ключові слова: ієрархічна кластеризація, k-means, mini-batch k-means, percentile, навчання без вчителя, різнотипові дані, профілювання користувача, оцінка задоволеності, безсерверна архітектура, мікросервісна архітектура.

ABSTRACT

Tkachyk O. A. Methods and tools for clustering heterogeneous data. – Qualification scientific work on the rights of the manuscript. Dissertation for obtaining the degree of Doctor of Philosophy in specialty 122 "Computer Science". – National University "Lviv Polytechnic", Lviv, 2023. In the first chapter, algorithms for processing heterogeneous data, such as K-means, DBSCAN, hierarchical clustering, and fuzzy clustering, were analyzed. A number of challenges and limitations were identified, including issues of scaling, interpretation, universality, and adaptability to changing conditions and data structures. It was also discovered that not all systems currently on the market use information from unstructured or semi-structured data sources to improve their services.

Content of the abstract. The dissertation is dedicated to the development of methods and means of artificial intelligence for the operational creation of user profiles for an online real estate market platform. In the future, this will allow system managers to conduct more operations with users and will increase user satisfaction due to more accurate proposals.

This opens up opportunities for further expansion and improvement. Based on these observations, it can be argued that further research should be conducted to develop more efficient and flexible clustering methods for heterogeneous data. This will not only improve the ability of systems to create more accurate and informative user profiles, but also lead to the discovery of new opportunities in using this data for business analytics, forecasting, and other purposes.

In the second chapter, data clustering methods are analyzed. A formal definition of user profiles is provided, which includes demographic, behavioral, psychographic, motivational, and knowledge-information criteria. Each of these criteria allows the formation of data subsets from the available dataset, formed according to certain features. In turn, data subsets consist of heterogeneous data, such as numerical, textual, ordinal, categorical, and others. The process of classifying user profiles based on user attributes is described. The user satisfaction level is defined

using the metrics CSS, NPS, CES, and CSI. Each metric allows determining how satisfied customers are with the quality of services, communication with brokers and real estate agents, the transaction process, the level of professionalism, prices, contract terms, and the overall experience of interacting with the company or real estate agency. This, in turn, allows a more accurate understanding of user needs.

In the third chapter, an algorithm for data preparation was developed. An analysis and comparison of methods for handling missing values was conducted. The methods of detecting and removing duplicates were analyzed, as well as various approaches to detecting and eliminating outliers. The work of data dimensionality reduction methods and the extraction of new features was analyzed. The statistical percentile method was applied to calculate the initial centroids. A clustering method for heterogeneous data was developed, which allows working with streaming data based on batch division. In the fourth chapter of the dissertation research, an architecture of an information system for automated user profiling based on heterogeneous customer data was developed. To reduce the cost of deploying such a system, a serverless approach based on the third-party service Google Cloud Functions was used. The implementation of the proposed architecture led to a fourfold decrease in the final cost of the system compared to the standard architecture that uses droplets.

The main results of the dissertation are published in 7 scientific papers, including: three articles – in scientific professional periodical publications of Ukraine; one - in a foreign professional periodical publication; three publications of materials of international scientific and technical conferences.

Key words: hierarchical clustering, k-means, mini-batch k-means, percentile, unsupervised learning, heterogeneous data, user profiling, satisfaction assessment, serverless architecture, microservice architecture.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Статті у виданнях інших держав:

1. Mytnyk B., Tkachyk O., Shakhovska N., Fedushko S., Syerov Yu. Application of Artificial Intelligence for Fraudulent Banking. *Big Data Cogn. Comput.* 2023, 7(2), 93. https://doi.org/10.3390/bdcc7020093 (квартиль Q1 у НМБД Scopus)

Статті у фахових виданнях України:

1. Ткачик О. Застосування методів кластеризації даних для створення цільових груп користувачів на ринку нерухомості. Вісник Хмельницького національного університету, 2023. Том 2 (319), С. 300-307. https://www.doi.org/10.31891/2307-5732-2023-319-1-300-307
2. Бойко Н. І., Ткачик О. А. Алгоритми та методи кластеризації для різноманітних даних. Науковий вісник Ужгородського університету. Серія «Математика і інформатика» / редкол.: М. М. Маляр (гол. ред.) та інші. Ужгород: Видавництво УжНУ «Говерла», 2023. Т. 42, No 1. С. 131-150. https://www.doi.org/10.24144/2616-7700.2023.42(1).129-147
3. Бойко Н.І., Ткачик О.А. Оцінка методів кластеризації різнотипових даних. Journal Automation of technological and business –processes, 2023. Vol. 15(1), pp. 1-12. https://doi.org/10.15673/atbp.v15i1.2508

Матеріали конференцій:

1. Havano B., Kytsun H., Tkachyk O. Web-server cross-site request forgery protection, in: VII International Youth Conference "Perspectives of Science and Education", 2020 New York, USA, pp. 9–16. https://doi.org/10.29013/VII-Conf-USA-7-9-16
2. Boyko N., Tkachyk O. Frequency pattern growth algorithm (FP) for multimodal data extraction, in: Proceedings of the 3rd International Workshop on Intelligent Information Technologies & Systems of Information Security Khmelnytskyi, Ukraine, March 23–25, 2022, pp. 72-82.

3. Boyko N., Tkachyk O. Model for Finding Frequent Sets in FP-growth for Multimodal Data, in: Proceedings of The Fifth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2022), Zaporizhzhia, Ukraine, May 12, 2022, pp.126-143. https://doi.org/10.327 82/cmis/3137-11

ЗМІСТ

ВСТУП **12**

РОЗДІЛ 1. АНАЛІЗ ТЕХНОЛОГІЇ КЛАСТЕРИЗАЦІЇ РІЗНОТИПОВИХ ДАНИХ **16**

1.1. Оцінка впливу штучного інтелекту на нерухомість і персоналізацію споживачів 17

1.2. Визначення різнотипових даних 19

1.3. Визначення задачі профілювання 24

1.4. Аналіз існуючих рішень щодо збору та опрацювання даних 28

1.4.1. Методи збору даних 28

1.4.2. Сімейства алгоритмів кластеризації даних 31

1.4.3. Кластеризація на основі щільності 32

1.4.4. Ієрархічна кластеризація 33

1.4.5. Кластеризація на основі центроїдів 35

1.4.6. Кластеризація на основі розподілу 41

1.4.7. Розмита кластеризація 42

1.4.8. Практичні рішення щодо опрацювання даних 45

1.5. Постановка задачі 46

1.6. Висновки до розділу 1 47

РОЗДІЛ 2. МАТЕМАТИЧНІ МЕТОДИ КЛАСТЕРИЗАЦІЇ РІЗНОТИПОВИХ ДАНИХ ТА АЛГОРИТМІВ ОПРАЦЮВАННЯ РІЗНОТИПОВИХ ДАНИХ **49**

2.1. Аналіз методів кластеризації даних 49

2.2. Формальне визначення профілю користувача 60

2.3. Класифікація профілів користувачів 65

2.4. Визначення зв’язків між критеріями профілю користувача 66

2.4.1. Визначення оцінки рівня задоволеності клієнта 66
2.4.2. Визначення сили зв’язку між об’єктами 69
2.5. Висновки до розділу 2 70
РОЗДІЛ 3. РОЗРОБЛЕННЯ АЛГОРИТМІВ ОПРАЦЮВАННЯ РІЗНОТИПОВИХ ДАНИХ 71
3.1. Розробка алгоритму підготовки даних 71
3.1.1. Алгоритм очищення даних 71
3.1.2. Обробка пропущених значень та заповнення даних 72
3.1.3. Виявлення та видалення дублікатів 80
3.1.4. Виявлення та видалення викидів 86
3.2. Розробка алгоритму препроцесингу даних та зменшення розмірності перед кластеризацією 88
3.2.1. Очищення даних 89
3.2.2. Відбір ознак 90
3.2.3 Зменшення розмірності даних 92
3.2.4. Виділення нових ознак 94
3.3. Розробка методу кластеризації даних із урахуванням ваг 99
3.3.1. Застосування статистичного методу перцентилів 99
3.3.2. Розрахунок початкових центроїдів 103
3.3.3. Розроблення модифікованого методу Mini Batch K-means 104
3.4. Висновки до розділу 3 109
РОЗДІЛ 4. РОЗРОБЛЕННЯ АРХІТЕКТУРИ ТА АПРОБАЦІЯ РЕЗУЛЬТАТІВ 110
4.1. Побудова архітектури інформаційної системи 110
4.2. Проектування структури даних у інформаційній системі 116
4.3. Аналіз та моделювання бізнес-процесів 120

4.4. Оптимізація вартості розгортання системи. Аналіз витрат 124
4.5. Апробація результатів 125
4.5.1. Порівняння швидкодії алгоритмів 130
4.5.2. Результати роботи системи 131
4.6. Висновки до розділу 4 134
ВИСНОВКИ **135**
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ **136**
ДОДАТОК А. АКТИ ВПРОВАДЖЕННЯ **149**

ВСТУП

Актуальність теми

В сучасному світі, де нерухомість відіграє важливу роль у житті людей, виникає потреба в новаторських підходах до удосконалення індивідуального обслуговування, електронної торгівлі нерухомістю та поліпшення задоволення клієнтів. Швидкий розвиток технологій та доступ до великого обсягу даних надають нам унікальну можливість використовувати методи кластеризації даних для створення профілів користувачів, що зацікавлені в нерухомості.

Покращення персоналізації є одним з найважливіших завдань для компаній у сфері нерухомості. Сьогодні клієнти очікують індивідуального підходу та послуг, які відповідають їхнім потребам та уподобанням. За допомогою створення профілів користувачів та використання методів кластеризації даних, можна отримати унікальну можливість розуміти потреби та уподобання клієнтів, що дозволить компаніям забезпечити персоналізовані пропозиції та покращити задоволеність клієнтів.

Електронна комерція нерухомості швидко розвивається та стає більш популярною. Онлайн-платформи надають споживачам зручність та доступність при пошуку, виборі та придбанні нерухомості. Проте, великий обсяг інформації та зростаюча конкуренція вимагають впровадження персоналізованих підходів. Шляхом створення профілів користувачів та їх класифікації за допомогою методу кластеризації даних, компанії можуть пропонувати індивідуальні рекомендації та пропозиції, що найкраще відповідають унікальним потребам та побажанням кожного клієнта.

Покращення обслуговування клієнтів залишається головним завданням для будь-якого бізнесу. У сфері нерухомості, де взаємодія з клієнтами має велике значення, розуміння їхніх потреб, вимог та попереднього досвіду являється критичним. Створення профілів користувачів та використання методу кластеризації даних допомагає систематизувати інформацію про клієнтів, що в

свою чергу сприяє покращенню комунікації, індивідуального підходу та задоволеності клієнтів.

Враховуючи усі зазначені фактори, можна визначити, що підвищення рівня персоналізації, розвиток електронної комерції в галузі нерухомості та поліпшення обслуговування клієнтів є актуальними аспектами сучасного ринку нерухомості.

Зв'язок роботи з науковими програмами, планами і темами.

Дисертація виконувалася відповідно до пріоритетних напрямків науково-дослідних робіт Національного університету "Львівська політехніка", відповідно до координаційних планів Міністерства освіти і науки України. Зокрема, дослідження проводились в рамках держбюджетної теми кафедри систем штучного інтелекту «Методи та засоби інформаційної безпеки та гігієни на основі інтерпретованого штучного інтелекту» (№ державного реєстру 0123U101687).

Метою дисертаційної роботи є розроблення методів та засобів кластеризації різнотипових даних.

Досягнення поставленої мети включає наступні задачі:

1. Провести аналіз існуючих методів кластеризації даних і визначити їхні переваги та недоліки в контексті вирішення завдань персоналізованого підходу до користувачів онлайн-платформ ринку нерухомості.
2. Розробити методику аналізу поведінкових і психографічних характеристик клієнтів з метою побудови профілю клієнта та визначення рівня його задоволеності.
3. Здійснити порівняльний аналіз різних методів кластеризації різнотипових даних на прикладі користувачів, зацікавлених в нерухомості.
4. Модифікувати метод кластеризації різнотипових даних, який враховує структуровані та напівструктуровані дані за допомогою поділу їх на пакети та застосування перцентилів.

5. Розробити алгоритм класифікації профілів клієнтів, який би на першому етапі враховував зважування відгуків, а на другому – використовував кластеризацію для формування профілів клієнтів.
6. Розробити архітектуру інформаційної системи з урахуванням можливостей зниження вартості розгортання системи за допомогою безсерверної архітектури.

Об'єкт дослідження: процес опрацювання різнотипових даних набором методів кластеризації.

Предмет дослідження: методи і засоби опрацювання різнотипових даних.

Методи дослідження: методи кластеризації, метрики оцінки задоволеності користувачів, методи оцінки якості кластеризації, статистичні методи та методи об’єктно-орієнтованого аналізу та проектування.

Наукова новизна одержаних результатів:

- Отримав подальший розвиток метод кластеризації різнотипових даних, який дозволяє працювати зі структурованими та напівструктурованими даними на основі поділу на пакети та врахування зважування характеристик під час препроцесингу даних.
- Вперше побудовано модель профілю клієнта, яка відрізняється від існуючих тим, що включає поведінкові та психографічні характеристики, що дає можливість визначити рівень задоволеності клієнта.
- Вперше розроблено класифікатор профілів клієнтів, який, на відміну від існуючих рішень, застосовує зважування відгуків на першому етапі та кластеризацію на другому, що дає можливість пришвидшити обслуговування клієнтів.

Практичне значення одержаних результатів:

1. Розроблено архітектуру інформаційної системи для здійснення автоматизованого профілювання користувачів на основі різнотипових даних клієнтів. Впровадження запропонованої архітектури призвело до

зниження кінцевої вартості системи в 4 рази порівняно зі стандартною архітектурою, що використовує дроплети.

2. Проведено порівняльний аналіз швидкодії методів кластеризації різнотипових даних на основі набору даних користувачів, зацікавлених у сфері нерухомості. Потокова кластеризація (mini-batch) показує стабільно швидкі результати, але за через розбиття даних на пакети якість кластеризації стає меншою. Це відбувається за рахунок випадкового обрання підмножин даних замість повного набору даних для кожної ітерації.
3. Розроблено метод кластеризації різнотипових даних, що використовує пакети даних, перцентильні аналізи та вагові коефіцієнти з метою покращення швидкодії та збереження якості кластеризації.

Особистий внесок здобувача: Основні положення та результати дисертаційної роботи одержані автором самостійно. Особисто здобувачеві належать наступні наукові результати: розроблення методу кластеризації різнотипових даних [97]; побудова ієрархічного класифікатора профілів [98], побудова профілів користувачів [99]. Запропоновано архітектуру системи обробки інформації [100], розроблено підхід для опрацювання різнотипових даних [101], проаналізовано роботу методів кластеризації різнотипових даних [102, 103].

Апробація результатів дисертації: Результати дисертаційних досліджень доповідались на: 7-й міжнародній конференції молодих вчених в Нью-Йорку, 5-му міжнародному семінарі з комп'ютерного моделювання та інтелектуальних систем у Запоріжжі, 3-му міжнародному семінарі з інтелектуальних інформаційних технологій та систем інформаційної безпеки у Хмельницьку. Також результати доповідались на семінарах кафедри систем штучного інтелекту «Національного університету «Львівська політехніка»..

Структура та обсяг дисертації: Дисертаційна робота викладена на 149 сторінках та складається із змісту, вступу, чотирьох основних розділів, в яких

містяться 27 рисунків, 8 таблиць, списку використаних джерел із 108 найменувань та додатків.

РОЗДІЛ 1. АНАЛІЗ ТЕХНОЛОГІЇ КЛАСТЕРИЗАЦІЇ РІЗНОТИПОВИХ ДАНИХ

У розділі здійснено огляд та аналіз методів кластеризації даних, проаналізовано та описано, що собою представляють різнотипові дані, описано задачу профілювання, та проаналізовано стратегії збору даних.

Результати розділу опубліковано у працях автора [97, 103]

1.1. Оцінка впливу штучного інтелекту на нерухомість і персоналізацію споживачів

Інвестиції в штучний інтелект (ШІ) суттєво змінили ландшафт різних галузей, і ринок нерухомості не є винятком. У 2022 році інвестиції в ШІ на світовому ринку нерухомості були оцінені в 351,9 мільйона доларів США, і прогнозується, що до 2032 року ця сума зросте приблизно до 1,047 мільярда доларів США. Це значне збільшення підкреслює зростаючу залежність та інтеграцію технологій ШІ у секторі нерухомості.

Інтеграція ШІ в сфері нерухомості кардинально змінила галузь, зробивши процеси більш ефективними та орієнтованими на дані. Найпоширеніші та практичні застосування ШІ в сфері нерухомості включають прогнозування, 3D-моделювання, розрахунки іпотечних платежів та використання чат-ботів для вирішення поширених запитань. Ці інструменти не тільки оптимізують операції, але й покращують взаємодію з клієнтами, надаючи їм більш точну та персоналізовану інформацію.

Однак, хоча ці нововведення призвели до значних змін, вплив ШІ на персоналізацію – це питання, яке вимагає подальшого дослідження. Потреба сучасних споживачів полягає у пошуку персоналізованого підходу при взаємодії з бізнесом. Користувачі прагнуть отримувати індивідуальний підхід та готові відмовитися від послуг компанії, якщо вона не може запропонувати персоналізований підхід, або не може виділитись на тлі конкурентів.

Персоналізований маркетинг – адаптація рекламних повідомлень, пропозицій та інформації про об'єкти нерухомості згідно з індивідуальними

потребами та інтересами потенційних покупців або орендарів. Це може включати в себе використання даних про попередні пошуки та перегляди об'єктів нерухомості конкретного користувача для надання йому пропозицій, які найкраще відповідають його запитам та потребам. Персоналізований підхід допомагає агентам нерухомості та власникам об'єктів більш точно та ефективно взаємодіяти з потенційними клієнтами, підвищуючи задоволеність клієнтів та збільшуючи шанси на успішну угоду. Основні виклики впровадження персоналізації в маркетинг на ринку нерухомості полягають у логістичних труднощах та необхідності збирати великий обсяг інформації для ефективного використання цього підходу. Традиційно, ручне управління персоналізацією виявилося економічно невигідним і недостатньо практичним у великих масштабах. З появою штучного інтелекту та автоматизації маркетингу ситуація почала змінюватися. Бренди вклали великі ресурси в автоматизацію маркетингових процесів з метою відтворення застарілого підходу до обслуговування клієнтів за загальними правилами. Важливим висновком є те, що автоматизація маркетингу та персоналізація є взаємопов'язаними. В той час як автоматизація забезпечує економію коштів та підвищення ефективності бізнесу, персоналізація є ключовим компонентом для забезпечення унікальності взаємодії з клієнтом. Цей перехід до більш автоматизованих і водночас персоналізованих методів маркетингу відображає стрімкий розвиток технологій та зміну вимог споживачів.

Дослідження [104] показує, що 71% споживачів вважають персоналізацію важливою, а 88% - вважають, що досвід, який компанія надає, є таким же важливим, як і її продукція або послуги. Більше того, 56% користувачів очікують, що пропозиції будуть персоналізовані, а 65% розраховують на негайну відповідь, коли вони звертаються до компанії. В той же час, 71% компаній вважають, що вони мають добре розроблену маркетингову персоналізацію, але тільки 34% споживачів погоджуються з цією точкою зору.

Це розбіжність показує, що існує розрив між тим, що компанії вважають, що вони надають, і тим, що насправді відчувають споживачі.

Авторами [105] розглянуто використання ШІ у сфері нерухомості, що дозволяє зменшити розрив між очікуваннями, надаючи більш персоналізовані послуги. Втім, очевидно, що ще необхідно вкласти значні зусилля для того, аби повноцінно реалізувати потенціал штучного інтелекту в сфері персоналізації послуг. Науково-дослідні підходи [106] та методи, засновані на даних, можуть відігравати ключову роль у розробці нових алгоритмів та систем, здатних адаптуватися до індивідуальних потреб та переваг споживачів.

Через постійний розвиток ринку нерухомості, компаніям необхідно пріоритетно визначити стратегії залучення клієнтів, спираючись на персоналізацію. Підходи [107], засновані на аналізі великих даних та машинному навчанні, можуть допомогти в ідентифікації та систематизації потреб клієнтів, а також забезпечити адаптивність послуг.

Продуктивне використання ШІ у галузі нерухомості має можливість підняти рівень взаємодії із клієнтами, запропонувавши їм більш точні та персоналізовані рішення. В дослідженні [108] описана методологія ефективного застосування штучного інтелекту, але існують і нерозвідані горизонти та можливості, що дозволяють повноцінно використовувати методи машинного навчання для покращення процесу створення персоналізованих рішень. Тільки розробка стратегій та забезпечення їх відповідності індивідуальним потребам споживачів, зможуть забезпечити компаніям конкурентну перевагу на ринку нерухомості.

1.2. Визначення різнотипових даних

Різнотипові дані, або "heterogeneous data" – це складна категорія даних, яка об'єднує в собі структуровані, напів-структуровані та неструктуровані дані, що використовуються разом. Вони можуть містити різні форми і типи даних, що зберігаються в різноманітних джерелах і форматах, включаючи числа, текст,

зображення, аудіо та відеофайли, електронні таблиці, бази даних, веб-сторінки тощо [1].

В основі, різнотипові дані мають певний рівень неконсистентності або невідповідності, що виникає через різноманітність джерел та форматів, в яких вони зберігаються і обробляються [2]. Неконсистентність та невідповідність може виникнути з декількох причин:

1. Різноманітність джерел даних:
    - Дані можуть надходити з різних джерел, які мають свої власні стандарти та формати зберігання інформації.
    - Може виникнути неузгодженість між даними з різних джерел, наприклад, в одному джерелі ім'я користувача може бути зазначено повністю, а в іншому - лише ініціали.
2. Різноманітність форматів даних:
    - Дані можуть бути збережені у різних форматах, наприклад, текст, числа, зображення тощо.
    - Переведення даних з одного формату в інший може призвести до втрати інформації або її спотворення.
3. Проблеми якості даних:
    - Дані можуть містити помилки, відсутність інформації, дублікати тощо.
    - Ці проблеми можуть ускладнити обробку даних та вплинути на точність результатів.

Ця категорія даних зазвичай потребує складних методів для їх обробки та аналізу, зокрема кластеризації, профілювання, шаблонного аналізу та інших [3].

Структуровані дані - це дані, що мають чітко визначену структуру, такі як числові дані або дата/час. Вони легко зберігаються в реляційних базах даних і можуть бути легко проаналізовані [4].

Напівструктуровані дані, з іншого боку, мають деякі структурні характеристики, але не такі чітко визначені, як у структурованих даних. Вони

можуть включати дані, що зберігаються в форматі XML або JSON, а також електронну пошту.

У свою чергу, неструктуровані дані не мають визначеної форми або структури і можуть включати такі дані, як текст, зображення, відео та аудіо [5]. Неструктуровані дані не можна організувати та зберегти у реляційних базах даних. Недостатність контролю над неструктурованими даними є одним із найбільших обмежень для аналізу цих даних. Неструктуровані дані відрізняються від структурованих тим, що вони не мають попередньо визначеної структури чи схеми, яка б організовувала їхній зміст. До таких даних відносяться тексти, зображення, відео, аудіо та інші форми інформації.

В таблиці 1.1. наведено порівняльну характеристику між структурованими та неструктурованими даними.

Таблиця 1.1

Порівняльна характеристика між структурованими та неструктурованими даними

Характеристика	Структуровані дані	Неструктуровані дані
Визначення	Дані, що організовані та відформатовані специфічним чином, відповідно до попередньо визначеної моделі або схеми	Дані, що не мають специфічної структури або формату, як правило, є неорганізованими або в сирому вигляді
Організація	Добре організовані з визначеним форматом, таким як таблиці і колонки	Не мають визначеного формату і є неорганізованими
Доступність	Дуже доступні і можуть бути легко отримані за допомогою структурованої мови запитів (SQL) або інших інструментів баз даних	Менш доступні і вимагають використання високорівневих технік для екстракції та аналізу
Приклади	Інформація про клієнтів, записи про транзакції, списки інвентарю, фінансові дані	Електронні листи, повідомлення в соціальних мережах, мультимедійні файли, дані з сенсорів

Продовження таблиці 1.1

Характеристика	Структуровані дані	Неструктуровані дані
Аналіз	Легко аналізуються за допомогою традиційних статистичних методів та технік видобування даних	Вимагають використання високорівневих технік, таких як обробка природної мови (NLP) та машинне навчання, для аналізу
Масштабованість	Мають обмежену масштабованість через визначені схеми та фіксовані структури даних	Високо масштабовані і можуть вміщувати будь-який тип даних без зміни існуючої структури
Використання	Бізнес-аналітика, аналіз даних, фінансова звітність	Аналіз настроїв, моніторинг соціальних мереж, текстовий аналіз

Аналіз різнотипових даних є складним процесом саме через їх різноманітність. Вони можуть містити великі обсяги інформації, що є важливими для аналізу. Проте дані потребують використання складних методів для їх обробки і аналізу: глибинне навчання, текстовий аналіз за допомогою NLP (Natural Language Processing), комп'ютерний зір та ін. Ці методи можуть включати кластеризацію, профілювання, шаблонний аналіз та інше.

Різнотипові дані мають певний рівень неконсистентності або невідповідності через різноманітність джерел і форматів, в яких вони зберігаються і обробляються. Вони можуть містити дублювання, неповні та суперечливі дані. Важливо розробити методи обробки і аналізу, які враховують цю неконсистентність, щоб забезпечити точність аналізу.

Дослідники [6] пропонують різні методи обробки різнотипових даних, включаючи підходи на основі машинного навчання. Використання методів для обробки різнорідних даних включає застосування кластеризації, класифікації, регресійного аналізу та підкріпленого навчання з метою ідентифікації та дослідження закономірностей у цих даних.

Ключовим аспектом різнотипових даних є їхнє різноманітне походження і різноманітність форм та форматів, в яких вони можуть бути представлені. Ці дані включають різні типи інформації, що зберігаються і обробляються в різноманітних джерелах, що призводить до певного рівня неконсистентності

або невідповідності. Незважаючи на це, вони можуть містити важливу інформацію, яка може бути використана для аналізу за допомогою розроблених методів обробки, включаючи кластеризацію, профілювання, шаблонний аналіз та інші.

На ринку нерухомості, різнотипові дані можуть бути використані для прогнозування цінових тенденцій, аналізу попиту та пропозиції, виявлення закономірностей та ризиків, для забезпечення кращого взаємодії з клієнтами та поліпшення процесів комунікації. Прикладом таких даних можуть бути такі:

- Основна інформація про користувача, така як: ім'я, вік, стать, місце проживання.
- Дані про взаємодію з системою ринку нерухомості, такі як час і дата відвідування, переглянуті сторінки, продукти або послуги, які вони вибрали або купили.
- Дані про перегляд оголошень та покупки, такі як вид нерухомості, розташування, ціновий діапазон.
- Дані про відгуки користувачів.
- Геолокаційні дані.

Важливо зазначити, що наведені вище дані можуть бути повністю структурованими, оскільки у них є чітка схема, яка визначає типи даних, відносини між таблицями та індекси. Далі можна розглянути приклад неструктурованих даних:

- Відгуки користувачів: це можуть бути відгуки, які вони залишили, або відгуки, які вони читали.
- Повідомлення в соціальних медіа: це може включати їхні публікації, коментарі або вподобання в соціальних медіа.
- Текстові чи e-mail повідомлення, які вони відправили, або отримали від системи.
- Дані про пошук: це можуть бути ключові слова, які вони використовували під час пошуку в системі, або результати, які вони вибрали.

Структуровані дані можна описати предикатним правилом. Нехай H – це набір об'єктів, який складається з окремих об'єктів $\{O_1, O_2, ..., O_n\}$, де N – загальна кількість об'єктів у H, а O_i - і-й об'єкт у H. Кожен об'єкт O_i визначається унікальним ідентифікатором об'єкта, O_i.ID. Для доступу до ідентифікатора та інших складових частин об'єкта використовується крапкова нотація – синтаксис, який використовується для доступу до властивостей чи методів об'єкта в багатьох мовах програмування, таких як JavaScript, Python та інших. У наборі різнотипових даних об'єкти також визначаються кількістю компонентів або елементів $O_i = \{E^1_{O_i}, ..., E^j_{O_i}, ..., E^M_{O_i}\}$, де M представляє загальну кількість елементів, а $E^j_{O_i}$ представляє дані, що стосуються E_jдля O_i. Кожен повний елемент E^j, для $1 \leq j \leq M$, можна вважати як представлення та збереження різних типів даних. Тому H можна розглядати із двох різних перспектив: як множину об'єктів, що містять дані для кожного елемента, або як множину елементів, що містять дані для кожного об'єкта. Обидва підходи дозволяють отримати необхідну інформацію для майбутнього створення функцій, необхідних для кластеризації даних. Наприклад, O_3вказує на всі доступні елементи для об'єкта 3 (наприклад, конкретного користувача з певним ідентифікатором); $O_3.E_2$ вказує на другий елемент для об'єкта 3 (наприклад, набір відгуків користувача); E_2 вказує на всі значення об'єктів для елемента 2 (наприклад, всі набори відгуків усіх користувачів).

1.3. Визначення задачі профілювання

Профілювання – це процес збору та аналізу інформації з метою виявлення характерних закономірностей або атрибутів. У контексті обробки даних і машинного навчання, це часто включає використання статистичного аналізу (середнє, медіана, мода, стандартне відхилення та ін.) та моделей машинного

навчання (класифікація, регресія, кластеризація, зменшення розмірності та ін.) для виявлення критеріїв в даних [7].

Крім того, Шихаб, та ін. [8] визначають профілювання як процес виявлення та вивчення характеристик об'єкта з метою класифікації. Це може включати вивчення окремих атрибутів, таких як вік, стать, місце роботи, кількість осіб у сім’ї, дохід чи місце проживання, а також взаємозв'язки між різними атрибутами.

Визначення профілювання [9] акцентує увагу на використанні профілювання для виявлення аномалій або виокремлення значущих критеріїв в наборі даних (таких як рівень доходу та кількість неповнолітніх дітей у сім’ї). Вони зазначають, що профілювання може використовуватися для виявлення виключень з нормальних шаблонів, які можуть вказувати на неправильну поведінку або можливі проблеми. У даному випадки шаблони містять типові патерни поведінки користувача: характеристики користувацької активності в системі ринку нерухомості, реакція на пропозиції, звички, тощо.

Профілювання користувачів – це субкатегорія профілювання даних, що фокусується на зборі та аналізі даних, які стосуються окремих користувачів або груп користувачів. Це зазвичай включає аналіз даних про поведінку користувачів, персональні характеристики (психографічні, соціальні, демографічні, тощо), інтереси та уподобання. Головною метою профілювання користувачів є створення так званого "профілю", який розкриває тенденції в поведінці користувача, які можуть бути використані для прогнозування майбутньої поведінки та інтересів [10].

Згідно з визначенням Маурица та Сільвестріса [11], профілювання користувачів полягає в процесі виявлення, збору та аналізу даних про користувачів з метою розробки детального "профілю" користувача, який відображає інтереси, поведінку та характер користувача. Такий профіль може бути використаний для прогнозування майбутніх дій користувача, рекомендацій, або налаштування персоналізованих послуг.

Інше визначення [12] пояснює, що профілювання користувачів може включати збір і аналіз не тільки даних про поведінку користувача, але й даних про контекст, в якому відбувається ця поведінка. Це може включати аналіз даних про місце, час, ситуацію та інші фактори, що впливають на поведінку користувача. Вони зазначають, що цей "контекст" може бути критично важливим для точного прогнозування та розуміння поведінки користувача.

Загалом, профілювання користувачів включає збір, аналіз та інтерпретацію даних про користувача з метою розробки деталізованого профілю, який відображає інтереси, характеристики та поведінку користувача. Ця інформація може бути використана для різноманітних цілей, включаючи рекомендаційні системи, персоналізацію послуг та прогнозування поведінки користувача.

Цілями профілювання є:

Покращення якості рекомендацій – завдяки профілюванню користувачів система пошуку нерухомості може надавати більш точні, релевантні та персоналізовані рекомендації нерухомостей. Аналізуючи дані про вподобання та попередні дії користувачів, система може зібрати більше інформації про їхні очікування, що сприяє наданню більш точних рекомендацій.

Підвищення задоволеності користувача – коли система здатна точно визначати та задовольняти потреби користувача, це призводить до збільшення задоволеності користувача, лояльності та в кінцевому підсумку призводить до повторного використання послуг.

Покращення користувацького досвіду. Користувацький досвід - це сприйняття, емоції та враження, що виникають у користувача під час взаємодії з продуктом, системою, послугою або веб-сайтом. Він відображає якість та задоволення, які користувач отримує в результаті використання даного продукту чи послуги. Шляхом персоналізованих рекомендацій, система може зробити пошук нерухомості більше зручним та ефективним для користувачів. Вони отримують доступ до нерухомостей, які краще відповідають їхнім потребам, і

не витрачають час на перегляд не цікавих об'єктів. Це сприяє підвищенню задоволення користувачів, забезпеченню більшого комфорту та зменшенню зусиль у пошуку нерухомості [28].

Існує декілька підходів для створення профілів користувачів – статистичне, кластерне, багатовимірне та поведінкове профілювання. Вони базуються на відповідних вимогах. Кожен із методів має свої особливості та переваги.

Статистичне профілювання користувачів є процесом, в якому збираються і аналізуються статистичні дані про користувача або групу користувачів. Воно включає в себе збір різних типів даних, таких як вік, стать, відгуки, досвід із ринком нерухомості, місце проживання, інтереси, та ін., та їх аналіз для виявлення загальних шаблонів та тенденцій. Дані потім використовуються для створення профілю користувача [13]. За допомогою статистичного профілювання система ринку нерухомості може виявити, що середній вік її користувачів складає 25 років і що вони в середньому проводять на платформі 15 хвилин.

Багатовимірне профілювання – це метод аналізу даних, який використовує багатовимірні статистичні моделі для створення детальних профілів користувачів на основі різних характеристик і взаємодій. Цей метод здатен об'єднати велику кількість інформації з різних джерел і зробити більш глибокий і комплексний аналіз. До прикладу, це може бути створення профілю користувача, який включає інформацію про його вік, стать, інтереси, історію покупок, час відвідування сайту тощо.

Поведінкове профілювання користувачів зосереджується на зборі та аналізі даних про поведінку користувача. Це може включати дані про веб-перегляд, пошукові запити, покупки, соціальні взаємодії, та ін. Ця інформація потім використовується для розуміння інтересів та передчуттів користувача, що може підвищити якість персоналізованих рекомендацій. Типовим прикладом такого профілювання є виявлення того, що певний

користувач зазвичай відвідує сайт в ранкові години та переглядає пропозиції з конкретним певним набором фільтрів.

Кластерне профілювання користувачів полягає в групуванні користувачів на основі схожості в їхніх профілях. Техніка кластеризації дозволяє виявити групи користувачів з подібними інтересами та поведінкою, що може бути використано для персоналізації сервісів та рекомендацій [14]. Щоб згрупувати користувачів, перш за все необхідно визначити критерії схожості (наприклад із допомогою кореляційного аналізу). Це можуть бути демографічні дані (вік, стать, місцезнаходження), історія покупок, перегляди веб-сторінок, соціальні активності тощо. Далі необхідно провести кластеризацію даних. Після кластеризації користувачів їх можна розділити на групи, засновуючись на їхніх схожих інтересах і поведінці. Коли групи користувачів визначені, компанії можуть надавати їм більш цільові рекомендації. Прикладом такого профілювання може бути виявлення групи користувачів віком в певному діапазоні, які зацікавлені в оренді нерухомості. Саме цей підхід і буде застосований у даній роботі, оскільки він дозволить розділити користувачів на певні групи та допоможе працювати із ними у більш вузькому профілі.

1.4. Аналіз існуючих рішень щодо збору та опрацювання даних

1.4.1. Методи збору даних

Збір даних – це систематичний процес збору та вимірювання інформації з різних джерел. Він виконується для отримання необхідної інформації і подальшого аналізу для досягнення певної дослідницької мети. Збір даних може бути кількісним або якісним і може здійснюватися в реальному часі або ретроспективно, у заданий проміжок часу [15].

Методи збору даних відносяться до планування і виконання процесу збору даних з метою забезпечення точності, якості та достовірності зібраної інформації. Ці стратегії можуть включати вибір найбільш відповідних методів для збору даних, визначення кількості даних для збору, планування і розробка

засобів збору даних, визначення, як і коли збирати дані, і визначення, як дані будуть зберігатися та аналізуватися. Серед найпоширеніших методів збору даних можна виділити наступні:

- Прямий збір даних: цей підхід включає в себе збір даних безпосередньо від джерела. Наприклад, це можуть бути дані, які користувач вводить в систему купівлі/продажу нерухомості, такі як інформація про профіль, запити пошуку та історія транзакцій.
- Непрямий збір даних: цей підхід включає в себе збір даних без безпосередньої взаємодії з користувачем. Наприклад, це можуть бути дані про активність користувача в соціальних мережах, такі як уподобання, коментарі, публікації та відгуки.
- Пасивний збір даних: цей підхід включає в себе збір даних в процесі нормальної діяльності користувача. Наприклад, це можуть бути дані, зібрані з використанням кукі (cookies) або журналів веб-сервера.
- Активний збір даних: цей підхід включає в себе збір даних за допомогою активних взаємодій з користувачем, таких як опитування, анкетування або інтерв'ю.

Кожен з цих методів має свої переваги та недоліки, і вибір певної стратегії залежить від конкретного контексту та цілей дослідження. Для ефективного збору даних часто використовують комбінацію різних стратегій.

В залежності від контексту, потреб дослідження та доступності джерел інформації, інструменти збору даних можуть суттєво варіюватися. Ось деякі приклади збору даних:

Анкетування - один з найпоширеніших методів збору даних, який використовується для отримання інформації від великої кількості людей. Анкети можуть включати різні типи запитань, включаючи відкриті та закриті питання, та можуть бути проведені в різних форматах, включаючи паперові анкети, онлайн-анкети та телефонні опитування [16].

Спостереження можуть використовуватися для збору даних про поведінку людей, процеси або події. Спостереження можуть бути структуровані (за наперед визначеними категоріями) або неструктуровані (з відкритими категоріями для записування всього, що відбувається).

Інтерв'ю можуть бути використані для збору детальної інформації про думки та досвід людей. Інтерв'ю можуть бути структуровані (за наперед визначеними питаннями), напівструктуровані (з відкритими питаннями та можливістю відхилення від них) або неструктуровані (без фіксованих питань).

Аналіз документів і записів включає в себе збір даних з різних джерел, таких як газети, журнали, щоденники, веб-сайти, урядові документи, статистичні бази даних та ін..

Експерименти використовуються для вивчення причинно-наслідкових зв'язків між різними змінними. Причинно-наслідкові зв'язки - це взаємозв'язки між подіями чи явищами, коли одна подія чи явище є причиною іншого. Іншими словами - це зв'язки, де одна змінна впливає на іншу змінну. Вони включають в себе контрольоване маніпулювання однією або декількома незалежними змінними та вимірювання їх впливу на залежну змінну.

Методи машинного навчання і аналітики великих даних використовуються для автоматичного збору, обробки та аналізу великої кількості даних. Ці методи включають в себе використання алгоритмів машинного навчання та статистичних моделей для отримання інсайтів з даних.

Збір інформації про користувачів, які цікавляться нерухомістю, може включати такі аспекти:

- Демографічна інформація: дані про вік, стать, професію, сімейний стан користувача і так далі. Ці дані можуть допомогти у формуванні базового профілю користувача і визначенні його потреб та можливостей щодо купівлі нерухомості.
- Дані про пошук: інформація про те, які об'єкти нерухомості користувач відвідав, які запити він вводив, і які оголошення нерухомості він відкрив.

Це може допомогти зрозуміти, які критерії та параметри нерухомості важливі для користувача.

- Історія транзакцій: дані про раніше придбану нерухомість, її характеристики, вартість, місцезнаходження, і так далі. Відгуки та відповіді: Це можуть бути відгуки, які користувач залишав про конкретні об'єкти нерухомості, а також його відповіді на запитання анкет, опитувань, тощо.

Щодо збору даних з соціальних мереж, вони можуть включати такі аспекти:

- Інформація з профілю користувача: дані з профілю користувача в соціальних мережах, такі як вік, стать, професія, місце проживання, інтереси.
- Активність в соціальних мережах: публікації, коментарі, вподобання та інші дії, пов'язані з темою нерухомості.
- Дані соціальних графів: дані про соціальні зв'язки користувача, його друзів, відношення до інших учасників.
- Відгуки та відповіді: Це можуть бути відгуки та відповіді користувача на публікації, статті, оголошення, пов'язані з нерухомістю.

Усі ці дані можна використовувати для детального профілювання користувача, яке в подальшому може слугувати основою для кластеризації користувачів на основі їх поведінки, інтересів, потреб та можливостей.

1.4.2. Сімейства алгоритмів кластеризації даних

Кластеризація – це метод машинного навчання без вчителя, який широко використовується для групування схожих об'єктів. В контексті різнотипових даних, що містять інформацію про користувачів, зацікавлених у нерухомості, а також метрики з соціальних мереж, кластеризація може допомогти виявити закономірності та тенденції у поведінці користувачів [17].

Існує декілька сімей алгоритмів кластеризації даних, які використовуються для групування набору даних на основі схожості їх характеристик – кластеризація на основі зв'язку (ієрархічна кластеризація), кластеризація на основі центроїдів (методи розділення), кластеризація на основі розподілу, кластеризація на основі щільності (методи, що базуються на моделях) та розмита кластеризація.

1.4.3. Кластеризація на основі щільності

Кластеризація на основі щільності — це тип кластеризації, що в основному зосереджується на ідентифікації областей в просторі даних, де спостереження зосереджені щільно. Вона є особливо корисною, коли області високої щільності розділені областями низької щільності.

У кластеризації на основі щільності, дані групуються в області, де точки мають високу концентрацію, оточені областями з меншою концентрацією. Алгоритм виявляє такі області, де щільність точок є великою, і визначає їх як кластери. Однією з головних переваг цього підходу є те, що кластери можуть мати будь-яку форму, не обмежуючись очікуваними умовами. Алгоритми цього типу не спрямовані на віднесення викидів до кластерів, тому вони не звертають на них увагу.

У цьому алгоритму кластеризації області високої щільності точок об'єднуються в кластери. Це дозволяє моделювати розподіл даних з будь-якою формою, за умови, що області високої щільності можуть бути з'єднані. У алгоритмів цього типу можуть виникати проблеми з даними, що мають змінну щільність та високу розмірність. Крім того, за своєю конструкцією ці алгоритми не намагаються включати викиди до кластерів, тому їм не приділяється увага.

У роботі [20] автори провели експериментальне дослідження ефективності та продуктивності алгоритму на основі щільності, який називається DBSCAN, використовуючи синтетичні дані та реальні дані з бенчмарку SEQUOIA 2000 та довели більшу ефективність роботи цього методу, аніж метод CLARANS з точки зору продуктивності через хорошу якість для

кластерів довільної форми. Однак, серед ключових труднощів, що виникають під час роботи з різнотиповими даними, слід відзначити те, що різні види даних можуть мати не однакову вагу та значущість, а також розташовуватися на різних шкалах. Це може вимагати нормалізації або стандартизації даних перед кластеризацією.

1.4.4. Ієрархічна кластеризація

Ієрархічна кластеризація є одним з методів кластерного аналізу, який дозволяє групувати об'єкти у класи або кластери на основі їхньої подібності. Вона будує ієрархічну структуру кластерів, де кожен об'єкт може бути представлений як окремий кластер або підкластер, або злитий з іншими кластерами для формування більших кластерів [18].

Ієрархічна кластеризація може бути представлена у вигляді дерева або дендрограми, де вершини представляють кластери або підкластери, а ребра вказують на з'єднання між ними. Дендрограма надає інформацію про подібність між кластерами та їхню ієрархію. Приклад дендрограми зображено на рис. 1.1.

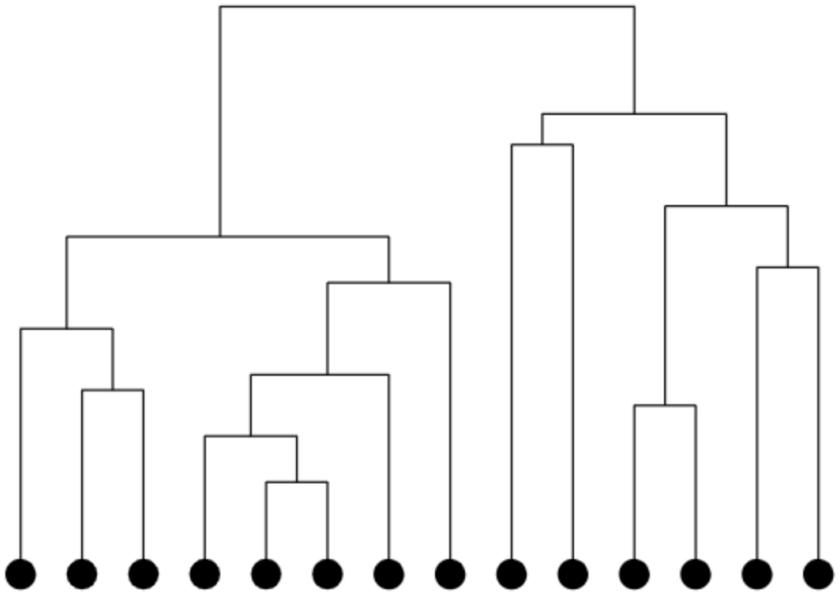

Рис.1.1 Приклад дендрограми ієрархічної кластеризації

Існують два підходи ієрархічної кластеризації - агломеративний та дивізивний. Їх спільними рисами є те, що обидва підходи використовують

ієрархічну структуру для представлення кластерів, виконують послідовне об'єднання (агломеративний) або розбиття (дивізивний) кластерів для формування кінцевої структури кластерів. Вони також базуються на вимірюванні відстаней або подібності між об'єктами для визначення ступеня подібності або віддаленості між кластерами. Відмінність цих підходів полягає у тому, що вони використовують рекурсивні стратегії, але в протилежних напрямках [19].

- Агломеративні алгоритми кластеризації: Ці алгоритми розпочинають з того, що кожне спостереження розглядається як окремий кластер, а потім ці кластери об'єднуються на основі відстані або схожості між ними. Наприклад, для даних про користувачів, зацікавлених у нерухомості, можна використовувати агломеративні методи для групування користувачів на основі спільних характеристик, таких як ціновий діапазон, тип нерухомості або географічний регіон, оскільки вони дозволяють згрупувати користувачів на основі спільних характеристик, поступово об'єднуючи найбільш схожі об'єкти у все більші кластери, що допомагає виділити загальні тренди і закономірності в даних..
- Дивізивні алгоритми кластеризації: на відміну від агломеративних, дивізивні алгоритми розпочинають процес кластеризації з усіх даних, розглядаючи їх як один єдиний великий кластер, після чого послідовно розбивають дані на менші підмножини через рекурсивне відокремлення. Це може бути використано, наприклад, для розбиття метрик соціальних медіа на менші групи, такі як підгрупи користувачів з різними інтересами або паттернами взаємодії.

Отже, на виході агломеративної ієрархічної кластеризації отримується дендрограма, яка візуалізує ієрархію кластерів. Вершини дендрограми представляють окремі об'єкти або об'єднані кластери, а відстань між вершинами вказує на подібність між ними. На виході дивізивної ієрархічної кластеризації отримується дендрограма, яка візуалізує ієрархію кластерів, але в цьому

випадку дендрограма побудована знизу вгору, починаючи з найменших кластерів та об'єднуючи їх у більші кластери. На рис. 1.2. зображено алгоритм роботи цих підходів.

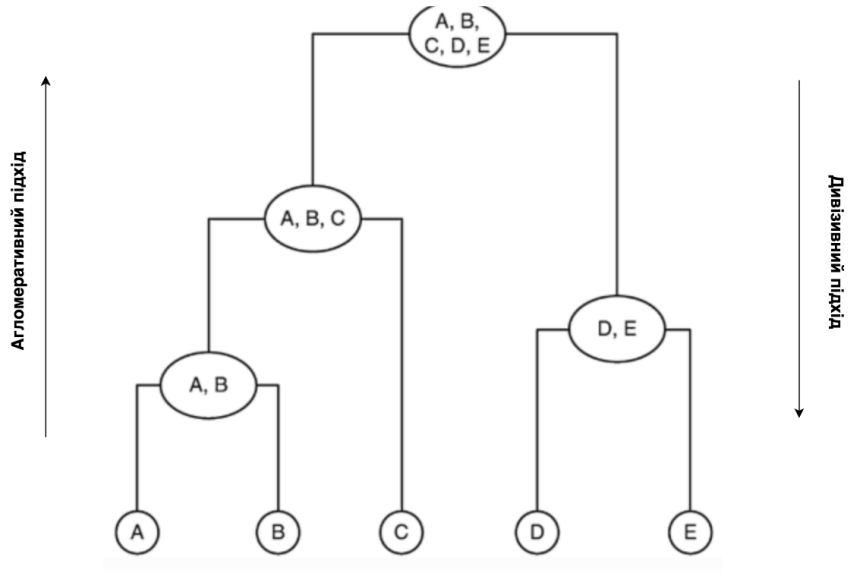

Рис. 1.2. Алгоритм роботи агломеративного та дивізивного підходів

В дослідженні [21] автор розглядає різні алгоритми ієрархічної кластеризації, надаючи детальний огляд цієї теми. Проводиться порівняльний аналіз декількох алгоритмів ієрархічної кластеризації, включаючи метод Ворда, кластеризацію k-середніх та спектральну кластеризацію, з метою отримання уявлення про їхні переваги та недоліки. Автор статті [21] також розглядає актуальні напрямки досліджень у галузі ієрархічної кластеризації, такі як напів наглядна кластеризація та кластеризація на основі графів, і наголошує на потенційних застосуваннях ієрархічної кластеризації в різних галузях.

1.4.5. Кластеризація на основі центроїдів

Кластеризація на основі центроїдів – це метод кластеризації, в якому кластери визначаються на основі "центрів" або "центроїдів" кластерів. Центроїди визначаються як середнє арифметичне всіх точок в кластері. Ці

методи спрямовані на мінімізацію варіації всередині кожного кластера та максимізацію варіації між кластерами [22].

Алгоритм кластеризації на основі центроїдів розпочинається з випадкового вибору кількох центроїдів в межах досліджуваної вибірки даних. Потім кожен об'єкт даних призначається до найближчого центроїда на основі відстані між ними. Після цього обчислюється нове положення центроїдів, яке відображає середні значення атрибутів у кожному кластері.

Процес кластеризації повторюється ітеративно, доки зміщення центроїдів стає мінімальним або досягає певної заздалегідь встановленої точності. На кожній ітерації об'єкти перерозподіляються між кластерами на основі їхньої відстані до центроїдів. Цей процес триває до досягнення стабільної конфігурації центроїдів та призведення до збалансованих та однорідних, тобто чітких кластерів.

Алгоритми кластеризації на основі цетроїдів (наприклад к-середніх), як і більшість інших традиційних методів кластеризації добре працюють лише з однотипними даними – числовими, або не числовими. Автори [23] запропонували підхід до сегментації клієнтів, застосовуючи метод навчання без учителя k-means на основі даних, що були згенеровані за допомогою моделі RFMTS (Recency, Frequency, Monetary, Time) – методу аналізу поведінки клієнтів, який допомагає компаніям краще зрозуміти своїх клієнтів та підвищити ефективність маркетингових кампаній. Метою було покращення взаємин з клієнтами та розробка більш ефективних персоналізованих стратегій маркетингу.

Маркетингові стратегії покращують досвід клієнта, а також підсилюють ефективність маркетингових зусиль у сфері нерухомості. Існує багато маркетингових кампаній для різних напрямів та сфер, проте у сфері нерухомості можна виділити 7 основних:

- Сегментовані е-мейл кампанії – класифікація потенційних клієнтів на основі їхніх специфічних інтересів, таких як житлові та комерційні

об'єкти, інвестування проти житла, або діапазонів бюджету, може призвести до більш дієвої співпраці. Регулярні новини, що описують ринкові тенденції, пропозиції нерухомості, адаптовані до інтересів одержувача, або оновлення місцевого законодавства з нерухомості можна розсилати на основі сегментації користувачів.

- Динамічний вміст веб-сайту – веб-сайти часто служать першою точкою контакту для потенційних інвесторів або покупців. Використовуючи файли cookie для відстеження та аналізуючи поведінку користувача, ріелтори можуть представляти пропозиції або вміст, який відповідає вподобанням користувача. Впровадження плагінів, що керуються ШІ, на платформах нерухомості, які демонструють пропозиції на основі взаємодії користувача, може значно покращити досвід користувача та рівень залученості.
- Е-мейл кампанії на основі поведінки – потенційні інвестори та покупці часто потребують декількох варіантів перед прийняттям рішення. Основа цієї стратегії полягає у надсиланні електронних листів на основі конкретних дій, здійснених користувачами на веб-сайті. Наприклад, якщо користувач зберіг певний список нерухомості, але не запланував візит чи дзвінок, можна буде активувати відстежуючий електронний лист з додатковою інформацією або пропозицією обмеженою за часом. Це може включати надсилання запрошень на віртуальний тур, інформації про іпотеку або фінансування, пов'язану з переглянутими об'єктами, або демонстрацію схожих пропозицій.
- Пропозиції на основі місцезнаходження – географічні вподобання є важливою опцією у рішеннях з нерухомості. Використання даних геолокації для надсилання специфічних сповіщень або пропозицій щодо нерухомості, коли користувач знаходиться поруч із бажаною локацією, може спонукати до негайних дій. Наприклад, якщо потенційний покупець досліджує певний район, сповіщення про день відкритих дверей або

нещодавно виставлену на продаж нерухомість у цій місцевості може бути надіслано в реальному часі.

- Інтерактивний вміст – процес вибору нерухомості часто супроводжується відкриттями. Пропонування інтерактивних інструментів, таких як калькулятори іпотеки, віртуальні екскурсії по нерухомості, або калькулятори інвестицій, може надати персоналізовані рекомендації на основі введення даних користувачем. Такі інструменти не тільки покращують досвід користувача, але й позиціонують ріелтора як партнера, що додає цінність у процесі придбання нерухомості.
- Взаємодія у соціальних мережах – платформи соціальних мереж все більше стають простором для демонстрації нерухомості та обговорень. Ріелтори можуть відстежувати згадування, коментарі або запитання щодо нерухомості та надавати адаптовані відповіді. Регулярна взаємодія через персоналізований вміст, такий як демонстрація об'єктів нерухомості, які відповідають трендам або інтересам підписників, може підвищити видимість та довіру до бренду. Наприклад, створення коротких відео екскурсій по об'єктах нерухомості та мітки для зацікавлених підписників або проведення прямих сесій запитань та відповідей з інвестиціями у нерухомість.
- Відображення шляху клієнта – процес купівлі нерухомості включає декілька етапів, кожен з яких вимагає різної інформації та гарантій. Розуміння точок контакту та джерел, через які клієнти взаємодіють, дозволяє ріелторам надсилати адаптований вміст на кожному етапі. Це може включати первісні пропозиції щодо нерухомості, варіанти фінансування, юридичні консультації або поради з технічного обслуговування після покупки. Детально відображаючи шлях клієнта (покроково описуючи всі етапи для прозорості), ріелтори можуть гарантувати, що вони послідовно задовольняють еволюційні потреби

своїх клієнтів, сформовуючи довіру та сприяючи більш гладким транзакціям.

Стаття [24] розглядає метод кластеризації k-середніх++, який є вдосконаленням стандартного k-середніх, що поліпшує швидкість та якість кластеризації. Незважаючи на те, що точність результату не гарантується, простота та швидкість роботи методу є його великою перевагою на практиці. Шляхом включення дуже простої, випадкової техніки вибору початкових центроїдів до k-means, було отримано алгоритм, який має конкурентну складність *O(logk)* порівняно з оптимальною кластеризацією.

У реальності в дуже багатьох реальних наборах даних зустрічаються різнотипові дані, що складаються з комбінації обох типів. Метод к-середніх, не може належним чином обробляти різнотипові дані, оскільки він не має можливості безпосередньо виміряти Евклідову відстань між векторами, що містять змішані числові та ненумеровані дані [29]. Для кластеризації різнотипових даних було запропоновано кілька алгоритмів, які загалом можна розділити на два типи: ті, що безпосередньо кластеризують різнотипові дані, та ті, що кластеризують різнотипові дані на основі трансформації ознак.

Перший тип кластеризації заснований на ієрархічній кластеризації. У ньому алгоритми включають агломеративну кластеризацію, орієнтовану на схожість (SBAC) [30], розширену самоорганізовану карту [31] та алгоритм кластеризації на основі дисперсії та ентропії (CAVE) [32]. Зокрема, SBAC використовує вимірювання відмінності Гудолла, яке вимірює відстань та щільність для числових та ненумеричних даних. у свою чергу, розширена самоорганізована карта та CAVE будують ієрархії відстаней, які можуть бути застосовані для різнотипових даних. Однак, недоліком цього підходу є те, що ієрархічна кластеризація є обчислювально витратною та не підходить для різних джерел наборів даних.

В порівнянні з ієрархічною кластеризацією, кластеризація на основі центроїдів є менш затратною з обчислювальної точки зору і ефективною для

практичного застосування. Типовий метод кластеризації на основі центроїдів для гетерогенних даних, такий як k-prototypes, використовує різні метри відмінності для оцінки схожості даних. Він використовує евклідову відстань для числових даних і відстань Хеммінга для ненумеричних даних, після чого комбінує ці дві відстані [33]. Однак, встановлення та регулювання вагового коефіцієнта для відстані Хеммінга вимагає ручного втручання. Крім того, лінійне поєднання евклідової та відстані Хеммінга може бути проблематичним через їх різні фізичні значення. Для вирішення цих проблем був запропонований метод KL-FCM-GM (fuzzy c-means на основі інформації Кульбака-Лейблера та гаусово-мультиноміального розподілу) [34]. Цей метод використовує негативний логарифм правдоподібності гаусового розподілу, поєднаний з нечіткістю як міру відмінності для кластеризації змішаних даних в більш складний спосіб. Використовуючи негативний логарифм щільності ймовірності, KL-FCM-GM уникає необхідності прямого поєднання евклідової та відстані Хеммінга. Однак, для KL-FCM-GM все ще потрібно встановлювати параметр для контролю рівня нечіткості. Крім того, припущення про гаусово-мультиноміальний розподіл може бути непридатним для числових частин деяких наборів даних.

Другий тип алгоритмів кластеризації для різнотипових даних використовує трансформацію ознак для уніфікації формату даних та може бути застосованим для кластеризації даних з одним форматом ознак (числовим або ненумеричним). Наприклад, SpectralCAT [35] перетворює числові ознаки в нечислові для кластеризації гетерогенних даних. Проте такий вид трансформації може призвести до втрати інформації, оскільки відстані між даними, що містяться в числових даних, видаляються [31]. З іншого боку, метод калібрування ознак (Feature Calibrating (FC)) є класичним методом, який може перетворювати нечислові ознаки в числові [36]. Однак, такий навчальний алгоритм використовує інформацію про розподіли ймовірностей міток класів і

не може бути застосованим для задачі кластеризації без вчителя. Таким чином, FC не підходить для кластеризації гетерогенних даних.

1.4.6. Кластеризація на основі розподілу

Кластеризація на основі розподілу є одним з основних підходів до завдання кластеризації. Вона передбачає моделювання даних як що вони були взяті з мішання кількох груп або кластерів, кожен з яких представляє окремий статистичний розподіл. Кожна точка даних розглядається як член одного з цих кластерів, і модель намагається ідентифікувати кластери, з яких найімовірніше були взяті дані. До прикладів алгоритмів кластеризації на основі розподілу відносяться алгоритми, які використовують моделі Гаусової суміші або інші статистичні розподіли.

Застосування методу кластеризації на основі розподілу до різнотипових даних може бути складним підходом. До прикладу, якщо використовувати модель Гаусової суміші (GMM), то спершу потрібно вирішити, як нормалізувати і стандартизувати ці декілька типів даних. Це важливий крок, оскільки він впливає на результати кластеризації. Після цього можна застосувати модель Гаусової суміші. Кожен кластер буде представлений як гаусовий розподіл, і кожен елемент буде приписаний до кластеру, з якого ймовірніше всього отримані його дані. Параметри гаусових розподілів (середнє та дисперсія для кожного типу даних) можуть бути використані для інтерпретації кластерів.

Однак, важливо пам'ятати, що GMM робить певні припущення про дані, наприклад, вона припускає, що кластери мають гаусову форму. Якщо це припущення не виконується, то GMM може не дати оптимальних результатів. Цей підхід використовує метод навчання без вчителя, тому "правильних" відповідей для порівняння з прогнозами моделі не існує. Замість цього потрібно використовувати інші метрики (такі як критерій Акаіке, критерій Байєса або крос-валідація), щоб визначити, яка модель працює найкраще.

У статті [25] автори стверджують, що класичні методи модельної класифікації проявляють невелику ефективність, працюючи із різними джерелами наборів даних та розглядають підходи до зменшення розмірності, методи на основі регуляризації, економне моделювання, методи кластеризації підпросторів та методи кластеризації на основі вибору змінних.

1.4.7. Розмита кластеризація

Розмита кластеризація, відома також як нечітка кластеризація – це підхід до кластеризації, який допускає належність об'єкта до декількох кластерів одночасно. Відмінність розмитої кластеризації від традиційної кластеризації полягає в тому, що вона призначає ступінь належності до кластеру для кожного об'єкта, замість простого призначення до одного кластеру. Це дозволяє врахувати неоднозначності та нечіткості, які можуть виникнути при кластеризації реальних даних [26].

Нечітка кластеризація (Fuzzy C-Means (FCM)), може бути використана для обробки різнотипових даних. Алгоритм роботи можна розглянути на прикладі даних про користувачів, та метрик соціальних мереж:

- Крок 1 – підготовка даних: Перш за все, всі дані повинні бути представлені у числовому форматі. Це може вимагати перетворення категорійних даних (наприклад, тип нерухомості, яким користувачі цікавляться, або платформи соціальних медіа, якими вони користуються) в числові значення.
- Крок 2 – визначення числа кластерів: У цьому кроці визначається, скільки кластерів потрібно створити. Це може залежати від специфіки даних і мети.
- Крок 3 – ініціалізація: Обираються випадкові центри для кластерів, або використовуються деякі початкові значення, якщо існують попередні підозри щодо того, де центри кластерів могли б бути.
- Крок 4 – ітераційний процес: Кожен об'єкт присвоюється до кожного кластера з певним ступенем належності. Ці ступені належності

обчислюються на основі відстані від об'єкта до центру кластера. Потім центри кластерів обчислюються знову як середньозважене значення об'єктів, ваги яких - це ступені належності об'єктів до кластера. Ці два кроки повторюються, поки центри кластерів не перестануть змінюватися.

- Крок 5 – Оцінка: Після того, як алгоритм зупинився, отримується розмита приналежність об'єктів до кластерів. Якщо потрібно отримати жорстку приналежність, можна присвоїти кожному об'єкту кластер, до якого він належить найбільше.

Алгоритм FCM є варіацією алгоритму k-means і тому він може мати проблеми з роботою з різнотиповими даними або даними високої розмірності. FCM припускає, що кластери сферичні і мають однаковий розмір, що може не відповідати дійсності для різних типів даних з різними розподілами та формами. FCM чутливий до масштабу даних. Різні типи даних можуть вимагати різних методів масштабування, що може вплинути на результати кластеризації. При застосуванні цього методу до реальних даних може бути корисним спочатку зменшити розмірність даних або нормалізувати числові характеристики.

У роботі [27] вивчається, чи є метод нечіткої кластеризації практично цінним для визначення міських ринків нерухомості порівняно з методами кластеризації, що базуються на класичній (або чіткій) теорії множин. Порівняння результатів нечіткої кластеризації з результатами чіткої теорії множин показує, що нечітка кластеризація дає менш чіткі результати кластерів через сильні шуми, викиди, та відсутність даних, як це часто буває серед різнотипових даних.

У таблиці 1.2. зображено узагальнену інформацію щодо попередньо описаних алгоритмів кластеризації.

Таблиця 1.2

Загальне порівняння алгоритмів кластеризації

Алгоритм	Опис	Переваги	Недоліки	Алгоритми
Ієрархічна кластеризація	Кластери створюються на основі ієрархії даних від верхнього до нижнього рівня	Легко реалізовувати, не потрібно передбачати кількість кластерів наперед, дендрограми легко інтерпретувати	Призначення кластеру є жорстким і не може бути скасованим, висока складність за часом, не працює для великого набору даних.	DIANA, AGNES та ін.
Кластеризація на основі центроїдів	Базується на центроїдах, і точки даних призначаються до кластеру на основі їхньої близькості до центроїда кластеру	Легко реалізовувати, швидка обробка, може працювати з великими обсягами даних, легко інтерпретувати результати	Потрібно передбачити кількість центроїдів наперед, кластери, які утворюються, мають неоднаковий розмір і щільність, піддаються впливу шуму та викидів	k-means, k-medians, k-modes
Кластеризація на основі розподілу	Базується на основі імовірнісного розподілу даних, кластери визначаються за допомогою різних метрик, таких як середнє, дисперсія і т. д.	Кількість кластерів не потрібно визначати наперед, працює зі значеннями даних у реальному часі, метрики легкі у розумінні та налаштуванні	Складний алгоритм та повільна обробка, не може масштабуватися до великого обсягу даних	Gaussian Mixed Models, DBCLASD

Продовження таблиці 1.2

Алгоритм	Опис	Переваги	Недоліки	Алгоритми
Кластеризація на основі щільності	Базується на основі щільності точок даних, також відома як кластеризація на основі моделі	Може обробляти шум і викиди, не потрібно визначати кількість кластерів спочатку, утворені кластери високо однорідні, відсутні обмеження на форму кластерів	Складний алгоритм та повільна обробка, не може масштабуватися до великого обсягу даних	DENCAST, DBSCAN
Нечітка кластеризація	Базований на підході розділення, але точки даних можуть належати до більш ніж одного кластеру.	Може працювати з даними, які накладаються, вища швидкість збіжності	Потрібно визначити кількість центроїдів наперед, піддається впливу шуму та викидів, повільний алгоритм і не може бути масштабований	Fuzzy C Means, Rough k means

1.4.8. Практичні рішення щодо опрацювання даних

Декілька провідних компаній, які працюють на ринку нерухомості, визнали потенціал ШІ і використовують його для вдосконалення своїх послуг. Одними з таких компаній є Zillow, Redfin, Opendoor та Compass

Zillow - це одна з найбільших онлайн-платформ нерухомості у США. Вона використовує AI для прогнозування цін на нерухомість за допомогою своєї функції "Zestimate", яка використовує мільйони точок даних для визначення вартості нерухомості.

Redfin використовує штучний інтелект для аналізу ринку нерухомості і допомоги своїм користувачам знайти ідеальну нерухомість. Їхній алгоритм

дозволяє робити прогнози на основі різноманітних даних, включаючи фактичні продажі, розташування та описи об'єктів нерухомості.

Opendoor використовує AI для створення простого та швидкого способу продажу та купівлі будинків. Їх система оцінки використовує дані про нерухомість, ринок, та історію продажів для визначення конкурентоспроможної ціни пропозиції.

Compass - це платформа нерухомості, яка використовує штучний інтелект для надання менеджерам і користувачам персоналізованої інформації. Вона використовує дані та аналітику для допомоги агентам в покращенні їхнього обслуговування клієнтів.

Якщо йде мова про створення профілів користувачів, то це може включати в себе збір та аналіз великого обсягу даних про користувачів, включаючи їхні інтереси, поведінку, вподобання, фінансові можливості тощо. На основі доступної інформації та аналізу, не всі з наведених компаній активно використовують цей підхід в своїй роботі.

Zillow і Redfin використовують AI для створення прогнозів стосовно вартості нерухомості та для персоналізації рекомендацій, але вони можуть не мати всебічного профілю користувача в тому сенсі, що їх основний фокус - на ринку нерухомості, а не на віддільних користувачах.

Opendoor працює більше з точки зору швидкого обороту нерухомості, що вимагає глибокого розуміння ринку, а не індивідуальних користувачів.

Compass зосереджується на підтримці агентів з нерухомості і, хоча вони надають персоналізовані рекомендації, не повністю зрозуміло, наскільки повні їх профілі користувачів. Крім цього система часто фокусує увагу користувачів на більш дорогі об'єкти та має обмежене географічне охоплення.

1.5. Постановка задачі

Метою даної дисертаційної роботи є побудова інформаційної системи, котра може здійснювати ефективне та автоматизоване профілювання користувачів.

При цьому, інформаційна система повинна бути ефективною з точки зору використання ресурсів. Для забезпечення мети дисертаційного дослідження виділено наступні задачі:

- Провести аналіз існуючих методів кластеризації даних і визначити їхні переваги та недоліки в контексті вирішення завдань персоналізованого підходу до користувачів онлайн-платформ з ринку нерухомості.
- Розробити методику аналізу поведінкових і психографічних характеристик клієнтів з метою побудови профілю клієнта та визначення рівня його задоволеності.
- Здійснити порівняльний аналіз різних методів кластеризації різнотипових даних на прикладі користувачів, зацікавлених в нерухомості.
- Модифікувати метод кластеризації різнотипових даних, який би враховував структуровані та напівструктуровані дані за допомогою поділу на пакети та застосування перцентилів.
- Розробити алгоритм класифікації профілів клієнтів, який би на першому етапі враховував зважування відгуків, а на другому – використовував кластеризацію для формування профілів клієнтів.
- Розробити архітектуру інформаційної системи з урахуванням можливостей зменшення вартості розгортання системи за допомогою безсерверної архітектури.

1.6. Висновки до розділу 1

У даному розділі проаналізовано алгоритми опрацювання різнотипових даних, таких як як K-середніх, DBSCAN, ієрархічна кластеризація, та нечітка кластеризація.

Виявлено ряд викликів і обмежень, які включають питання масштабування, інтерпретації, універсальності та адаптивності до змінюваних умов та структур даних.

Також було виявлено, що не всі системи, що зараз є на ринку, використовують інформацію з соціальних мереж або інших неструктурованих чи напівструктурованих джерел даних для покращення своїх служб.

Це відкриває можливості для подальшого розширення та удосконалення маркетингових підходів. Базуючись на цих спостереженнях, можна стверджувати, що потрібно проводити експерименти з метою розробки більш ефективних та гнучких методів кластеризації для різнотипових даних. Це не тільки зможе дозволити системам створювати більш точні та інформативні профілі користувачів, але й привести до відкриття нових можливостей у використанні цих даних для бізнес-аналітики, прогнозування та інших цілей.

РОЗДІЛ 2. МАТЕМАТИЧНІ МЕТОДИ КЛАСТЕРИЗАЦІЇ РІЗНОТИПОВИХ ДАНИХ ТА АЛГОРИТМІВ ОПРАЦЮВАННЯ РІЗНОТИПОВИХ ДАНИХ

У розділі проведено аналіз методів кластеризації даних, описано формальне визначення профілів користувачів, описано процес класифікації профілів користувачів, проведено визначення рівня задоволеності користувача.

Результати розділу опубліковано у працях автора [98, 101]

2.1. Аналіз методів кластеризації даних

Кластеризація з використанням середньої зв'язності (Average-Linkage Clustering) є одним із методів ієрархічної кластеризації, який використовується для групування об'єктів на основі їх схожості. Цей метод працює в два етапи: обчислення матриці схожості між кожною парою об'єктів і об'єднання найближчих кластерів у кожній ітерації [45].

Алгоритм кластеризації з використанням середньої зв'язності починається з того, що кожен об'єкт у вихідному наборі даних розглядається як окремий кластер. Спочатку обчислюється матриця схожості (або матриця відстаней) між кожною парою кластерів. Для обчислення схожості між кластерами використовуються різні метри, такі як відстань Евкліда, косинусна схожість або кореляція [47].

У кожній ітерації алгоритм зливає два найближчі кластери, тобто ті, які мають найвищий ступінь зближення або найменшу відстань між ними. Цей процес триває до тих пір, поки всі об'єкти не будуть об'єднані в один великий кластер або до досягнення певного критерію зупинки.

Основна ідея методу полягає у використанні середнього значення схожості (або відстані) між об'єктами у двох кластерах для визначення ступеня зближення цих кластерів. Таким чином, кластери, які мають більше подібних об'єктів, будуть злиті раніше, оскільки у них вищий ступінь зближення. Однією з особливостей кластеризації з використанням середньої зв'язності є чутливість

до монотонного перетворення неподібностей. Це означає, що якщо змінити шкалу або застосувати монотонну функцію до матриці схожості, то це може вплинути на кінцеві результати кластеризації [48].

Щодо складності алгоритму, часова складність кластеризації за середнім зв'язком є повільнішою, ніж у методів з одинарним або повним зв'язком, оскільки обчислюється середня відстань між кожною парою об'єктів із різних кластерів. Ця процедура вимагає значно більше обчислень, ніж обчислення мінімальної (одинарний зв'язок) або максимальної (повний зв'язок) відстані між об'єктами у кластерах. Також, складність обчислення є квадратичною, оскільки потрібно зберігати матрицю схожості між всіма парами кластерів.

Кластеризація з використанням середньої зв'язності також має налаштовувані параметри, такі як кількість груп або висота дендрограми, за допомогою яких можна контролювати кількість утворених кластерів.

Зв'язок між кластерами в кластеризації з використанням середньої зв'язності можна виразити за допомогою формул:

- Обчислення відстані (дисперсії) між кластерами: Для кожної пари кластерів A і B, вираховується їхня відстань або дисперсія (cohesion) за допомогою середньої відстані між об'єктами у кожному з кластерів:

$$cohesion(A,\ B)\ = \frac{1}{(|A| * |B|)} \times \sum dist(a, b)$$

де $|A|$ і $|B|$ – кількість об'єктів у кластерах A і B відповідно;
$dist(a, b)$ – відстань між об'єктом a з кластера A та об'єктом b з кластера B;

$\sum$ - сума по всіх парам об'єктів з кластерів A і B.

- Обчислення матриці схожості (або відстаней) між кластерами: Для кожної пари кластерів обчислюється їхня схожість або відстань, що використовується для визначення найближчих кластерів. Матриця

схожості може бути представлена у вигляді квадратної матриці, де кожен елемент представляє схожість або відстань між кластерами.

- Об'єднання найближчих кластерів: У кожній ітерації алгоритм знаходить пару кластерів з найменшою відстанню або найвищим ступенем зближення і об'єднує їх у новий кластер.

Кластеризація з повним зв'язком (Complete-Linkage Clustering) є методом ієрархічної кластеризації, який використовується для групування об'єктів на основі їх схожості. У цьому методі кластери об'єднуються на основі максимальної відстані (або найбільш віддалених спостережень) між ними.

Алгоритм кластеризації з повним зв'язком (Complete-Linkage Clustering) починається з того, що кожен об'єкт у вихідному наборі даних розглядається як окремий кластер. Далі обчислюється матриця відстаней між кожною парою кластерів. Для обчислення відстані між кластерами використовуються різні метри, такі як відстань Евкліда, косинусна схожість або кореляція [50].

На кожній ітерації алгоритм знаходить два кластери, які мають максимальну відстань між найбільш віддаленими спостереженнями. Ці два кластери об'єднуються у новий кластер. Цей процес повторюється до тих пір, поки всі об'єкти не об'єднаються в один великий кластер або до досягнення певного критерію зупинки.

Одна з особливостей кластеризації з повним зв'язком полягає в тому, що вона уникне явища "ланцюгової зв'язності", що спостерігається у кластеризації з одинарним зв'язком. Це означає, що кластери в кластеризації з повним зв'язком мають тенденцію мати приблизно рівні діаметри, оскільки враховується найбільш віддалена відстань між спостереженнями.

Часова складність кластеризації з повним зв'язком може бути повільною, особливо при побудові матриці відстаней, особливо з високовимірними даними та великою кількістю спостережень [51].

Складність обчислення також є квадратичною, оскільки потрібно зберігати матрицю відстаней між кластерами. Кластеризація з повним зв'язком є

детермінованим методом, що означає, що результати будуть однакові для одних і тих самих вхідних даних. Крім того, метод може бути налаштований шляхом розбиття дендрограми за кількістю груп або висотою.

Кластеризація з повним зв'язком виражається наступним підходом та формулою:

- Обчислення відстані (або схожості) між кластерами: Для кожної пари кластерів A і B, обчислюється їхня відстань або схожість, використовуючи максимальну відстань між найбільш віддаленими спостереженнями:

$$distance(A, B) = max(dist(a, b))$$

де $dist(a, b)$ – відстань між найбільш віддаленими спостереженнями a і b з кластерів A і B;

max – операція вибору максимального значення.

- Обчислення матриці відстаней (або схожості) між кластерами: Для кожної пари кластерів обчислюється їхня відстань або схожість, що використовується для визначення найбільш віддалених кластерів. Матриця відстаней може бути представлена у вигляді квадратної матриці, де кожен елемент представляє відстань або схожість між кластерами.
- Об'єднання найбільш віддалених кластерів: У кожній ітерації алгоритм знаходить пару кластерів, що мають найбільшу відстань або найменшу схожість, і об'єднує їх у новий кластер.

Метод найближчого сусіда (Single-Linkage Clustering) є одним з популярних методів ієрархічної кластеризації. Він базується на ідеї об'єднання двох найближчих сусідніх кластерів на кожному кроці, що приводить до створення деревоподібної структури, відомої як дендрограма [46].

Алгоритм роботи методу наступний:

1. Створення початкових кластерів: Кожен об'єкт з початкового набору даних починає як окремий кластер.
2. Обчислення матриці відстаней: Обчислюється матриця відстаней між кожною парою кластерів. Відстань між кластерами може бути виміряна за

допомогою різних метрик, таких як Евклідова відстань, Манхеттенська відстань або кореляційна відстань.

3. Об'єднання найближчих сусідів: На кожному кроці об'єднуються два найближчі сусідні кластери на основі їх відстані. Це може бути зроблено шляхом пошуку мінімального значення відстані в матриці відстаней.
4. Оновлення матриці відстаней: Після об'єднання кластерів оновлюється матриця відстаней шляхом перерахунку відстаней між новими об'єднаними кластерами та решта кластерів.
5. Повторення кроків 3-4: Кроки об'єднання найближчих сусідів та оновлення матриці відстаней повторюються до тих пір, поки всі об'єкти не будуть об'єднані в один кластер або до досягнення заданого критерію зупинки.
6. Побудова дендрограми: Після закінчення кластеризації будується дендрограма, що візуалізує процес об'єднання кластерів у формі дерева. Дендрограма дозволяє визначити оптимальну кількість кластерів, шляхом аналізу відстаней між об'єднаними кластерами.

Особливості методу найближчого сусіда:

Чутливість до шуму: Метод найближчого сусіда може бути чутливим до викидів і шуму. Оскільки він базується на найближчих сусідствах, великі відстані між викидами та іншими кластерами можуть вплинути на процес об'єднання кластерів [49].

Проблема "припинення ланцюга": Метод найближчого сусіда може страждати від проблеми "припинення ланцюга" (chaining phenomenon). Великі кластери можуть бути розділені на декілька менших кластерів через довгі ланцюги з кількох відносно близьких об'єктів.

Часова складність: Обчислення матриці відстаней та оновлення її на кожному кроці може бути обчислювально витратним завданням, особливо при великій кількості об'єктів.

Інтерпретація дендрограми: Вибір оптимального кількості кластерів з дендрограми може бути суб'єктивним і вимагати додаткового аналізу даних.

Формули, які використовуються в методі найближчого сусіда: Обчислення відстані між двома кластерами:

$$d(C1,\ C2)\ =\ min(d(x,\ y))$$

де x – об'єкт з кластера С1;

y – об'єкт з кластера С2;

$d(x,\ y)$ – відстань між об'єктами.

Оновлення матриці відстаней: Після об'єднання двох найближчих сусідніх кластерів, оновлюється матриця відстаней, використовуючи оновлені відстані між новим об'єднаним кластером та рештою кластерів.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) є методом кластеризації, який використовує щільність даних для групування об'єктів. Він є одним з найпопулярніших методів кластеризації на основі щільності і дозволяє виявити кластери різної форми та щільності в наборі даних.

Основна ідея DBSCAN полягає в тому, що він з'єднує точки, які задовольняють критерію щільності. Кожна точка розглядається як щільна, якщо вона має достатню кількість інших точок (задану як мінімальна кількість) у заданому радіусі. Такі щільні точки утворюють ядро кластеру. Крім того, точки, які не мають достатньої кількості сусідів, вважаються шумом або окремими об'єктами [52].

DBSCAN виявляє кластери, обходячи кожну щільну точку та знаходячи всі її сусіди в межах заданого радіусу. Потім, використовуючи цей процес, він розширює кластери, знаходячи нові щільні точки, які досяжні з уже відомих кластерів. Цей процес продовжується, поки всі точки не будуть оброблені.

Одна з важливих особливостей DBSCAN є те, що він не вимагає передбачення кількості кластерів у вхідних даних. Він спроможний виявляти

кластери будь-якої форми та розміру. Крім того, DBSCAN може виділяти шумові точки, які не належать до жодного кластеру.

DBSCAN має декілька важливих параметрів, таких як радіус епсилон і мінімальна кількість сусідів. Вони визначають, як точки будуть включатись у кластери. Оптимальний вибір цих параметрів залежить від конкретної задачі та властивостей даних.

DBSCAN використовується для кластеризації на основі щільності і не використовує точні математичні формули. Однак, варто зазначити кілька ключових понять і критерії, що використовуються в алгоритмі [53]:

- Епсилон (ε): Визначає радіус, в межах якого шукаються сусіди кожної точки. Це один з параметрів алгоритму DBSCAN.
- Мінімальна кількість сусідів (MinPts): Визначає мінімальну кількість сусідів, необхідних для визнання точки щільною. Це також параметр алгоритму DBSCAN.

На основі цих понять ми можемо сформулювати критерії, використовувані в DBSCAN:

- Кортеж (P, ε)-щільності (P, ε-Density): Кажемо, що точка P є (P, ε) -щільною, якщо кількість сусідів у відстані ε від точки P більше або дорівнює MinPts.
- Пряма (P, ε)-досяжності (P, ε-Reachability): Визначається як мінімальна (P, ε)-щільність будь-якої точки Q, доступної від точки P, враховуючи найбільш щільні сусіди. Виражає ступінь досяжності точки P до точки Q.
- (P, ε)-зв'язок (P, ε-Connectivity): Якщо точка P має пряму (P, ε)-досяжність до точки Q, а точка Q є (Q, ε)-щільною, то кажуть, що точка P є (P, ε) -зв'язаною з точкою Q.

За допомогою цих понять і критеріїв DBSCAN визначає кластери, з'єднуючи точки, які є (P, ε)-зв'язаними та є достатньою щільністю. Точки, які не є достатньо щільними, вважаються шумом або окремими об'єктами.

K-Means також є одним з найпопулярніших методів кластеризації, який використовується для групування схожих об'єктів в наборі даних. Основна мета методу K-Means полягає в розділенні об'єктів на кластери таким чином, щоб об'єкти всередині одного кластера були максимально схожими між собою, а об'єкти з різних кластерів були суттєво відмінними.

Опис методу K-Means [54]:

1. Вибір кількості кластерів K: Починається з визначення кількості кластерів, яку можна задати користувачем або вибрати шляхом використання евристичних методів або аналізу даних.
2. Ініціалізація центроїдів кластерів: Початкові центроїди вибираються випадковим чином або за допомогою інших методів. Центроїди представляють центри кожного кластера.
3. Призначення об'єктів до кластерів: Кожен об'єкт призначається до найближчого центроїда на основі відстані між ними. Зазвичай використовується Евклідова відстань, але можуть використовуватись інші відстані. Об'єкт призначається до кластера з найменшою відстанню до центроїда.
4. Перерахунок центроїдів кластерів: Після призначення об'єктів до кластерів обчислюються нові центроїди для кожного кластера. Це здійснюється шляхом обчислення середнього значення всіх об'єктів, які належать до кожного кластера. Нові центроїди представляють оновлені центри кластерів.
5. Повторення кроків 3-4: Кроки призначення об'єктів до кластерів та перерахунку центроїдів виконуються ітеративно до досягнення збіжності. Збіжність може бути досягнута, коли зміни в призначеннях та центроїдах стають незначними.
6. Завершення алгоритму: Алгоритм може бути завершений, коли досягнуто збіжності або досягнуто максимальної кількості ітерацій.

Особливості методу K-Means:

- Часова складність: K-Means є відносно швидким алгоритмом. Однак, він може застрягати в локальних мінімах, тому може бути корисно виконати кілька запусків з різних початкових точок для отримання більш точних результатів.
- Складність пам'яті: Використання пам'яті K-Means має підквадратичну складність, оскільки необхідно зберігати координати центроїдів та мітки приналежності об'єктів до кластерів [55].
- Визначеність результатів: Результати K-Means залежать від початкового вибору центроїдів. Різний початковий вибір може призвести до різних результуючих кластерів.
- Налаштування параметрів: K-Means має невелику кількість гіперпараметрів, таких як кількість кластерів K. Це робить його відносно простим для налаштування та використання.

Результатом алгоритму K-Means є центри кластерів, які можуть слугувати як представники кожного кластера. Форми кластерів, отриманих за допомогою K-Means, зазвичай можуть бути лише гіперсферами, оскільки використовується Евклідова відстань для обчислення близькості між точками.

Формула K-Means включає два основних кроки: призначення та перерахунок центроїдів кластерів. Основна мета - мінімізувати суму квадратів відстаней між кожним об'єктом і його призначеним центроїдом.

Кроки алгоритму K-Means [56]:

Призначення: Кожному об'єкту призначається найближчий центроїд кластера. Відстань між об'єктом і центроїдом може бути виміряна, наприклад, за допомогою Евклідової відстані. Призначення об'єкта до кластера здійснюється за допомогою формули:

$$d(x,\ c)\ =\ min(dist(x,\ c_i))$$

де $d(x,\ c)$ – відстань між об'єктом x і центроїдом c;

$dist(x,\ c_i)$ – відстань між об'єктом x і центроїдом c_i.

Перерахунок центроїдів: Після призначення об'єктів до кластерів перераховуються нові центроїди кластерів. Центроїд обчислюється як середнє значення всіх об'єктів, що належать до кластера.

Формула для обчислення нового центроїда:

$$c = \frac{1}{|C|} \times \sum x_i$$

де c – новий центроїд кластера;

$|C|$ – кількість об'єктів у кластері;

$\sum x_i$ – сума всіх об'єктів у кластері.

Ці два кроки виконуються ітеративно до збіжності, коли зміни в призначеннях об'єктів та центроїдах стають мінімальними.

Метод K-Nearest Neighbors (KNN) є одним з найпростіших та популярних методів у машинному навчанні, особливо в задачах класифікації та регресії. KNN використовується для призначення класу або передбачення значення для нового зразка, заснованого на його "найближчих сусідах" у навчальному наборі даних.

Метод K-Nearest Neighbors працює наступним чином — спершу навчальний набір даних розбивається на вхідні ознаки та відповідні класи або цільові значення.

Далі визначається кількість найближчих сусідів, які будуть використовуватися для вирішення задачі. Це може бути задано користувачем або вибрано шляхом перехресної перевірки (cross-validation).

Потім відстань між новим зразком і всіма іншими зразками в навчальному наборі обчислюється за допомогою метрики відстані, такої як Евклідова відстань або Манхеттенська відстань [59].

Після цього вибираються K зразків з навчального набору, які мають найменшу відстань до нового зразка. Ці зразки стають "сусідами" нового зразка.

Наступні кроки застосовуються залежно від типу задачі:

- Класифікація: На основі класів (або міток) сусідів за допомогою голосування (наприклад, більшість голосів) визначається клас, до якого належить новий зразок.
- Регресія: Значення виходу для нового зразка обчислюється, наприклад, шляхом усереднення значень вихідних атрибутів сусідів.

Надалі виконується оцінка ефективності моделі KNN, наприклад, за допомогою метрик точності, або середньої квадратичної помилки. Параметри, такі як кількість сусідів K або використана метрика відстані, можуть бути налаштовані для досягнення кращої продуктивності [61].

KNN є непараметричним методом, оскільки не вимагає припущень про розподіл даних або параметричні моделі. Він може працювати добре в різноманітних ситуаціях, включаючи нелінійні та нерегулярні відносини між ознаками та вихідними значеннями. Окрім цього, він може бути чутливим до шкал даних, оскільки відстань між зразками обчислюється безпосередньо на основі значень ознак. Перед застосуванням KNN рекомендується провести нормалізацію або стандартизацію даних для забезпечення рівної ваги всіх ознак.

Коли кількість зразків у навчальному наборі стає дуже великою, пошук найближчих сусідів може стати обчислювально витратним завданням. Існують ефективні алгоритми, такі як дерева KD-дерева або шейдерні структури даних, які можуть прискорити цей процес.

Вибір оптимального значення K є важливим завданням. Малий значення K може призводити до нестабільності моделі і підвергати її впливу шуму, тоді як велике значення K може згладжувати границі між класами і знижувати чутливість моделі до локальних відносин в даних.

Формули, що використовуються в методі KNN:

1. Обчислення відстані між двома зразками [60]:
    - Евклідова відстань:

$$d(x, y) = \sqrt{(\Sigma(x_i - y_i)^2)}$$

де x і y – вектори ознак двох зразків;

x_i, y_i – відповідні ознаки.

- Мангеттенська відстань:

$$d(x, y) = \Sigma|x_i - y_i|$$

де x і y – вектори ознак двох зразків;

x_i, y_i – відповідні ознаки. Інші метрики відстані можуть також використовуватися в залежності від вимог задачі.

2. Вибір K найближчих сусідів: Знаходиться K зразків з навчального набору, які мають найменшу відстань до нового зразка. Ці зразки стають "сусідами" нового зразка.
3. Класифікація На основі класів (або міток) сусідів визначається клас, до якого належить новий зразок. Клас, який отримує більше голосів серед K сусідів, вважається передбаченим класом для нового зразка.
4. Регресія: Значення виходу для нового зразка обчислюється шляхом обчислення середнього значення вихідних атрибутів K сусідів.

2.2. Формальне визначення профілю користувача

Профіль клієнта - це структурована інформація, що відображає сукупність характеристик, уподобань, потреб, поведінки та демографічних даних конкретного клієнта, зібрана та аналізована з метою встановлення його типології, групування з іншими клієнтами з подібними характеристиками та визначення специфічних чинників, які впливають на його участь у ринку нерухомості. Отримання профілю клієнта включає процес збору та аналізу різноманітних даних про клієнта, таких як демографічні дані, фінансові показники, історія покупок, інформація про його нерухомість та уподобання. Одним із досліджень, що підтверджують важливість профілювання користувачів, було проведено авторами[38], які стверджують, що профілювання

може покращити якість рекомендацій у сфері нерухомості, використовуючи алгоритми машинного навчання.

Профіль клієнта S характеризується множиною ознак, які включають демографічні (S_{dm}), поведінкові (S_{bh}), психографічні (S_{psch}), мотиваційні (S_{mtv}) та знаннєво-інформаційні критерії (S_{knwlg})[62]:

$$S = S_{dm} \cap S_{bh} \cap S_{psch} \cap S_{mtv} \cap S_{knwlg}$$

Демографічні характеристики S_{dm} включають основні атрибути користувачів:

$$S_{dm} = \langle c \mid c \in (D_1, D_2, \ldots, D_n) \rangle$$

де c представляє окремого користувача, для якого визначаються його характеристики та приналежність до демографічної підмножини, D_i представляють усі можливі демографічні підмножини або категорії, до яких може належати користувач c.

Їх можна розділити на числові, категоріальні та текстові типи даних:

1. Числові характеристики:
    - Вік: Вік клієнта можна виміряти в роках.
    - Дохід: Дохід клієнта можна виміряти у валютних одиницях.
2. Категоріальні характеристики:
    - Стать: Клієнт може бути чоловіком або жінкою.
    - Рівень освіти: Клієнт може мати різний рівень освіти, такий як середня освіта, вища освіта, кандидат наук і т.д.
    - Професія: Клієнт може належати до різних професійних груп.
    - Сімейний стан: Клієнт може бути неодруженим, одруженим, розлученим і т.д.
3. Текстові характеристики:
    - Географічне розташування: Місце проживання клієнта може включати країну, регіон, місто, поштовий індекс і т.д.

Ці критерії допомагають визначити основні групи клієнтів та розуміти їхні потреби та попит на ринку нерухомості. Наприклад, молоді сім'ї можуть бути зацікавлені в придбанні першого житла, тоді як люди похилого віку можуть шукати нерухомість для пенсійного забезпечення. У дослідженні [37], демографічні дані використовувалися для виявлення відмінностей у вподобаннях різних груп користувачів щодо нерухомості.

Поведінкові характеристики S_{bh} відображають звички, ставлення до ризику, інвестиційні стратегії та інші дії клієнтів у контексті нерухомості. Ці критерії включають частоту та тривалість інвестицій, вибір різних типів нерухомості (комерційна, житлова, орендна тощо), ставлення до ризику та прийняття рішень щодо купівлі або продажу нерухомості. У роботі [39] було використано поведінкові характеристики для виявлення уподобань користувачів щодо місця проживання.

Поведінкові характеристики відображаються як

$$S_{bh} = \langle c \mid c \in (c \mid c \in B_1,\ B_2,\ \ldots,\ B_n) \rangle$$

де B_i представляють підмножини або категорії, які описують поведінкові характеристики клієнта в контексті нерухомості. Кожна з цих підмножин включає різні параметри або критерії, які використовуються для класифікації клієнтів з погляду їх поведінки та підходу до інвестування в нерухомість.

1. Числові характеристики:
    - Частота і тривалість інвестицій в нерухомість.
    - Кількість здійснених інвестицій.
    - Сума інвестицій в нерухомість за певний період часу.
    - Кількість відвідувань веб-сайту за день.
    - Загальний час, проведений на сайті.
2. Категоріальні:
    - Вибір типу нерухомості (комерційна, житлова, орендна).
    - Типи переглянутих оголошень (житлова, комерційна, орендна).

- Тип інвестицій (короткострокові, довгострокові).

3. Ординальні:
   - Ставлення до ризику (низьке, середнє, високе).

Психографічні характеристики S_{psch} відображають особистісні уподобання, цінності, інтереси, життєві пріоритети та ставлення до інвестицій у нерухомість. Ці критерії включають схильність до ризику, інвестиційні цілі, особисті інтереси (наприклад, екологічна сталість, інновації), сприйняття коливань на ринку нерухомості та ставлення до нерухомості як до активу. Робота [40] показує, як психографічні характеристики можуть бути використані для аналізу поведінки користувачів на веб-сайтах нерухомості.

Їх можна відобразити як

$$S_{psch} = \langle c \mid c \in (P_1, P_2, \ldots, P_n) \rangle$$

Де P_i представляють підмножини або категорії, що описують психографічні характеристики клієнта в контексті нерухомості. Кожна з цих підмножин включає різні параметри або критерії, які використовуються для класифікації клієнтів з погляду їх психологічних та особистісних характеристик.

1. Числові характеристики:
   - Рівень небезпеки: готовність клієнта ризикувати в інвестиціях.
2. Категоріальні:
   - Основні цінності (екологічність, сталість, інновації).
3. Ординальні:
   - Рівень значимості кожної цінності для клієнта (низький, середній, високий).

Мотиваційні характеристики S_{mtv} відображають основні мотиви та цілі, які будують інвестиційні рішення клієнтів у сфері нерухомості. Ці критерії включають фінансові цілі (наприклад, отримання прибутку, забезпечення

пенсійного забезпечення), стиль життя, соціальний статус, спадщина та податкове планування.

$$S_{mtv} = \langle c \mid c \in (M_1, M_2, \ldots, M_i) \rangle$$

Де M_i представляють підмножини або категорії, що описують психографічні характеристики клієнта в контексті нерухомості.

1. Числові:
    - Очікувані річні доходи від інвестицій.
    - Очікувана ставка повернення.
2. Категоріальні:
    - Стиль життя, який клієнт намагається підтримати (бізнес, екологічний, мінімаліст і тд.).
    - Мета інвестицій (забезпечення пенсії, отримання прибутку, забезпечення стабільності).

Знаннєво-інформаційні критерії S_{knwlg} відображають рівень знань, усвідомлення ринкових тенденцій, джерела інформації та розуміння інвестиційної термінології клієнтами. Критерії знань можуть включати рівні експертизи в галузі нерухомості, аналіз ринкових тенденцій, джерела отримання інформації (наприклад, консультанти, друковані видання, Інтернет) та розуміння термінології, пов'язаної з інвестиціями в нерухомість.

$$S_{knwlg} = \langle c \mid c \in (K_1, K_2, \ldots, K_i) \rangle$$

K_i представляють підмножини або категорії, що описують критерії знань та інформаційних характеристик клієнта в контексті нерухомості. Кожна з цих підмножин включає різні параметри або критерії, які використовуються для класифікації клієнтів з погляду їх рівня знань та усвідомлення інформації.

1. Ординальні характеристики:
    - Рівень обізнаності: може бути виміряний за допомогою тестів на знання про нерухомість або інвестиції.

2. Категоріальні:
    - Основні джерела інформації (інтернет, журнали, консультанти).

2.3. Класифікація профілів користувачів

Концепція профілів користувачів є невід'ємною частиною маркетингу. З часом еволюція методів збору та аналізу даних наділила організації величезними масивами даних користувачів. Однак, станом на сьогодні, просте накопичення даних вже є недостатнім. Для формування корисної інформації потрібна структура та класифікація.

Власне, класифікація профілів користувача дозволяє сформувати корисну інформацію. Це методологічна категоризація користувачів на основі певних атрибутів, звичок, уподобань або поведінки. Такі класифікації дозволяють компаніям надавати більш цільові продукти, послуги та комунікації конкретним групам користувачів. Вони також сприяють кращому розумінню різних нюансів, які відрізняють одну групу користувачів від іншої.

Нехай U – це універсальна множина, що представляє всіх користувачів у наборі даних. Профіль, представлений множиною P, є підмножиною U таким чином, що $P \subseteq U$. Кожен елемент x в P задовольняє певні умови або атрибути, що визначають цей профіль.

Процес профілювання виглядає так:

- Визначення атрибутів:

- Для користувача c в U визначаються відповідні атрибути, такі як поведінка при при пошуку, демографічні дані, звички перегляду тощо.

- Атрибути позначаються як A_1, A_2, A_3, ..., A_n.

- Сегментація на основі атрибутів:

- Для кожного атрибута A_1 визначається предикат $P(A_1)$, який є умовою на основі значень атрибутів. Наприклад, $P(A_1)$ може позначати клієнтів віком 18-25 років.

- Для сегментації клієнтів на основі цих предикатів, проводиться кластеризація даних, отримуючи відмінні множини профілів P_1, P_2, P_3, ..., P_m.

- Перетини та об'єднання:

- Проводиться дослідження перетинів (спільних рис), або об'єднання (комбінації) цих множин, щоб виявити спільні, або комбіновані профілі. Наприклад, $P_1 \cap P_2$ може представляти молодих користувачів (з P_1), які також часто орендують дорогі помешкання (з P_2).

- Ці похідні множини можуть бути використані для визначення багатогранних профілів.

- Доповнюючі множини:

- Визначажться доповнення кожного набору профілів P_i щодо U, представлене як $P_i^{'}$. Це відображає клієнтів, які не вписуються в певний профіль. Важливо враховувати викиди або визначати потенційно нові профілі.

- Кінцева класифікація:

- Проводиться класифікація кожного користувача c в U на основі їхньої приналежності до множин профілів P_1, P_2, P_3, ..., P_m. Користувач може належати до кількох множин профілів, що вказує на багатовимірні характеристики.

Групування та класифікація профілів користувачів дозволяє систематизувати інформацію та зробити її більш корисною. Кожний профіль можна розглядати як підмножину більшого універсуму користувачів, що полегшує розуміння унікальних характеристик та викидів.

2.4. Визначення зв’язків між критеріями профілю користувача

2.4.1. Визначення оцінки рівня задоволеності клієнта

Оцінка задоволеності клієнта є важливим концептом у сфері нерухомості, оскільки вона дозволяє оцінити ступінь задоволення та задоволення клієнтів

щодо послуг, пов'язаних з купівлею, продажем або орендою нерухомості. Це важливий показник, який впливає на репутацію та успіх бізнесу в цій галузі [63].

Оцінка задоволеності клієнта може бути визначена як міра того, наскільки клієнти задоволені якістю послуг, спілкуванням з брокерами та агентами нерухомості, процесом транзакції, рівнем професіоналізму, цінами, умовами контракту та загальним досвідом взаємодії з компанією або агентством нерухомості[64].

Оцінка задоволеності клієнта може бути здійснена через різні методи, включаючи опитування, збір відгуків та коментарів, моніторинг соціальних мереж та використання метрик, таких як CSS (Customer Satisfaction Score)[65], NPS (Net Promoter Score)[66], CES (Customer Effort Score)[67], CSI (Customer Satisfaction Index)[68] та інших.

$$CSS = (Score_1 + Score_2 + \dots + Score_n)/n$$

Де: $Score_1$, $Score_2$, …, $Score_n$ — оцінки задоволеності, надані окремими клієнтами. n – кількість клієнтів, які надали оцінки задоволеності.

$$NPS = (\%Prom) - (\%Detractors)$$

Де: $\%Prom$: Кількість клієнтів, які надали оцінку 9 або 10 (на шкалі від 0 до 10) від загальної кількості клієнтів, виражена у відсотках.

$\%Detractors$: Кількість клієнтів, які надали оцінку від 0 до 6 (на шкалі від 0 до 10) від загальної кількості клієнтів, виражена у відсотках.

Клієнти, які надали оцінку 7-8, вважаються "пасивними" (passives) у контексті формули NPS. Вони не входять безпосередньо до складу промоутерів або детракторів, але також мають вплив на розрахунок NPS. Врахування "пасивних" клієнтів у формулі NPS дозволяє враховувати їхню нейтральну або не чітко виражену думку щодо компанії або продукту.

$$CES = (Effort_1 + Effort_2 + \dots + Effort_n) / n$$

Де: $Effort_1$, $Effort_2$, ..., $Effort_n$: Оцінки, надані клієнтами, щодо зусиль, необхідних для досягнення їхніх цілей. n: Кількість клієнтів, які надали оцінки зусиль.

$$CSI = (\frac{W_1 * Score_1 + ... + W_n * Score_n)}{W_1 + ... + W_n}$$

Де W_i - ваги, які призначаються відповідно оцінці якості обслуговування та оцінці загальної задоволеності, $Score_i$ - оцінки користувачів.

Ваги в формулі CSI відображають важливість аспектів задоволеності клієнтів. Ваги можуть бути призначені на основі різних критеріїв, таких як

Комунікації з клієнтами:

- Якість комунікації з клієнтами
- Реакція на запити та питання клієнтів
- Комунікація процесу продажу або оренди нерухомості

Якість обслуговування:

- Професійність та ввічливість персоналу
- Точність інформації про нерухомість
- Дотримання угод та термінів

Своєчасність виконання замовлень:

- Швидкість обробки та виконання документів
- Своєчасна передача ключів або доступу до нерухомості
- Виконання ремонтних робіт або покращень за запланованими термінами

Якість нерухомості:

- Стан та якість будівельних матеріалів
- Функціональність та зручність планування
- Енергоефективність та екологічність

Ці ваги можуть бути змінені або доповнені залежно від потреб та пріоритетів. Ваги слід призначати з урахуванням важливості кожного аспекту для клієнтів та стратегії розвитку.

2.4.2. Визначення сили зв’язку між об’єктами

Сила зв'язку між об'єктами є важливим параметром в численних наукових дослідженнях, особливо у таких галузях як соціологія, психологія, інформатика, та біологія. Цей підрозділ спрямований на аналіз та розуміння, яким чином можна виміряти та інтерпретувати цей зв'язок. Перш ніж визначати силу зв'язку, важливо розуміти, що саме розглядається як "об'єкт". У різних галузях науки під цим поняттям можуть приховуватися різні сутності: від атомів у фізиці до особистостей у соціології [69].

- Ініціалізація: Виберемо кількість кластерів K і початкові центроїди $(\mu_1, \mu_2, ..., \mu_k)$.
- Розрахунок матриці кореляції: Побудуємо матрицю кореляції для всіх змінних, включених у процес кластеризації. Нехай R буде матрицею кореляції розміром $N \times N$, де N - кількість змінних. $R = [r_{ij}]$, де r_{ij} представляє коефіцієнт кореляції між змінною i та змінною j.
- Ваговий коефіцієнт для кожної змінної: Встановимо ваговий коефіцієнт для кожної змінної w_i, де $i = 1, 2, ..., N$. Цей коефіцієнт відображає ступінь важливості змінної у визначенні схожості між об'єктами.
- Розрахунок схожості між об'єктами: Нехай x_1 та x_2 - об'єкти, які порівнюються. Схожість $s(x_1, x_2)$ між ними можна обчислити наступним чином: $s(x_1, x_2) = \sum_i w_i \times r_{ij}$, де $i = 1, 2, ..., N$, w_i - ваговий коефіцієнт для змінної i, r_{ij} – коефіцієнт кореляції між змінною i та змінною j, N – кількість змінних.

- Застосування модифікованого k-means: Замість стандартного розрахунку відстані між об'єктами використовуємо розраховану схожість. Приналежність до кластеру та оновлення центроїдів відбуваються з урахуванням цих схожостей.
- Повторення кроків 4 і 5 до збіжності або до досягнення максимальної кількості ітерацій.

2.5. Висновки до розділу 2

У даному розділі проаналізовано методи кластеризації даних. Проведено формальне визначення профілів користувачів, що включає демографічні, поведінкові, психографічні, мотиваційні та знаннєво-інформаційні критерії. Кожен із цих критеріїв дозволяє сформувати із наявного датасету підмножини даних, сформовані за певними ознаками. Підмножини даних, у свою чергу, складаються із різнотипових даних, таких як числові, текстові, ординальні, категоріальні та інших. Описано процес класифікації профілів користувачів на основі атрибутів користувачів. Проведено визначення рівня задоволеності користувача із допомогою метрик CSS, NPS, CES та CSI. Кожна метрика дозволяє визначити наскільки клієнти задоволені якістю послуг, спілкуванням з брокерами та агентами нерухомості, процесом транзакції, рівнем професіоналізму, цінами, умовами контракту та загальним досвідом взаємодії з компанією або агентством нерухомості і таким чином дозволяє зрозуміти потреби користувача більш точно.

РОЗДІЛ 3. РОЗРОБЛЕННЯ АЛГОРИТМІВ ОПРАЦЮВАННЯ РІЗНОТИПОВИХ ДАНИХ

У даному розділі розроблено алгоритм підготовки даних, розроблено метод кластеризації різнотипових даних, який дозволяє працювати з потоковими даними на основі поділу на пакети, проведено аналіз та порівняння методів обробки пропущених значень, проаналізовано статистичний метод перцентилів для розрахунку початкових центроїдів.

Результати розділу опубліковано у працях автора [102]

3.1. Розробка алгоритму підготовки даних

3.1.1. Алгоритм очищення даних

Алгоритми очищення та підготовки даних є ключовим кроком кластеризації даних, оскільки сприяє поліпшенню якості даних перед їх аналізом. Якість та достовірність даних мають вирішальне значення для отримання надійних та точних результатів досліджень. Очищення даних перед аналізом включає в себе виявлення та виправлення помилок, видалення аномалій та викидів, а також обробку пропущених значень [57].

Один із важливих аспектів очищення даних – обробка пропущених значень. Пропущені дані є поширеною проблемою у наборах даних і можуть виникати з різних причин, включаючи технічні помилки, відмови в зборі даних або неповну звітність. Обробка пропущених значень є важливим етапом, оскільки вони можуть впливати на аналітичні результати та точність моделей.

Інший важливий аспект – виявлення та видалення дублікатів. Дублікати - це повторювані записи в наборі даних, які можуть впроваджувати зайвість та спотворювати аналітичні результати. Виявлення та видалення дублікатів є важливим кроком для забезпечення якості даних.

Третім важливим моментом є виявлення та видалення викидів. Викиди є значеннями, які значно відрізняються від загального шаблону або очікуваного розподілу в наборі даних. Вони можуть виникати через помилки вимірювання, технічні аномалії або недостовірні дані [58].

3.1.2. Обробка пропущених значень та заповнення даних

Пропущені дані є однією з поширених проблем, з якими можна зіткнутися в наборі даних. Вони можуть виникати з різних причин, таких як технічні помилки, неповна звітність або відсутність даних в певних спостереженнях. Обробка пропущених значень є важливою складовою частиною процесу очищення даних. Завдання полягає в тому, щоб врахувати пропущені дані та заповнити їх адекватними значеннями, щоб забезпечити точність та надійність аналізу, або видалити.

Є декілька різних підходів для заповнення пропущених даних — це заповнення середнім значенням (mean imputation), заповнення на основі регресії (regression imputation), метод медіанного заповнення та інші [70]. Підхід із заповнення середнім значенням полягає в тому, що пропущені значення замінюються середнім значенням наявних даних. У свою чергу метод заповнення на основі регресії використовується, коли є залежність між пропущеними значеннями та іншими змінними в наборі даних. Він використовує регресійний аналіз для прогнозування пропущених значень на основі інших змінних (предикаторів). Натомість, медіанне заповнення використовує медіану (центральне значення) не пропущених значень даної змінної для заповнення пропущених значень [72]. Медіанне заповнення є вигідним в тих випадках, коли пропущені значення не мають вираженої залежності від інших змінних або коли викиди або екстремальні значення можуть вплинути на результуючі прогнози.

Заповнення середнім значенням не є завжди найкращим підходом для обробки даних профілів користувачів. Це зумовлено наступними чинниками [71]:

- Втрата інформації: Заповнення середнім значенням може призвести до втрати важливої інформації в даних. Кожен профіль користувача може мати свої унікальні характеристики та впливові фактори, і використання

загального середнього значення може спотворити ці індивідуальні розбіжності.

- Недостатня увага до контексту: Дані профілів користувачів, зацікавлених у нерухомості, можуть бути дуже специфічними і залежати від багатьох факторів, таких як розташування, тип нерухомості, ринкові тенденції та інше. Заповнення середнім значенням не враховує цей контекст і не враховує індивідуальні особливості кожного профілю.
- Збереження розподілу даних: У разі нерівномірного розподілу даних профілів користувачів, використання середнього значення може призвести до втрати розподілу та викривлення статистичних характеристик даних.

Можна розглянути приклад на основі профілю користувача, який може пояснити, чому заповнення середнім значенням не є завжди найкращим підходом для обробки даних. Можна припустити, що проводиться аналіз профілів користувачів (таблиця 3.1). Однією з характеристик є дохід:

Таблиця 3.1

Зріз даних користувачів

Ключ	Вік	Сімейний стан	Освіта	Дохід
1	30	Married	Higher	50000
2	45	Not Married	Secondary	35000
3	27	Not Married	Higher	n/a
4	52	Married	Secondary	60000
5	39	Married	n/a	45000

У даному прикладі, у записі з ID 3 відсутня інформація про дохід. Якщо вирішити заповнити пропущене значення середнім, то буде отримано наступний результат, зображений у таблиці 3.2.

Таблиця 3.2

Результат роботи методу заповнення середнім значенням даних користувачів

Ключ	Вік	Сімейний стан	Освіта	Дохід
1	30	Married	Higher	50000
2	45	Not Married	Secondary	35000
3	27	Not Married	Higher	48000
4	52	Married	Secondary	60000
5	39	Married	n/a	45000

Однак, заповнення середнім значенням може бути неправдивим і недоцільним в даному випадку. Вплив доходу на придбання нерухомості може бути значно відмінним для кожного користувача. Наприклад, молодша людина з вищою освітою може мати вищий дохід, ніж середнє значення, оскільки вона може займати високооплачувану роботу. Тобто, нехай Y - молода особа, H – представляє особу з вищою освітою. I – представляє особу з доходом вищим за середній. J – представляє особу, яка має високооплачувану роботу.

$$Y \wedge H \rightarrow I$$

Це правило стверджує, що якщо хтось молодший та має вищу освіту, то він може мати дохід вищий за середній.

$$I \rightarrow J$$

Це правило вказує, що якщо хтось має дохід вищий за середній, то він може мати високооплачувану роботу.

Тому заповнення пропущеного значення середнім може призвести до неточностей та викривлення результатів.

Заповнення на основі регресії може не бути вигідним для обробки даних профілів користувачів з кількох причин[73]:

- Складність моделювання: Регресійні моделі можуть бути складними в побудові і вимагати великої кількості додаткових змінних або умов для точного прогнозування доходу. Враховуючи, що даних профілів користувачів може бути обмежена кількість, побудова та застосування складних регресійних моделей може призвести до недостатньої точності та перенасиченості.
- Недостатня репрезентативність даних: Враховуючи, що дані профілів користувачів можуть бути унікальними та відображати індивідуальні характеристики, регресійна модель, побудована на загальному наборі даних, може не враховувати ці унікальності. Це може призвести до недооцінки або переоцінки прогнозованих значень доходу для окремих профілів користувачів.
- Залежність від інших змінних: Враховуючи, що дохід у нерухомості може залежати від багатьох факторів, таких як розташування, тип нерухомості, ринкові тенденції, регресійна модель може не враховувати всі ці фактори. Це може призвести до недостатньої точності та неправильних прогнозів.
- Вплив викидів та екстремальних значень: Регресійні моделі можуть бути чутливими до викидів або екстремальних значень в даних. У нерухомості можуть існувати значні розбіжності в цінах або характеристиках об'єктів, і використання регресійної моделі може призвести до нереалістичних або неточних оцінок доходу.

Регресійна модель може бути записана у наступному вигляді:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n + \epsilon$$

де β_n є регресійними коефіцієнтами, які представляють вплив кожної змінної на кількість інвестицій у облігації;

X - змінні полів (вік, освіта тощо).

є - випадкова помилка.

Для визначення регресійних коефіцієнтів в регресійній моделі використовується метод найменших квадратів. Цей метод дозволяє знайти такі значення регресійних коефіцієнтів, які мінімізують суму квадратів відхилень між спостережуваними значеннями залежної змінної та прогнозованими значеннями, отриманими з регресійної моделі.

Для простої лінійної регресії, де є одна незалежна змінна (X) та одна залежна змінна (Y), регресійний коефіцієнт можна визначити за допомогою наступних формул:

$$\beta_1 = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sum_{i=1}^{n}(X_i - \overline{X})^2}$$

Регресійний коефіцієнт β_0 (інтерсепт) можна визначити за допомогою формули:

$$\beta_0 = \overline{Y} - \beta_1 \overline{X}$$

де $\overline{Y}, \overline{X}$ середні значення залежної і незалежної змінних відповідно.

Для прикладу, можна взяти набір даних про профілі користувачів, які зацікавлені у фінансових інвестиціях. Однією з характеристик є вік користувача та їх дохід (таблиця 3.1). У даному прикладі, у записі з ID 3 відсутня інформація про дохід користувача. Побудова регресійної моделі матиме наступний вигляд:

$$Income = \beta_0 + \beta_1 \cdot Age + \beta_2 \cdot Gender + \beta_3 \cdot Education + \epsilon$$

Проводиться оцінка регресійних коефіцієнтів за допомогою методу найменших квадратів на основі наявних даних. Після обчислень отримано

наступні значення регресійних коефіцієнтів: $\beta_0 = 30000$, $\beta_1 = 500$, $\beta_2 = -5000$, $\beta_3 = 2000$.

Використовуючи отримані коефіцієнти, можемо прогнозувати значення доходу для записів з пропущеними значеннями: Для запису з віком 27, статтю жінка та вищою освітою:

$$Income = 30,000 + 500 \cdot 27 - 5,000 \cdot 1 + 2,000 \cdot 1 = 42,500$$

Результат зображений у таблиці 3.3.

Таблиця 3.3.

Результат роботи методу заповнення даних користувачів методом найменших квадратів

Вік	Стать	Освіта	Дохід
30	Female	Higher	50000
45	Male	Secondary	35000
27	Female	Higher	42500
52	Male	Secondary	60000
39	Female	n/a	45000

Проте, використання регресійного заповнення може бути недоцільним у цьому випадку. Вплив доходу на фінансові інвестиції може бути дуже індивідуальним і залежати від багатьох факторів, таких як знання ринку, фінансова цілеспрямованість, посада тощо. Враховуючи це, заповнення на основі регресії, яке використовує загальний зв'язок між віком, освітою, статтю та доходом, може призвести до не надто точних та нереалістичних прогнозів.

Щодо медіанного заповнення, то цей підхід є найбільш вигідним для обробки даних профілів користувачі, з кількох причин [74]:

- Врахування розподілу даних: Медіана є статистичною мірою, яка враховує центральну тенденцію даних, не залежить від екстремальних значень. Це дозволяє зберегти розподіл даних профілів користувачів і уникнути впливу викидів або екстремальних значень на результати.
- Резистентність до викидів: Медіана є менш чутливою до викидів або екстремальних значень, оскільки вона базується на ранжуванні значень і виборі центрального значення. Це дозволяє уникнути перекручення результатів через незвично високі, або низькі значення.
- Збереження контексту: Медіана дозволяє зберегти унікальність та розбіжності в даних профілів користувачів. Вона враховує вплив кожного спостереження у вибірці, допомагаючи зберегти індивідуальні особливості та варіабельність між профілями.
- Менше викривлення розподілу: Заповнення медіаною дозволяє зберегти розподіл даних більш точно, оскільки воно використовує центральну міру, яка розташовується посередині розподілу. Це допомагає зберегти реалістичність та точність опису даних профілів користувачів.

Формула медіани:

$$X' = Median([X_1, X_2, ..., X_n])$$

Де X′ є заповненим значенням для поля *X*;

$Median$ позначає медіану набору значень.

Медіана – це центральний елемент, який розділяє впорядковані значення на дві рівні частини, так що половина значень менше медіани, а інша половина - більше медіани. Для вирахування медіани, необхідно впорядкувати значення від найменшого до найбільшого і обрати значення, що перебуває у середині.

Формула для обчислення медіани в залежності від кількості значень *n* є наступною:

Якщо *n* непарне:

$$Median([X_1, X_2, ..., X_n]) = X_{\frac{n+1}{2}}$$

Якщо *n* парне:

$$Median([X_1, X_2, ..., X_n]) = \frac{X_{\frac{n}{2}} + X_{\frac{n}{2}+1}}{2}$$

де X_i - *i*-те значення в відсортованому наборі значень.

Отже, формула для медіанного заповнення поля X полягає в обчисленні медіани набору значень, які вже мають відомі значення, і використанні отриманої медіани як заповненого значення для пропущених записів.

Для прикладу, можна знову взяти зріз даних про профілі користувачів (таблиця 3.1). Множина значень доходу:{50000, 35000, n/a, 60000, 45000}.
Після впорядкування, значення матимуть наступний вигляд: {35000, 45000, 50000, 60000}
Медіанне значення: 50000.
Результат зображений у таблиці 3.4.

Таблиця 3.4

Результат роботи методу заповнення даних медіанним заповненням

ID	Age	Marital status	Education	Income
1	30	Married	Higher	50000
2	45	Not Married	Secondary	35000
3	27	Not Married	Higher	50000
4	52	Married	Secondary	60000
5	39	Married	n/a	45000

Заповнення медіанним значенням дозволяє зберегти характеристику центральної тенденції даних та уникнути впливу викидів чи значних відхилень на результати аналізу. В даному прикладі, медіанне заповнення дозволило зберегти відносно типове значення доходу серед користувачів, що сприяє більш реалістичному представленню даних і уникненню спотворень, які можуть виникнути при використанні середнього значення.

3.1.3. Виявлення та видалення дублікатів

Дублікати є повторюваними записами, які можуть виникати через помилки при зборі даних, повторне введення інформації або технічні проблеми. Виявлення дублікатів є важливим етапом очищення даних, оскільки вони можуть призвести до неточностей у статистичних аналізах, моделях прогнозування або базах даних, а також викривають ризик спотворення результатів дослідження або прийняття неправильних рішень[75].

Після виявлення дублікатів, наступним кроком є їх видалення. Існують різні стратегії для видалення дублікатів, включаючи збереження лише одного запису з дублікатів, об'єднання атрибутів з дублікатів у єдиний запис або позначення дублікатів спеціальним прапорцем[76].

Існують різні підходи для виявлення дублікатів, такі як метод порівняння схожості, використання хеш-функцій, алгоритми кластеризації або використання статистичних моделей. У роботі розглянуто та порівняно два поширені підходи — це метод порівняння схожості та використання хеш-функцій.

Метод порівняння схожості полягає в обчисленні схожості між двома записами на основі порівняння їх характеристик або атрибутів. Одна з формул, що часто використовується, називається коефіцієнтом схожості Жаккара і визначається наступним чином:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Де A та B - два порівнювані записи;

$|\cdot|$ – показник кількості елементів у множині. Ця формула вимірює співпадіння між елементами двох записів, де 0 означає відсутність схожості, а 1 – повну ідентичність.

Після обчислення схожості між всіма парами записів, можна встановити певний поріг схожості, який визначає, коли два записи вважатимуться дублікатами. Зазвичай використовуються порогові значення, наприклад, якщо схожість перевищує 0.9, записи вважаються дублікатами.

Для виявлення дублікатів у базі даних користувачів застосовується наступний алгоритм:

- З бази даних витягуються поля Ім’я, Прізвище, Номер телефону користувача (електронну адресу враховувати немає сенсу, оскільки вона повинна бути унікальною при реєстрації);
- Порівнюється значення Ім’я, Прізвища та номера телефону між всіма парами записів.

Якщо знайдено співпадіння, це може свідчити про наявність дублікатів. Потім можна вжити відповідних заходів, у даному випадку — це видалення дублікатів.

Цей метод дозволяє здійснювати детальне порівняння записів на основі їх конкретних атрибутів. Це дозволяє враховувати різноманітні аспекти записів та точно визначати їх ідентичність. Окрім цього метод порівняння атрибутів може забезпечити високу точність виявлення дублікатів, оскільки порівняння проводиться безпосередньо за значеннями атрибутів.

До недоліків методу можна віднести наступне:

- Обчислювальна складність: Порівняння атрибутів вимагає прямого порівняння значень кожного атрибуту між двома записами. При великій кількості записів це може стати дуже обчислювально витратною операцією.

- Затримки в часі: Чим більше атрибутів мають бути порівняні, тим більше часу потрібно для виконання порівняння між записами. У великих наборах даних це може призвести до значних затримок в обробці.
- Потреба в ресурсах: Порівняння атрибутів вимагає пам'яті для зберігання значень атрибутів кожного запису. При великих обсягах даних це може призвести до значного використання ресурсів пам'яті.
- Чутливість до помилок: Порівняння атрибутів може бути чутливим до незначних змін атрибутів, таких як розділові знаки, регістр символів тощо. Це може спричинити помилкове визначення дублікатів.

Окрім цього часова складність даного алгоритму є наступною:

$$O(N^2)$$

Метод порівняння атрибутів має лінійну часову складність, що означає, що час виконання залежить від кількості записів, що порівнюються.

Наступний розглянутий метод — це метод використання хеш-функцій. Хешування — це процес перетворення вхідних даних будь-якого розміру в фіксоване хеш-значення фіксованої довжини. Хеш-функції приймають на вхід послідовність даних будь-якого розміру і генерують унікальний вихідний код, який називається хеш-значенням[77]. Одна з найпоширеніших хеш-функцій - це SHA-256 (Secure Hash Algorithm 256-bit). Вона приймає вхідні дані будь-якого розміру і генерує хеш-значення фіксованої довжини 256 біт (32 байти). Окрім SHA-256 існує багато інших функцій, таких як MD5, SHA-512 та ін.

MD5 (Message Digest Algorithm 5): MD5 є відносно швидкою хеш-функцією, яка генерує 128-бітові хеш-значення. Однак, MD5 вважається застарілою та вразливою до колізійних атак. Колізія в MD5 означає можливість отримання двох різних вхідних повідомлень з однаковим хеш-значенням. З цієї причини MD5 не рекомендується для криптографічних застосувань [78].

SHA-256 (Secure Hash Algorithm 256-bit): SHA-256 є потужною хеш-функцією, яка генерує 256-бітові хеш-значення. Вона є стійкою до колізійних атак і часто використовується для криптографічних застосувань, таких як підписи даних та перевірка цілісності. SHA-256 є рекомендованою хеш-функцією для багатьох застосувань безпеки [79].

SHA-512 (Secure Hash Algorithm 512-bit): SHA-512 є більш потужною хеш-функцією, ніж SHA-256, і генерує 512-бітові хеш-значення. Вона також є стійкою до колізійних атак та використовується в криптографічних застосуваннях. SHA-512 може бути використаною у випадках, коли потрібна більша довжина хеш-значень або коли вимагається більша стійкість до атак [79].

Оскільки для виявлення дублікатів немає потреби створювати довгі хеш-значення — можна зосередитись на SHA-256.

Використання хеш-функцій, таких як SHA-256, для виявлення дублікатів передбачає порівняння отриманих хеш-значень. Якщо два записи мають однакове хеш-значення, це може вказувати на наявність потенційного дубліката.

Математично метод використання хеш-функцій описується так:

$$H(x) = h$$

Де x - вхідний запис;

$H(x)$ - його хеш-значення;

h - обчислене значення.

Після обчислення хеш-значень для всіх записів можна порівняти їх між собою. Якщо два записи мають однакове хеш-значення (h1 = h2), це може свідчити про наявність потенційного дубліката. В такому випадку додаткова перевірка атрибутів може бути проведена, щоб підтвердити ідентичність записів.

Цей підхід базується на тому, що хеш-функції генерують унікальне хеш-значення для кожного вхідного набору даних. Колізії, коли два різних записи мають однакове хеш-значення, є рідкісним явищем у стійких

хеш-функціях. Однак, варто зауважити, що існує теоретична можливість колізій, хоча вона дуже мала.

Робота методу використання хеш-функцій виглядає наступним чином:

- Для кожного користувача вираховується хеш-значення на основі певних атрибутів, таких як ім'я, прізвище та номер телефону. Наприклад, можна скласти рядок методом конкатенації, що містить ці атрибути, і застосувати хеш-функцію SHA-256 до цього рядка;
- Отримане хеш-значення зберігається разом з відповідними записами в базі даних;
- Перевіряється, чи існують хеш-значення, які збігаються між різними користувачами. Це може свідчити про можливі дублікати;

Виявлені дублікати видаляються з бази даних.

Наявними недоліками цього методу є:

- Колізії: Хеш-функції, особливо ті, які мають фіксований розмір вихідних даних, можуть мати колізії - ситуації, коли два різних вхідних набори даних мають однакове хеш-значення. Хоча вірогідність колізій у стійких хеш-функцій надзвичайно низька, вони все ж можливі, особливо при використанні коротких хеш-значень.
- Відсутність зворотного відновлення: Хеш-функції є односторонніми, що означає, що неможливо відновити початкові дані з хеш-значення. Це може бути недоліком, якщо потрібно отримати початкові дані з хеш-значення, наприклад, для відновлення вихідних даних з дублікатів.
- Вразливість до атак: Деякі хеш-функції можуть бути вразливими до певних атак, таких як атаки колізій або прямого вибору. Це особливо стосується застарілих або небезпечних хеш-функцій. Важливо використовувати стійкі хеш-функції, які мають доведену стійкість до різних атак.

- Великий розмір вихідних даних: Хеш-функції генерують хеш-значення фіксованого розміру, незалежно від розміру вхідних даних. Це може призвести до втрати частини інформації або збільшення розміру даних при обробці великих наборів даних.

Щодо часової складності, то вона зазвичай залежить від обсягу даних і розміру хеш-значення. При використанні хеш-функцій, таких як SHA-256, обчислення хеш-значень для N записів має лінійну часову складність *O(N)*. Порівняння хеш-значень має сталу часову складність, оскільки хеш-значення мають фіксований розмір.

Порівнявши та оцінивши роботу методів порівняння схожості та використання хеш-функцій, можна зробити висновок, що використання хеш-функцій є більш ефективним методом, оскільки:

- Хеш-функції забезпечують швидке обчислення хеш-значень для великих обсягів даних, що дозволяє ефективно виявляти дублікати в масштабі всієї бази даних.
- Використання хеш-функцій дозволяє зводити великий обсяг даних до фіксованого розміру хеш-значення, що спрощує зберігання та обробку хеш-значень.
- Хеш-функції гарантують унікальність хеш-значень, що дозволяє точно визначити наявність дублікатів.
- Хеш-функції незворотні, тобто неможливо відновити початкове повідомлення з хеш-значення, що забезпечує безпеку і конфіденційність даних.
- Використання хеш-функцій дозволяє ефективно використовувати хеш-таблиці та інші структури даних для швидкого пошуку та порівняння хеш-значень.

У порівнянні з методом порівняння схожості, який може бути часо та ресурсозатратним, використання хеш-функцій забезпечує швидше та

ефективніше виявлення дублікатів. Враховуючи ці фактори, можна стверджувати, що використання хеш-функцій є більш ефективним та прогресивним методом для виявлення дублікатів в базах даних.

3.1.4. Виявлення та видалення викидів

Викиди - це дані, які суттєво відхиляються від загального шаблону набору даних. Вони можуть вводити шум і спотворювати результати статистичного аналізу. Виявлення викидів є важливим кроком у очищенні даних.

Нормалізація даних - це процес перетворення значень ознак таким чином, щоб вони знаходилися у визначеному діапазоні або мали певні статистичні властивості. Головна мета нормалізації - забезпечити рівномірність масштабування ознак, щоб вони могли бути порівняні та оброблені належним чином [80].

Існує декілька методів нормалізації - мінімаксна нормалізація, нормалізація за максимальним абсолютним значенням, стійка нормалізація та стандартна нормалізація.

Мінімаксна нормалізація – це метод масштабування даних, який перетворює ознаки таким чином, що їх значення знаходяться в діапазоні між 0 та 1. Це один з найпопулярніших методів нормалізації, який часто використовується у передобробці даних перед застосуванням алгоритмів машинного навчання [81]. Цей метод нормалізації працює найкраще, коли розподіл даних приблизно нормальний і немає великих викидів.

Формула мінімаксної нормалізації представлена у такому вигляді:

$$x' = \frac{x - min(X)}{max(X) - min(X)}$$

Де x' – нове значення після нормалізації, $min(X)$ – мінімальне значення вибірки X, $max(X)$ – максимальне значення вибірки X.

У свою чергу, нормалізація за максимальним абсолютним значенням перетворює дані шляхом ділення кожного елемента на найбільше абсолютне

значення у вибірці [81]. Цей метод масштабування не зміщує центр розподілу даних, тому після трансформації середнє не обов'язково дорівнюватиме нулю. Основна перевага цього підходу полягає в тому, що він може працювати із розрідженими матрицями (датасетами), де більшість значень дорівнює нулю. Оскільки більшість присутніх даних мають значення відмінне від нуля – цей спосіб нормалізації не буде ефективним.

Щодо методу стійкої нормалізації – цей метод використовує медіану та міжквартильний розмах для масштабування ознак. Його недоліком є низька можливість виявлення викидів, що спричиняє зменшення точності нормалізації.

Стандартна нормалізація, або метод Z-оцінки є одним з найпоширеніших підходів до виявлення викидів в наборі даних. Цей метод використовує стандартні відхилення та середнє значення для вимірювання того, наскільки кожне значення відхиляється від очікуваного середнього [82]. Застосування стандартної нормалізації забезпечує, що розподіл ознак після масштабування буде мати форму стандартного нормального розподілу. Окрім цього, дані повинні бути приведеними до одного типу. Оскільки, у даному випадку, дані приводяться до одного типу – було вирішено використати стандартну нормалізацію.

Стандартна формула методу Z-оцінки виглядає так:

$$Z = \frac{X-\mu}{\sigma}$$

Де X - значення точки даних;

μ - середнє значення набору даних;

σ - стандартне відхилення.

Нехай існує вхідний набір даних $X = [x_1, x_2, \ldots, x_n]$, де x_i - значення поля *Income* для кожного клієнта. Знаходиться мінімальне значення *Income*: *min(X)*. Знаходиться середнє значення *Income*: *X*. Для кожного значення x_i використовується формула:.

$$x_i^{'} = \frac{x_i - \mu}{\sigma}$$

Де $x_i^{'}$ – нормалізоване значення кожного x_i-го елементу;

x_i – оригінальне значення *i*-го елементу;

μ - середнє значення набору даних;

σ - стандартне відхилення.

Значення Z-оцінки вказує на те, наскільки дане значення відхиляється від середнього значення в стандартних відхиленнях. Зазвичай використовується певне порогове значення для визначення, коли значення Z-оцінки вважається викидом. За замовчуванням, якщо значення Z-оцінки перевищує 3 або -3 (що відповідає відхиленню на 3 стандартних відхилення від середнього), це вказує на те, що дане значення вважається викидом.

Після виявлення викидів наступним кроком є видалення викидів. Простий підхід - заміна викидів відсутніми значеннями або подальше дослідження. Формула для видалення викидів може бути представлена так:

$$X' = X, \text{ якщо } |Z| \leq \text{поріг}$$

$$X' = \text{викид, якщо } |Z| > \text{поріг}$$

Де X ′ представляє очищену точку даних;

Z - Z-оцінка;

поріг - задане значення.

3.2. Розробка алгоритму препроцесингу даних та зменшення розмірності перед кластеризацією

Перед застосуванням алгоритмів кластеризації до даних, важливо виконати попередній етап підготовки даних, відомий як препроцесинг.

Препроцесинг (передобробка) даних - це процес підготовки та обробки вхідних даних перед їх подальшим аналізом або застосуванням алгоритмів машинного навчання [83]. Препроцесинг даних є важливою складовою частиною аналізу даних, оскільки він допомагає забезпечити якість, надійність та відповідність даних для подальшого використання. Препроцесинг даних включає такі кроки, як очищення даних від шуму, видалення аномалій, нормалізацію даних, а також зменшення розмірності набору даних.

Нормалізація даних - це процес приведення значень ознак до спільного масштабу або діапазону. Вона використовується для забезпечення однакової ваги або важливості різних ознак у наборі даних. Головна мета нормалізації полягає в усуненні проблем, які виникають внаслідок різних масштабів або діапазонів значень ознак.

Зменшення розмірності даних - це процес зниження кількості ознак або змінних у наборі даних, зберігаючи при цьому значущість та інформаційну вартість цих даних. Зменшення розмірності відіграє важливу роль у аналізі даних і машинному навчанні, оскільки допомагає уникнути проблем, пов'язаних з високою розмірністю даних і покращити ефективність алгоритмів обробки та моделювання.

3.2.1. Очищення даних

Першим кроком очищення даних є виявлення відсутніх значень. Нехай D буде вхідним набором даних, $D = \{x_1, x_2, x_3, \ldots, x_n\}$, де x_i представляє записи даних. Проводиться перевірка кожного поля даних x_i на наявність пропущених значень. Фіксуються пропущені значення як m_i, де $m_i = 1$, якщо значення відсутнє, $m_i = 0$, якщо значення присутнє.

Наступним кроком є видалення записів із відсутніми значеннями. Нехай M буде матрицею пропущених значень, $M = \{m_1, m_2, \ldots, m_n\}$. Видалення записів з відсутніми значеннями можна виразити як $D' = \{x_i \mid m_i = 0\}$.

Для отримання більшого уявлення про датасет, необхідно сформувати інформацію про наявність пустих значень. На рис.3.1. зображено зведену інформацію про типи даних та кількість непустих значень.

```
user_listing_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2000 entries, 1 to 2000
Data columns (total 10 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   date_of_birth           2000 non-null   int64
 1   gender                  2000 non-null   object
 2   family_status           2000 non-null   object
 3   kids_count              2000 non-null   int64
 4   social_benefits         2000 non-null   object
 5   employment_status       2000 non-null   object
 6   employed_since          2000 non-null   int64
 7   net_income              1942 non-null   float64
 8   interested_in           1876 non-null   object
 9   search_performed_count  2000 non-null   int64
dtypes: float64(1), int64(4), object(5)
memory usage: 171.9+ KB
```

Рис.3.1. Зведена інформація про дані

З наведеного вище огляду можна зробити висновок, що пусті поля присутні у записах. Ці записи необхідно видалити.

3.2.2. Відбір ознак

Відбір ознак - це процес вибору підмножини ознак з вхідного набору даних, що мають найбільший вплив на досягнення певної метрики якості моделі або аналітичних результатів.

Нехай існує вхідний набір даних $X = [x_1, x_2, ..., x_n]$, де кожний x_i є вектором ознак розмірності m ($x_i = [x_{i1}, x_{i2}, ..., x_{in}]$). Також існує цільова змінна $Y = [y_1, y_2, ..., y_n]$, де y_i є цільовим значенням, яке потрібно прогнозувати, або аналізувати. Формально, відбір ознак можна визначити як оптимізаційну задачу, де шукається підмножина ознак $S \subseteq \{1, 2, ..., m\}$, що максимізує або мінімізує певну метрику якості $J(S)$ на основі заданої функції ваги $W(S)$, обмежень і контексту задачі: $max\ J(S) * W(S)$, або $min\ J(S) * W(S)$, де $J(S)$ - функція метрики якості, що оцінює ефективність моделі або аналітичних результатів з використанням підмножини ознак S, $W(S)$ - функція ваги, яка

призначає вагу кожній ознаці в підмножині S, обмеження можуть включати кількість обраних ознак, обмеження на розмірність ознак, обмеження на використання певних груп ознак тощо.

Процес відбору ознак можна провести на основі матриці кореляції. Алгоритм дій буде наступний:

- Обчислення матриці кореляції: Обчислюється матриця кореляції, яка відображає коефіцієнти кореляції між кожною парою ознак. Матриця кореляції C розмірності $m \times m$, де C_i□ - коефіцієнт кореляції між ознаками x_i та x□. Формула для обчислення коефіцієнта кореляції Пірсона між ознаками x_i та x□:

  $r(x_i, x□) = (\Sigma((x_i□ - \bar{y}_i)(x□□ - \bar{y}□))) / \sqrt{((\Sigma(x_i□ - \bar{y}_i)^2)(\Sigma(x□□ - \bar{y}□)^2))}$

  де x_i□ - значення ознаки x_i для об'єкту j,

  $\bar{y}_i$ - середнє значення ознаки x_i,

  x□□ - значення ознаки x□ для об'єкту j,

  ȳ□ - середнє значення ознаки x□.
- Вибір ознак з високою кореляцією: Ознаки, для яких абсолютне значення кореляції є великим, вважаються інформативними і можуть бути включені до моделі. Поріг кореляції може бути встановлений на основі емпіричного досвіду або застосування певної критичної межі. Наприклад, можна вибрати ознаки, для яких $|r(x_i, y)| > 0.5$ вважатимуться високо корельованими.
- Видалення взаємно корельованих ознак: Якщо дві або більше ознаки мають високу кореляцію між собою, одна з них може бути вилучена, оскільки вона не додає додаткової інформації до моделі. Формула для обчислення коефіцієнта кореляції між ознаками x_i та x□ допомагає виявити взаємно корельовані ознаки. Якщо $|r(x_i, x□)| > порогу\ кореляції$, одну з ознак можна виключити.
- Оцінка впливу видалених ознак: Після видалення ознак можна оцінити вплив цього процесу на якість моделі або аналітичні

результати. Виконуються перевірки на покращення продуктивності моделі або зниження перевантаження внаслідок відсутності непотрібних ознак.

3.2.3 Зменшення розмірності даних

Зменшення розмірності (Dimensionality Reduction) є процесом зменшення кількості ознак у вхідному наборі даних, зберігаючи при цьому якнайбільше значення та інформацію, що міститься в даних. Це дозволяє скоротити обсяг даних та покращити ефективність обробки та аналізу.

Зменшення розмірності можна розглядати як процес проектування нового простору ознак, в якому враховується важлива інформація з вихідного простору. Це може бути досягнуто шляхом обчислення нових ознак, які є комбінацією вихідних ознак або шляхом видалення незначних ознак.

Основними методами зменшення розмірності даних є метод головних компонент (PCA), аналіз дискримінантних ознак (Discriminant Feature Analysis, DFA), відбір ознак на основі статистичних тестів, метод розкладу на сингулярні значення (Singular Value Decomposition, SVD), алгоритми вбудовані в моделі (Embedded Methods) та ін..

PCA є статистичним методом, який використовується для знаходження нових ознак (головних компонент) шляхом перетворення вихідних ознак у такий спосіб, щоб вони були лінійно некорельовані та максимально дисперсійні. Головні компоненти відповідають напрямам з найбільшою дисперсією в даних [84].

DFA є методом, який спрямований на знаходження ознак, які найбільше впливають на розрізнення між класами або категоріями. Цей метод може бути використаний, якщо необхідно знайти найбільш інформативні ознаки, що відрізняють різні групи клієнтів у вашій базі даних [85].

Використання статистичних тестів, таких як аналіз дисперсії (ANOVA) або t-тест, може допомогти вибрати ознаки, які мають статистично значущий вплив на цільову змінну або певні групи даних.

SVD є методом математичного аналізу, який здійснює розклад матриці даних на три компоненти: матрицю лівих сингулярних векторів, діагональну матрицю сингулярних значень та матрицю правих сингулярних векторів. Застосування SVD дозволяє вибрати головні сингулярні значення та відповідні власні вектори для зменшення розмірності даних [86].

Деякі моделі машинного навчання, такі як регуляризована логістична регресія та дерева рішень, можуть автоматично виконувати відбір ознак в процесі навчання. Це означає, що модель сама вирішує, які ознаки є найбільш інформативними для досягнення кращої прогностичної точності.

Для роботи із базою даних користувачів, зацікавлених у ринку нерухомості метод PCA буде достатньо ефективним, оскільки він дозволяє зменшити розмірність даних шляхом проекції їх на головні компоненти, що знаходяться у напрямках максимальної дисперсії. Це дозволяє зберегти більшу частину інформації, маючи меншу кількість ознак. Окрім цього метод гарантує, що що кожна головна компонента містить унікальну інформацію, не дублюючи інші компоненти. Отже, для зменшення розмірності обрано PCA метод.

Нехай існує вхідний набір даних X, який складається з n об'єктів, кожен з яких має m ознак. Матриця X має розмірність n × m, де кожен рядок представляє один об'єкт, а кожний стовпець відповідає одній ознаці.

Спочатку, виконується стандартизація даних шляхом віднімання середнього значення кожної ознаки та поділу на стандартне відхилення. Це допомагає зберегти відносну вагу ознак та уникнути впливу різних масштабів.

Потім, обчислюється коваріаційна матриця C розмірності m × m, де кожний елемент $C_{i□}$ представляє коваріацію між ознаками i та j. Формула для обчислення коваріації між ознаками x_i та $x□$: $C_{i□} = (1 / (n - 1)) * Σ((x_i - ȳ_i)(x□ - ȳ□))$, де x_i та $x□$ - значення ознаки i та j відповідно, $ȳ_i$ та $ȳ□$ - середнє значення ознаки i та j відповідно, $Σ$ - сума по всіх об'єктах.

Далі визначаються головні компоненти шляхом розкладу коваріаційної матриці. Головні компоненти - це нові ознаки, які є лінійними комбінаціями

вихідних ознак. Перша головна компонента (PC1) має найбільшу дисперсію, друга головна компонента (PC2) має другу за величиною дисперсію, і так далі. Головні компоненти можна обчислити шляхом розкладу коваріаційної матриці C на власні вектори та сингулярні значення. Власні вектори відповідають головним компонентам, а сингулярні значення вказують на дисперсію, пов'язану з кожною головною компонентою.

Головні компоненти можна відсортувати за спаданням дисперсії (сингулярних значень) і вибрати перші k компонент, які зберігають більшість варіації в даних. Вибір кількості головних компонент залежить від власної дослідницької мети та критеріїв збереження варіації.

Дані можна проекціювати на новий підпростір, створений з вибраних головних компонент. Це досягається шляхом множення матриці вихідних даних X на матрицю головних компонент, яка складається з власних векторів.

3.2.4. Виділення нових ознак

Виділення нових ознак - це процес створення додаткових характеристик або ознак на основі наявних даних з метою збагачення інформації та покращення розуміння даних. Цей процес може включати комбінування, перетворення, агрегацію або розширення існуючих ознак, а також використання додаткових знань або алгоритмів для створення нових ознак, які можуть виявити патерни, взаємозв'язки або важливі аспекти даних. В результаті виділення нових ознак можна отримати більш повну та репрезентативну картину про досліджувані дані та використовувати їх для подальшого аналізу, моделювання чи прийняття рішень.

База даних користувачів, зацікавлених у сфері нерухомості містить великий набір полів, таких як дата народження, дохід, сімейний статус, кількість дітей, оцінки рівня задоволення клієнтів, статус зайнятості, кількість пошукових запитів, зацікавленість у певному виді нерухомості тощо. На основі цих полів можна створити нові ознаки:

- Вікові групи: Створюється нова ознака, яка відображатиме вікові групи клієнтів на основі їх дати народження. Наприклад, можна створити категорії, такі як "молоді" (18-30 років), "дорослі" (31-50 років) та "літні" (51+ років). Це дозволить враховувати віковий фактор у подальшому аналізі.
- Сімейний статус та кількість дітей: Можна поєднати сімейний статус та кількість дітей, щоб створити нову ознаку, яка відображатиме наявність сім'ї та її розмір. Наприклад, можна створити категорії, такі як "одинокий", "одружений без дітей", "одружений з однією дитиною" і т.д. Це дозволить розглядати сімейний статус та кількість дітей як одну змінну.
- Індекс задоволеності клієнтів: Є можливість об'єднати різні оцінки задоволеності клієнтів (Customer Satisfaction Score, Net Promoter Score, Customer Effort Score) у нову ознаку - індекс задоволеності клієнтів. Це може бути середнє значення або взважене комбінування цих оцінок. Це дозволить враховувати загальний рівень задоволеності клієнтів у подальшому аналізі.
- Соціальні пільги та статус зайнятості: Створюється нова ознака, яка відображатиме наявність соціальних пільг у залежності від статусу зайнятості клієнта. Наприклад, можна створити бінарну ознаку, яка вказуватиме, чи має клієнт соціальні пільги, основуючись на їх статусі зайнятості та наявності соціальних пільг у полях бази даних.
- Інтереси та діяльність клієнтів: Створюються нові ознаки, які відображатимуть зацікавленість клієнтів у конкретних видів інвестицій, продажу чи оренді нерухомості. Наприклад, можна створити бінарні ознаки, які вказують, чи цікавить клієнта інвестиції, продаж або оренда нерухомості.

- Стаж роботи: Можна обчислити стаж роботи клієнтів, віднімаючи рік зайнятості (employed_since) від поточного року. Це дасть нову ознаку, яка відображає тривалість робочого досвіду клієнта.
- Відносний дохід: Обчислюється відношення чистого доходу (net_income) до середнього доходу в базі даних. Це дозволить отримати нову ознаку, яка відображає, наскільки клієнт знаходиться вище або нижче від середнього рівня доходу.
- Активність пошуку: Можна обчислити відношення кількості пошукових запитів (search_performed_count) до загальної кількості клієнтів. Це дозволить створити нову ознаку, яка відображатиме активність клієнтів у пошуку інформації про інвестиції, продаж чи оренду нерухомості.
- Загальний індекс задоволеності: Можна обчислити загальний індекс задоволеності, який буде складатися з комбінації різних показників задоволеності, таких як Customer Satisfaction Score, Net Promoter Score, Customer Effort Score та Customer Satisfaction Index. Це надасть нову ознаку, яка узагальнює загальний рівень задоволеності клієнтів.
- Відносна зацікавленість: Обчислюється відсоток клієнтів, зацікавлених у конкретних видів інвестицій, продажу чи оренді нерухомості. Наприклад, можна обчислити відношення кількості клієнтів, зацікавлених у продажу, до загальної кількості клієнтів. Це надасть нові ознаки, які відображають відносну зацікавленість клієнтів.

На рис. 3.2. зображено створення нових ознак та приведення різних типів даних до числових.

```
#Feature for deriving gender
user_listing_data["Gender"] = user_listing_data["gender"].replace({"male": 1, "female":2})
#Feature for deriving employment status
user_listing_data["Employment_Status"] = user_listing_data["employment_status"].replace({"Employed": 1, "Unemployed":2})
#Feature for deriving customer interests
user_listing_data["Interested_In"] = user_listing_data["interested_in"].replace({"Rent": 1, "Buy":2})
#Feature for deriving employed since status
user_listing_data["Employed_For"] = 2023 - user_listing_data["employed_since"]
user_listing_data["Employed_For"] = user_listing_data["Employed_For"][user_listing_data["Employed_For"] >= 0]
user_listing_data = user_listing_data.dropna()
#Feature for deriving customers age
user_listing_data["Age"] = 2023 - user_listing_data["date_of_birth"]
#Feature for deriving family size
data["family_status"]=data["family_status"].replace({
    "Married":"Married",
    "Together":"Married",
    "Widow":"Single",
    "Divorced":"Single",
    "Single":"Single",})
user_listing_data["Family_Size"] = user_listing_data["family_status"].replace({"Single": 1, "Married":2}) + user_listing_data["kids_count"]
#Feature pertaining parenthood
user_listing_data["Is_Parent"] = np.where(user_listing_data.kids_count> 0, 1, 0)
# For clarity
user_listing_data=user_listing_data.rename(columns={
    "net_income": "Income",
    "search_performed_count":"Search_Activity"
})
#Dropping some of the redundant features
to_drop = [
    "gender",
    "interested_in",
    "date_of_birth",
    "social_benefits",
    "employed_since",
    "employment_status",
]
user_listing_data = user_listing_data.drop(to_drop, axis=1)
user_listing_data.describe()
```

Рис.3.2. створення нових ознак та приведення різних типів даних до числових

На основі даних можна проаналізувати та створити безліч ознак, проте це необхідно робити на основі експертних знань, попередніх досліджень, аналізувати їх та перевіряти їх ефективність. В разі, якщо ознака не є ефективною – її можна видалити. Для того, щоб оцінити ефективність ознаки, можна створити графік з обраних підмножин ознак.

Графік дозволить візуально проаналізувати наявні дані та провести подальше коригування датасету. Із графіка (рис.3.3.) видно, що такі дані як стать та зайнятість (працевлаштований чи безробітний) не несуть корисної інформації, тому, при побудові кластерів, вони будуть видалені з загального обсягу інформації.

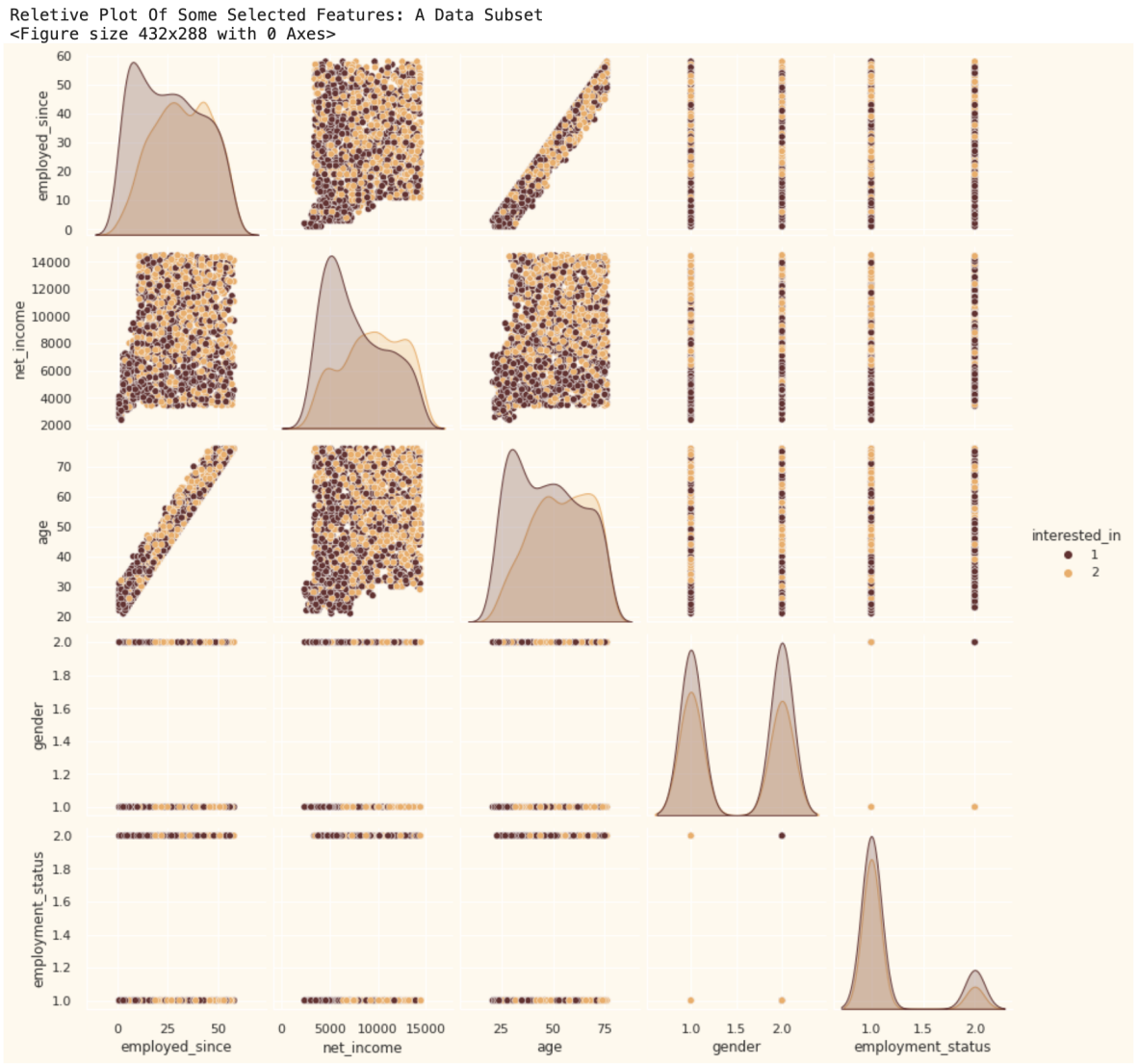

Рис.3.3. Графічне відображення підмножини ознак

Застосувавши вище описаний метод кореляції, можна перевірити чистоту даних та переконатись чи можна користуватись новими ознаками (рис 3.4.)

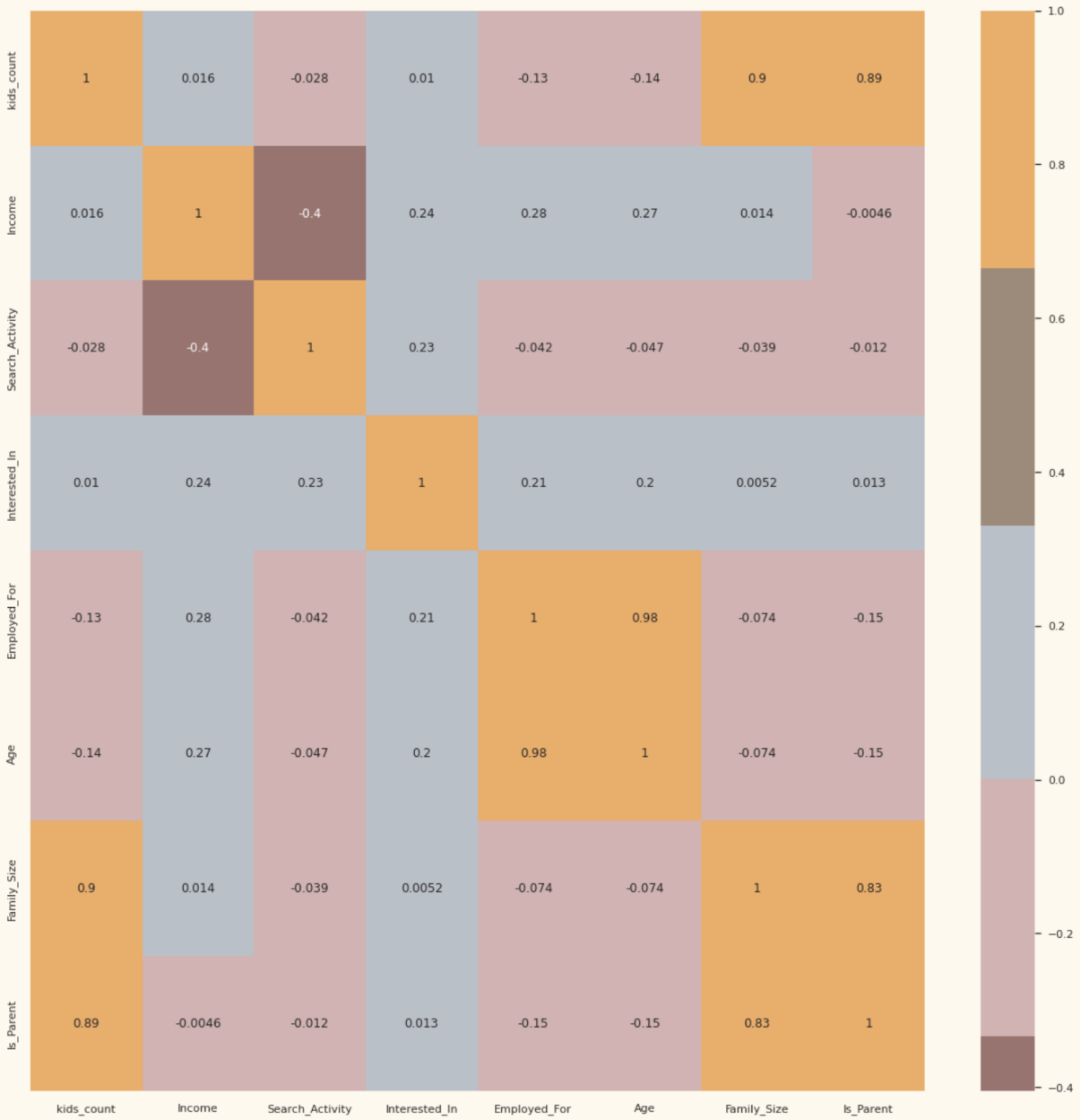

Рис.3.4. Інфографіка кореляції між ознаками

Кореляційний аналіз показує, що дані відносно чисті, тому ці функції можна додати та використовувати. Попри те, що кореляція між розміром сім'ї та доходом є невисокою, ця функція допоможе точніше описати майбутні кластери.

3.3. Розробка методу кластеризації даних із урахуванням ваг

3.3.1. Застосування статистичного методу перцентилів

Техніки кластеризації стали ключовими, сприяючи інтуїтивному розділенню великих наборів даних на ідентифіковані групи без залежності від

існуючих міток. Алгоритм K-means, завдяки своїй простоті та обчислювальній ефективності, завжди залишався основою в застосуваннях кластеризації. Однак він не позбавлений своїх слабких сторін. Основним серед його обмежень є виражена залежність від початкового розташування центроїдів. Ця властивість робить стандартний K-means вразливим до різних результатів у різних ітераціях, що може призвести до потенційної нестабільності в результатах кластеризації.

Знаючи про ці проблеми і обмеження та прагнучи заповнити ці прогалини, було вирішено змінити підхід. Основою підходу є інтеграція ваг. Це дозволяє тонко розрізняти точки даних на основі їх внутрішньої важливості, сприяючи механізму кластеризації, де значущі точки даних відіграють виражений вплив на формування кластера.

Стандартний метод k-means вважає, що всі ознаки мають однакову вагу, і враховує їх рівномірно при обчисленні відстаней між точками та центроїдами [87]. Однак, у реальних даних можуть бути ознаки, які мають більший вплив на кластеризацію або важливіші для досліджуваної задачі. Модифікований метод k-means дозволяє встановлювати різні ваги для різних ознак, враховуючи їхню важливість або вплив на кластеризацію. Це може виконуватися на основі експертного знання, досліджень або інших критеріїв, які визначають значущість ознак.

В спробі протистояти непередбачуваності, що виникає від випадкової ініціалізації центроїдів у стандартному K-means, пропонується змінена модель за відсотковим методом. Використовуючи аналіз головних компонентів (PCA) для попередньо оброблених даних, набір даних систематично розділяється за допомогою відсоткових значень для отримання початкових центроїдів. Цей перехід від випадкового до детермінованого, базованого на відсотках підходу, є важливим кроком, що забезпечує більш стійкі та надійні результати кластеризації.

Модель перцентилів визнана у світі статистики, є гнучким інструментом для методичного розділення наборів даних. Вона поділяє заданий набір даних на 100 відокремлених секцій або частин. Кожна секція представляє рівно 1 відсоток від усього змісту набору даних. До прикладу, якщо говорити про 25-й відсотковий показник, то мається на увазі підмножина даних, яка охоплює саме 25 відсотків від усього набору даних. Ця структура надає ілюстративний погляд на те, як розподілені дані. Перевага відсоткового підходу полягає у його адаптивності. Залежно від вимог та конкретних значень, які надаються, цей метод можна використовувати для розділення набору даних на різні розподіли, що надає значущу гнучкість у аналізі та інтерпретації даних [41].

Формула перцентилів представлена так:

$$R = \frac{P}{100} * (n + 1)$$

Де R - це ранг або позиція в наборі даних, що відповідає бажаному відсотку. P - це бажаний відсоток (наприклад, якщо проводиться пошук 25-го відсотка, P буде дорівнювати 25). n - це загальна кількість значень даних у наборі даних.

Стандартна формула передбачає однакову важливість всіх точок даних. Це припущення ігнорує можливі відмінності у важливості точок даних, що призводить до неоптимальних результатів кластеризації. Основний недолік стандартної формули полягає в її байдужості до різноманітності важливості даних. У практичних наборах даних, особливо у спеціалізованих областях, не всі точки даних є однаковими. Деякі мають більшу важливість через різноманітні фактори, включаючи їх частоту, контекст, в якому вони з'являються, або специфічну для домену релевантність.

Використовуючи універсальний підхід, звичайна формула перцентилів часто призводить до помилкового розміщення центроїдів, яке не відповідає справжній внутрішній структурі або важливості набору даних. Для виправлення цього необхідно внести зміни в традиційний підхід. Інтегруючи механізм

зважування, можна врахувати нерівномірну важливість, яка є властивою для багатьох реальних наборів даних. Таке глибоке розуміння гарантує, що розміщення центроїдів відображає не лише розподіл даних, але й наявну важливість кожної точки даних. Підхід, який базується на інтеграції ваг, відкриває шлях до більш стратегічного, орієнтованого на дані процесу кластеризації.

Для розрахунку зваженої позиції, потрібно видозмінити формулу, яка виглядатиме наступним чином:

$$position_i = \sum_{j=1}^{i} w_j$$

Де обчислюється зважена позиція $position_i$ для точки даних на *i*-й позиції. Ваги рівня оцінки задоволеності користувача *w* відображають важливість кожної точки даних у наборі даних. В загальному, формула підсумовує ваги всіх точок даних до і-тої точки.

Після отримання зваженої позиції $position_i$, наступна формула застосовується для конвертації значення у перцентиль.

$$percentile_i = \frac{position_i}{\sum_{j=1}^{N} w_j} * 100$$

Де $\sum_{j=1}^{N} w_j$, - це загальна вага всього набору даних, а N представляє загальну кількість точок даних. Поділяючи зважене положення на загальну вагу і домножуючи її на 100, можна розрахувати, який відсоток ваги положення *i*-тої точки даних представляє відносно ваги всього набору даних.

Цей підхід до визначення перцентилю враховує ваги та надає більш упереджене (на основі ваг) розподілення набору даних. Використання цього підходу на основі зваженого перцентилю для визначення початкових центроїдів означає, що центроїди розташовані стратегічно на основі не тільки розподілу даних, але й на значимості (як представлено вагами) кожної точки даних.

3.3.2. Розрахунок початкових центроїдів

На цьому етапі алгоритму завдання полягає у сегментації даних на основі розрахованих перцентилів таким чином, щоб враховувати внутрішні ваги даних. Набір даних ділиться на сегменти, де "важливість" кожного сегмента визначається сумою ваг точок даних в цьому сегменті. Потім ці сегменти використовуються для розрахунку початкових центроїдів. Формула, яка визначає початкові центроїди:

$$centroid_{segment} = \frac{\sum_{i=1}^{N_{segment}} (x_i * w_i)}{\sum_{i=1}^{N_{segment}} w_i}$$

Де $centroid_{segment}$ представляє центроїд конкретного сегмента даних. N - це кількість точок даних в цьому сегменті. x_i - це значення і-тої точки даних у сегменті. w_i - це вага, асоційована з і-тою точкою даних.

Ця формула розраховує зважене середнє конкретного сегмента даних для визначення його центроїда. Для кожної точки даних у сегменті її значення (x_i) множиться на її вагу (w_i), а потім підсумовується для всіх точок даних у сегменті. Потім ця сума ділиться на загальні ваги точок даних в цьому сегменті, щоб отримати центроїд цього сегмента. Результатом є центроїд, який не тільки відображає центральну тенденцію сегмента, але також коригується на основі ваги кожної точки даних.

Значущість цього підходу стає очевидною, коли проводиться розгляд наборі даних із різною важливістю між точками даних. Використовуючи зважене середнє замість простого арифметичного середнього, отримані центроїди розташовані таким чином, що вони усвідомлюють розподіл даних, а також значущість кожної точки даних. У сценаріях кластеризації це гарантує, що кластери формуються навколо центроїдів, які дійсно представляють центри даних, що призводить до більш точних та змістовних призначень кластерів.

3.3.3. Розроблення модифікованого методу Mini Batch K-means

Хоча стандартний K-means і є основою для методів кластеризації, у нього є свої недоліки, зокрема його залежність від випадкової ініціалізації центроїдів. Масштабування залишається ключовим питанням в кластеризації, особливо зі зростаючими об'ємами даних. З цією метою було інтегровано структуру Mini-Batch K-means [42]. Це сприяє швидкому зближенню, обробляючи 'міні-пакети' наборів даних в послідовних ітераціях, поєднуючи швидкість з точністю - комбінація, яка часто відсутня у стандартному алгоритмі K-means при роботі з великими наборами даних. Mini-Batch K-means – це оптимізований варіант його традиційного аналога, який спеціально розроблений для підвищення швидкості без істотного зниження якості кластерів. Замість використання всього набору даних на кожній ітерації, алгоритм Mini-batch K-means працює на випадкових підмножинах або "міні-пакетах" даних [43]. Ця фундаментальна зміна призводить до значних обчислювальних економій, роблячи алгоритм збіжним швидше і значно скорочуючи час обробки.

Алгоритм роботи наступний [44]:

- Ініціалізація центроїдів:

  Проводиться обрання початкових центроїдів, використовуючи центроїди, отримані з розбиття набору даних з вагами за моделлю відсоткового розподілу. Центроїди беруться з виводу сегментації даних на основі відсоткового розподілу. Ці центроїди, отримані з вагових відсотків, забезпечують більш орієнтовану на дані ініціалізацію, а не випадковий або навіть простий підхід k-means++.

  Нехай K - кількість кластерів.

  Нехай X = $\{ x_1, x_2, ..., x_n \}$ - множина точок даних.

  Нехай C = $\{ c_1, c_2, ..., c_K \}$ - множина початкових центроїдів.

  Нехай W = $\{ w_1, w_2, ..., w_K \}$ - множина ваг рівня оцінки задоволеності користувача для кожного кластеру.

- Розбиття датасету на частини:

  Випадково обирається підмножина (міні-пакет) набору даних. Ця підмножина буде використана в поточній ітерації оптимізації k-means. Замість використання повного набору даних, обирається менший, випадковий зразок. Це пришвидшує ітеративний процес і може допомогти уникнути локальних мінімумів.

$$M \subset D, |M| = b$$

  Де M – підмножина набору даних, D – весь набір даних, b – розмір пакету

- Обчислення відстані:

  Нехай *d(x, c)* - функція відстані між точкою *x* та центроїдом *c*.

  Нехай *s(x)* - рівень оцінки задоволеності користувача для точки *x*.

  Ваговий коефіцієнт *W(c)* використовується, щоб врахувати вагу рівня оцінки задоволеності для центроїда *c*. Тоді, формула для обчислення відстані з урахуванням ваги: *d(x, c) = W(c) * s(x)*.

- Призначення точок даних найближчому центроїду:

  Для кожної точки даних у міні-пакеті знаходиться найближчий центроїд, використовуючи зважену метрику відстані. Точки даних асоціюються з центроїдами на основі зваженої відстані. Тут метрика відстані коригується за допомогою ваги, тому точки з вищими вагами мають більший вплив на призначення центроїда.

$$C_i = arg\,min_{c \in C} w_i * d(x_i, c)$$

  Де C_i – призначений центроїд для точки даних x_i, C – множина центроїдів, w_i – вага точки даних x_i.

- Оновлення центроїду

$$c_j = \frac{\sum_{i \in M_j} w_i * x_i}{\sum_{i \in M_j} w_i}$$

Де c_j – нова позиція j-го центроїда, M_j – це підмножина точок даних міні-пакету, призначена j-му центроїду. Ця формула розраховує зважене середнє для точок даних у міні-пакеті, призначених кожному центроїду. Точки з вищими вагами матимуть більший вплив на оновлену позицію центроїда.

- Перевірка на збіжність:

  На цьому етапі перевіряється, чи стабілізувалися центроїди (тобто вони майже не рухаються між послідовними ітераціями) або чи алгоритм вже працював протягом максимальної кількості ітерацій. Якщо виконується хоча б одна з умов, алгоритм зупиняється, припускаючи, що він або знайшов хороше кластеризування, або не отримає користі від подальших ітерацій. Послідовність наступна:

  якщо $max_j\, d(c_j^{(new)}, c_j^{(old)}) < iteration > max\, iterations$ тоді зупинитись. Де $c_j^{(new)}, c_j^{(old)}$ – нова та стара позиції j-го центроїда відповідно.

Отже, особливостями та перевагами модифікованого алгоритму є:

- Стійке розташування початкових центроїдів: Традиційне кластеризування K-means, що є чутливим до початкового розташування центроїдів, часто стикається з проблемами, такими як потрапляння у локальні оптимуми, що може призвести до недостатньо ефективних результатів кластеризації. Модифікований алгоритм K-means, який використовує модель відсоткового розподілу, гарантує, що початкові центроїди розташовані не довільно чи випадково, але систематично розподілені з урахуванням внутрішньої структури даних. Це не тільки надає більш обгрунтовану відправну точку, але і підвищує ймовірність досягнення глобального оптимуму для рішення по кластеризації.
- Ефективний розрахунок за допомогою міні-пакетів: З ростом розміру наборів даних, обчислювальні витрати, пов'язані зі стандартним алгоритмом K-means, можуть стати невиправданими. За допомогою

техніки міні-пакетів, модифікований алгоритм обробляє підмножини даних на кожній ітерації. Цей підхід зберігає сутність розподілу даних, тим самим забезпечуючи точність кластеризації, при цьому значно прискорюючи обчислення, що робить його особливо вдалим для великих наборів даних.

- Включення ваги даних: особливість модифікованого алгоритму K-means - це його здатність призначати ваги окремим точкам даних. Це означає, що алгоритм може надавати різне значення різним точкам даних на основі зовнішніх знань або специфічних характеристик даних. Така система зважування дозволяє отримати більш витончене рішення по кластеризації, відображаючи основну значущість різних сегментів даних.
- Зменшення розмірності за допомогою PCA: Інтеграція аналізу основних компонент (PCA) дозволяє алгоритму працювати у просторі зі зменшеною розмірністю, акцентуючи найбільш значущі відхилення даних. Це не тільки зменшує обчислювальну складність, але й мінімізує вплив менш інформативних особливостей, тим самим зосереджуючись на найбільш важливих аспектах даних для кластеризації.
- Ефективність збіжності: Перевірка на збіжність у модифікованому алгоритмі гарантує, що ітеративний процес зупиняється, як тільки центроїди стабілізуються. Це уникає непотрібних обчислень і надає чіткий критерій зупинки, гарантуючи оптимальне використання ресурсів.

Після формування кластерів, необхідно зрозуміти до яких груп належать користувачі із допомогою спільних діаграм (рис.3.5.). Спільні діаграми корисні для дослідження взаємозв'язку між двома змінними, такими як їх кореляція, кластеризація або розподіл. Поєднуючи різні типи графіків, спільні діаграми можуть надати більш повне уявлення про дані, ніж окремі графіки. Вони також можуть бути налаштовані для виокремлення певних особливостей або патернів в даних, таких як викиди, тенденції або кластери. На основі спільних діаграм

можна буде дійти висновку про те, яка група може бути цільовою, а також дізнатись, хто потребує більше уваги з боку маркетингової команди.

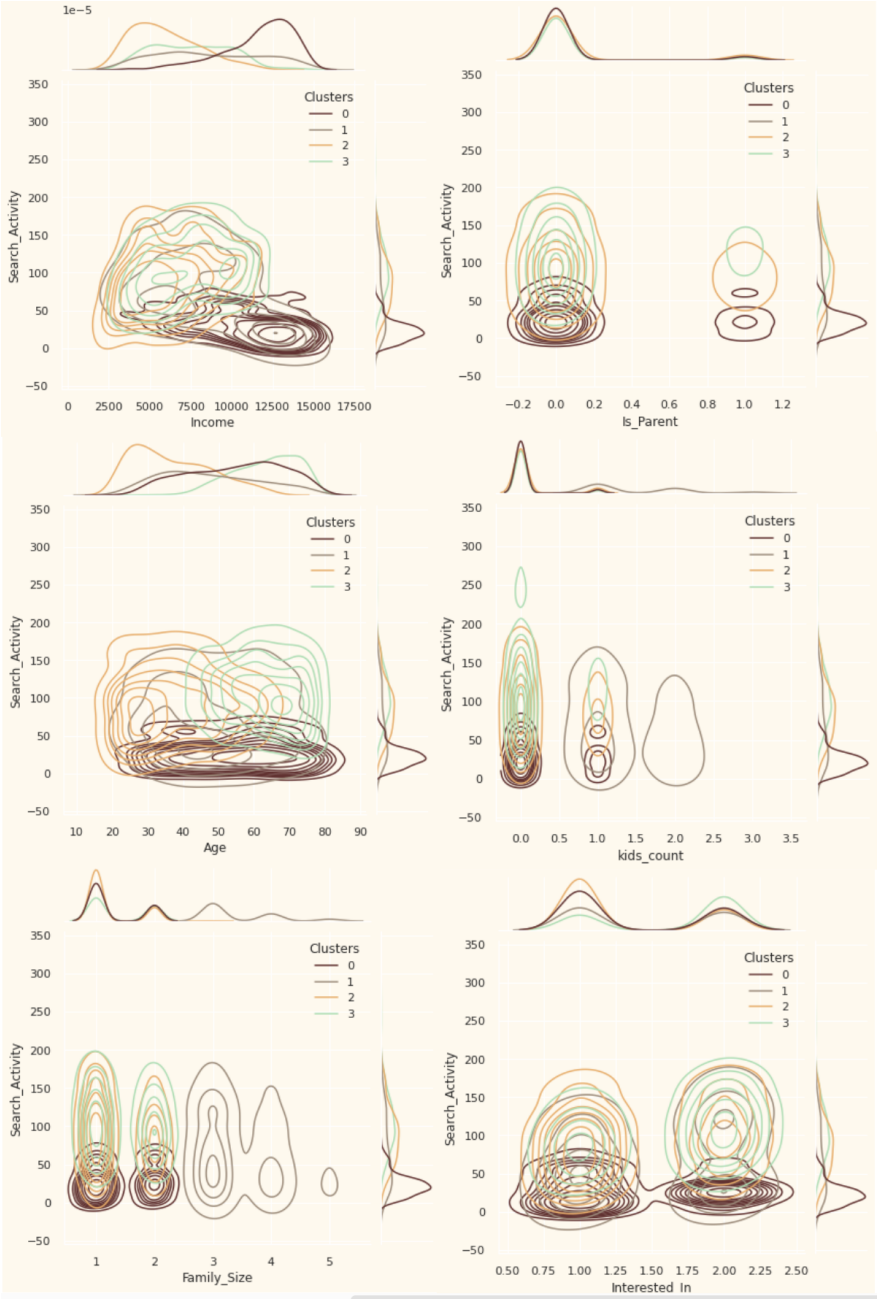

Рис. 3.5. Набір спільних діаграм на основі зазначених параметрів

Після огляду та аналізу наведених вище діаграм можна зробити загальне профілювання цільових груп:

- Перша група (0-й кластер) має високий дохід, в основному без дітей, вікова група переважно від 38 до 70 років. У більшості випадків

користувачі є без пари і більше зацікавлені у купівлі чи інвестиції у нерухомість.

- Друга група (1-й кластер) має дохід вище середнього, в середньому має 1-2 дитини, вікова група переважно від 30 до 50 років. У більшості випадків користувачі мають пару та здебільшого зацікавлені у купівлі нерухомості.
- Третя група (2-й кластер) має невеликий дохід, більшість не має дітей, переважає у віці 20–30 років. В основному користувачі без пари та зацікавлені у оренді нерухомості.
- Четверта група (3-й кластер) має середній дохід, в загальному вже немає малих дітей. Вікова група 60–80 років. По розміру сім'ї можна бачити, що статистичні дані розподілені більш-менш рівномірно – можуть бути як із парою, так і без. Більшість зацікавлена в купівлі чи інвестиції нерухомості, хоча є група, яка зацікавлена в оренді.

На основі цих даних працівники агентства ринку нерухомості матимуть можливість краще оцінити які групи користувачів у них існують, а також сформувати якісні маркетингові кампанії. Також, враховуючи низьку активність першої групи користувачів, а також зважаючи на високий дохід цієї групи, можна певним чином стимулювати співпрацю.

3.4. Висновки до розділу 3

У даному розділі розроблено алгоритм підготовки даних. Проведено аналіз та порівняння методів обробки пропущених значень. Проведено аналіз методів виявлення та видалення дублікатів, проаналізовано та порівняно різні підходи до виявлення та усунення викидів. Проаналізовано роботу методів зменшення розмірності даних та виділення нових ознак. Застосовано статистичний метод перцентилів для розрахунку початкових центроїдів. Розроблено метод кластеризації різнотипових даних, який дозволяє працювати з потоковими даними на основі поділу на пакети.

РОЗДІЛ 4. РОЗРОБЛЕННЯ АРХІТЕКТУРИ ТА АПРОБАЦІЯ РЕЗУЛЬТАТІВ

У даному розділі було розроблено архітектуру інформаційної системи та подано її представлення у вигляді діаграм зв’язків та діаграм послідовності. Також описано процес роботи системи та візуально з допомогою діаграми варіантів. Проведено порівняльні тестування швидкодії та якості кластеризації. Проведено зменшення вартості розгортання системи та аналіз витрат.

Результати розділу опубліковано у працях автора [100]

4.1. Побудова архітектури інформаційної системи

У даному дисертаційному дослідженні розробляється інформаційна система, яка призначена для створення та управління профілями користувачів, що цікавляться інвестиціями, продажем або орендою нерухомості. Обрано поширений тип інформаційної системи – веб-застосунок, який забезпечує взаємодію з користувачами через API-сервіси [88] та зберігання необхідних даних у базі даних.

Ця система має потенціал підтримувати різноманітні клієнтські потреби, аналізуючи їхню поведінку, інтереси та вподобання. Використання веб-застосунку дозволяє забезпечити широкий охоплюючий доступ до системи для користувачів з будь-якого місця, де є доступ до Інтернету. Веб-інтерфейс дозволяє зручно взаємодіяти з системою та забезпечує зручність і легкість використання для користувачів різного рівня технічної грамотності.

Глибоке розуміння системи, особливо в контексті управління відносинами з клієнтами, вимагає детального вивчення взаємодії та відношень між різними сутностями у межах цієї системи. Клієнти, менеджери та дані, що з'єднують їхню взаємодію, створюють триаду, яка є критично важливою для ефективного функціонування будь-якої орієнтованої на клієнта бізнес-моделі.

Розділяючи клієнтів на відмінні профілі, менеджери можуть більш ефективно налаштовувати свої стратегії та взаємодію. В основі цієї моделі

взаємодії лежить клієнт. Їхні поведінка, уподобання та відгуки формують дані, які генерує система. З іншого боку є менеджери, які виробляють стратегії та приймають рішення на основі висновків, отриманих з цих даних. На рис.4.1. зображено діаграму варіантів, яка описує взаємодію користувача, менеджера та системи.

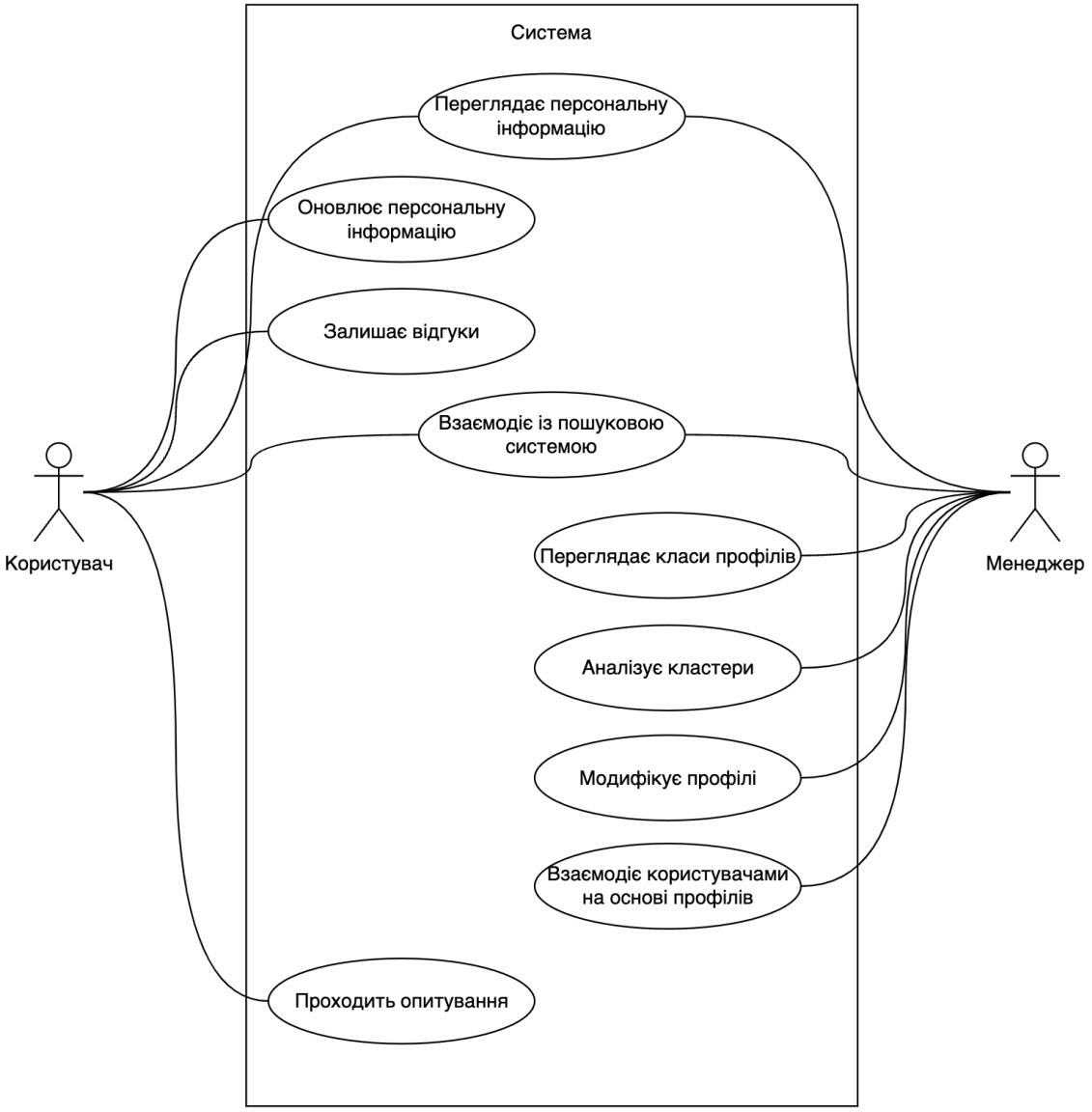

Рис.4.1. Діаграма варіантів взаємодії користувача, менеджера та системи

Для побудови системи було обрано мікросервісну архітектуру. Мікросервісна архітектура - це підхід до розробки програмного забезпечення, при якому додаток розбивається на невеликі, самодостатні та незалежні сервіси,

які працюють разом із застосуванням локальних викликів та мережевих протоколів [89]. Кожен мікросервіс має свою власну функціональність і може бути розгорнутий, масштабований та керований окремо від інших сервісів. Мікросервісна архітектура дозволяє розділити складний додаток на менші компоненти, що спрощує розробку, розгортання та підтримку системи. Кожен сервіс може бути розроблений, оновлений та масштабований незалежно, що забезпечує гнучкість та швидкість розробки. Мікросервіси можуть використовувати різні технології, бази даних та залежності, що дозволяє вибрати найкращі інструменти для кожного сервісу.

Система повинна надавати наступні можливості:

- забезпечувати можливості створення, редагування та видалення профілів користувачів, зберігання їх основних даних та зв'язків між ними.
- надавати можливість користувачам виражати свої інтереси, встановлювати переваги та налаштовувати свої уподобання.
- забезпечувати можливість надання персоналізованих рекомендацій та пропозицій користувачам на основі їхніх інтересів, потреб та історії взаємодії з системою.
- надавати можливість опрацьовувати дані користувачів та на їх основі профілювати їх у певні групи

Враховуючи дані вимоги, було прийнято рішення у створенні наступних сервісів:

- **Collaborators API** - сервіс, який дозволяє обробляти історію спілкування з користувачами та генерувати пропозиції. Він зберігає історію взаємодії з користувачами, такі як запити, відповіді, пропозиції та повідомлення. За допомогою цього сервісу, менеджери системи можуть переглядати та аналізувати історію взаємодії з користувачами, що допомагає зрозуміти їхні потреби та упередженості. Крім того, сервіс використовує цю інформацію для генерації персоналізованих пропозицій та рекомендацій користувачам на основі їхньої історії спілкування

- **Customer Profiling API** - це сервіс, який забезпечує функціональність профілювання користувачів. Він відповідає за збір та обробку даних про користувачів системи, таких як особиста інформація, інтереси, покупки, взаємодії з системою та інше. Цей сервіс дозволяє створювати та оновлювати профілі користувачів, а також виконувати аналітику та розрахунки для персоналізації пропозицій та рекомендацій.
- **Core Profiling Service** - центральний сервіс, який виконує основні операції профілювання користувачів. Він забезпечує обробку великого обсягу даних, виконує складні алгоритми аналізу та кластеризації, а також зберігає інформацію про профілі користувачів. Цей сервіс включає в себе різні модулі для відбору ознак, зменшення розмірності, обчислення ваг та інше.
- **Customer API** - сервіс, який надає доступ користувачам до їх особистої інформації та функціональності системи. Він дозволяє користувачам переглядати, редагувати та керувати своїми профілями, переглядати нерухомість, здійснювати транзакції та взаємодіяти з іншими компонентами системи.
- **Listing API** - сервіс, який забезпечує функціональність пов'язану з нерухомістю. Він дозволяє додавати нові нерухомості до системи, редагувати та видаляти наявні оголошення про нерухомість, а також отримувати інформацію про доступну нерухомість для користувачів.
- **Auth Service** - сервіс авторизації, який забезпечує безпеку та ідентифікацію користувачів. Він використовує механізми аутентифікації, генерації токенів та перевірки дозволів для забезпечення доступу до системи тільки авторизованим користувачам.
- **API Gateway** - компонент, який виконує функцію API Gateway і маршрутизації запитів до відповідних сервісів. Він контролює доступ до API, забезпечує автентифікацію та авторизацію, а також здійснює моніторинг та журналювання запитів.

- **[NAME] API Client** - компонент, який представляє клієнтську сторону системи і дозволяє взаємодіяти з різними сервісами за допомогою API. API Client використовується користувачами системи, розробниками додатків або інтеграцій, щоб отримувати доступ до функціональності системи через відповідні API. Цей компонент забезпечує зручний та простий спосіб взаємодії з системою, передавати запити, отримувати відповіді та обробляти дані.
- **Google Cloud Functions** - зовнішній сервіс, який є інтегрований з Core Profiling Service для реалізації додаткової функціональності, пов'язаної з профілюванням користувачів. Цей сервіс надає можливість розгортання і виконання функцій у хмарному середовищі Google Cloud.

На рис. 4.2. зображено загальну архітектуру системи

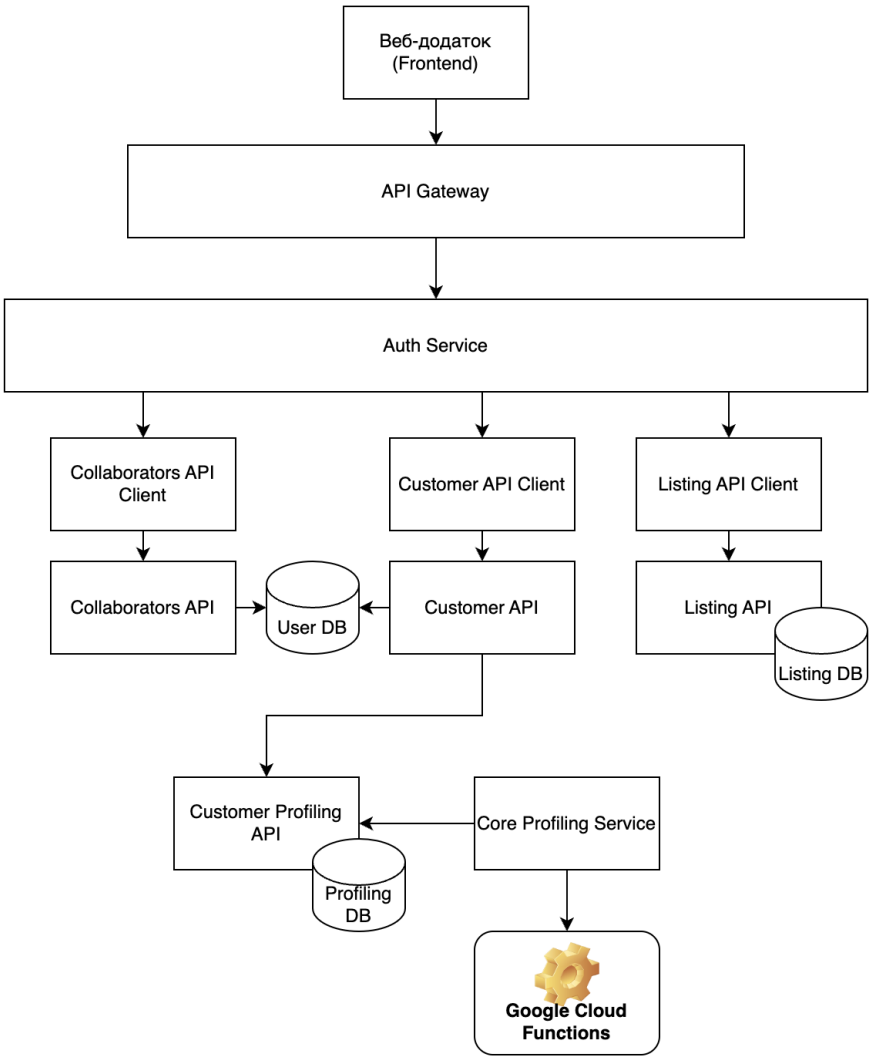

Рис. 4.2. Загальна архітектура системи профілювання користувачів

Обрання Google Cloud Functions для інтеграції з Core Profiling Service має кілька обґрунтованих причин. Google Cloud Functions є частиною хмарного середовища Google Cloud, що забезпечує масштабованість та надійність інфраструктури. Це дозволяє легко розгортати функції в хмарному середовищі і автоматично масштабувати їх в залежності від потреб системи. Такий підхід гарантує високу доступність та продуктивність системи навіть при зростанні обсягу оброблюваних даних. Окрім цього, Google Cloud Functions пропонує безшовну інтеграцію з іншими сервісами та інструментами, що надаються хмарним середовищем Google Cloud. Це включає можливість використовувати

інші сервіси, такі як бази даних, системи кешування, системи керування ідентифікацією та багато інших. Інтеграція з такими сервісами дозволяє розширити функціональність системи та забезпечити повну обробку та збереження даних профілювання.

4.2. Проектування структури даних у інформаційній системі

В залежності від вимог проекту та характеристик даних, були обрані такі інструменти, як PostgreSQL, Elasticsearch та Redis.

PostgreSQL є реляційною системою керування базами даних (СКБД), яка пропонує надійне зберігання даних і підтримує стандарти SQL. Він володіє широким спектром можливостей для моделювання даних, забезпечує ACID-властивості та гарантує цілісність даних [90]. PostgreSQL є надійним вибором для збереження структурованих різнотипових даних, таких як дані про користувачів, історії спілкування, відгуки, інформацію про нерухомість тощо.

Elasticsearch - це розподілена система пошуку та аналізу даних, яка спеціалізується на швидкому та ефективному повнотекстовому пошуку. Він дозволяє швидко виконувати пошук, агрегацію та аналітику великого обсягу даних [91]. Elasticsearch має потужні функції, такі як ранжування за релевантністю, розширений пошук по ключовим словам та можливості фасетного пошуку. Використання Elasticsearch дозволить надати потужні можливості пошуку та аналітики даних в системі.

Redis - це високопродуктивна система кешування та сховище даних, яка оперує у пам'яті. Він надає швидкий доступ до даних завдяки своїй унікальній архітектурі та використанню ключ-значення для зберігання даних [92]. Redis добре підходить для кешування часто використовуваних даних, швидкого доступу до сесій користувачів та тимчасового зберігання даних. Використання Redis дозволить прискорити обробку даних та покращити продуктивність системи.

Під час проектування інформаційної системи було виявлено, що основна база даних використовується як центральний репозиторій для збереження

різноманітних типів даних, включаючи дані про користувачів та їх взаємодію з системою, а також дані, що відносяться до менеджменту. Однак, таке поєднання різних типів даних в одній базі може призвести до значного збільшення навантаження на систему, особливо при великому обсязі даних та високому рівні активності користувачів.

З метою оптимізації продуктивності та забезпечення ефективної роботи системи, було прийнято рішення розділити базу даних на окремі частини. Таким чином, дані, пов'язані з користувачами та їх взаємодією з системою, були відокремлені і збережені в окремій базі даних, що спеціалізується на цих видів даних.

На рис. 4.3. зображено діаграму зв’язків між сутностями для сервісів Customer API та Collaborators API. Основними сутностями є Users (містить у собі загальну інформацію про користувача), Roles (для того, щоб мати можливість розрізняти менеджмент від клієнтів), Experience (містить у собі інформацію про наявний досвід кожного користувача), Conversations (зберігає у собі інформацію про листування) та Preferences (містить у собі вподобання у типах нерухомості).

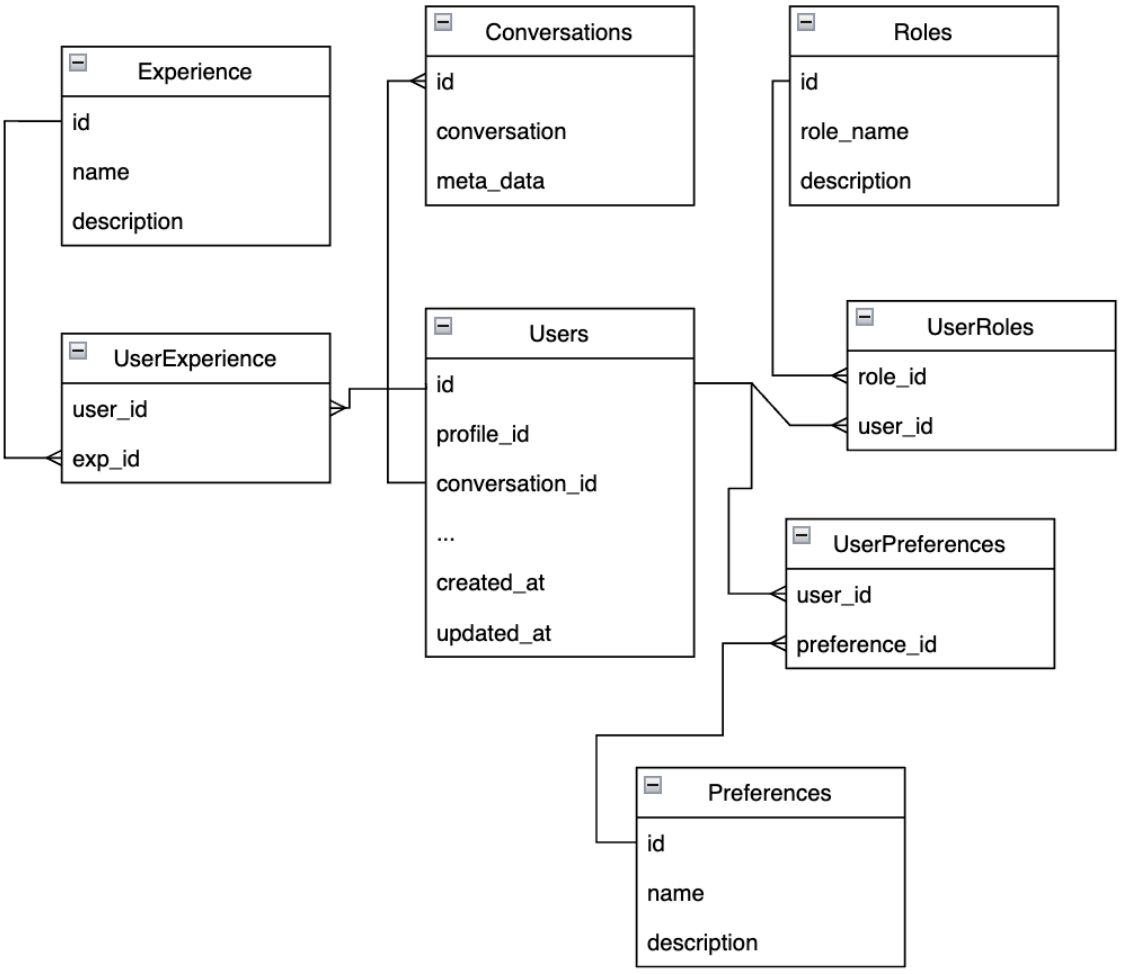

Рис.4.3. Діаграма зв’язків між сутностями для сервісів Customer API та Collaborators API

Далі розглядається БД для сервісу Customer Profiling API. Вона складається із сутностей Clusters, ClusterStats, Profiles та ProfileStats.

Сутність Clusters необхідна для збору загальної інформації про кластер. Її наявність дозволяє поглибити експертні знання менеджерів. Знання про кластери користувачів допомагає вдосконалювати системи рекомендацій та персоналізованого досвіду. Збереження інформації про кластери дозволяє налаштовувати рекомендації, контент та функціональні можливості, щоб краще відповідати потребам та інтересам кожної групи користувачів. Також збереження інформації про кластери дозволяє використовувати її для

моніторингу та оцінки ефективності кластеризації. Це дозволяє вдосконалювати алгоритми кластеризації, виправляти помилки та покращувати якість аналізу даних.

ClusterStats містить більш детальну інформацію та статистику про кожен кластер, таку як кількість користувачів у кожному кластері, середні значення певних атрибутів користувачів у кластері, розподіл користувачів за певними критеріями тощо. Це дозволяє зрозуміти характеристики кожного кластеру і використовувати цю інформацію для подальшого аналізу та прийняття рішень.

Profiles зберігає у собі інформацію та опис про кожен сформований профіль. Вона є частиною сутності User і доступ до неї робиться через API-запити.

ProfileStats зберігає у собі статистичну інформацію, таку як середній вік користувачів в кожному профілі, середній рівень задоволеності, популярність профілів тощо. ProfileStats є додатковим джерелом інформації для швидкого доступу до цих статистичних даних.

На рис. 4.4. зображено діаграму зв’язків для сервісу Customer Profiling API.

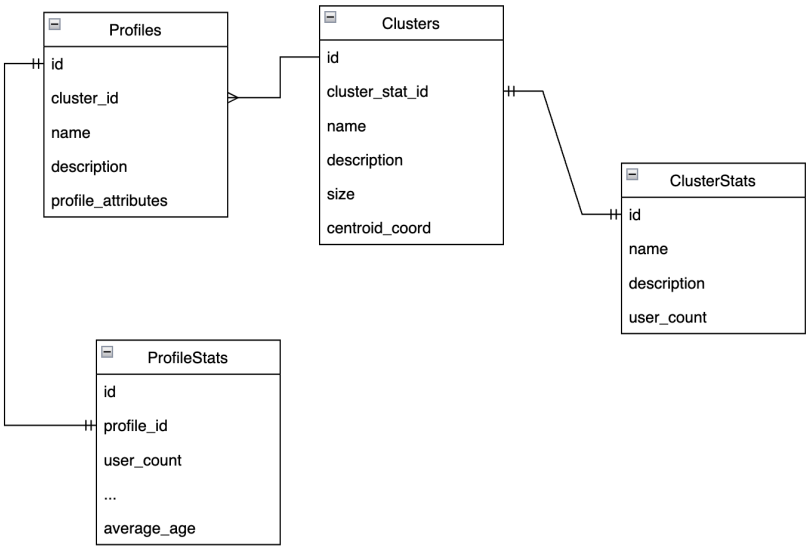

Рис. 4.4. Діаграма зв’язків для сервісу Customer Profiling API

4.3. Аналіз та моделювання бізнес-процесів

Для кращого розуміння роботи системи проводиться докладний аналіз та моделювання бізнес-процесів, які відбуваються всередині інформаційної системи, зосереджуючись на ключових етапах та послідовності дій, необхідних для досягнення поставлених цілей. Моделюються кроки, пов'язані з обробкою та зберіганням цих даних у базі даних системи. Також досліджуються процес кластеризації користувачів та створення профілів на основі отриманих даних.

На діаграмі послідовності (рис. 4.5.) показано взаємодію між користувачем і системою під час процесу реєстрації та заповнення персональної інформації. Користувач починає процес, відправляючи запит на реєстрацію до системи. Система відповідає запитом з формою реєстрації, на яку користувач повинен відповісти, заповнивши необхідні поля. Після цього користувач отримує запит на заповнення персональної інформації і отримує сторінку опитувальника з набором запитань. Після заповнення опитувальника користувач відправляє запит на збереження інформації, а система зберігає ці дані в базі даних. Користувач отримує підтвердження про успішну реєстрацію від системи.

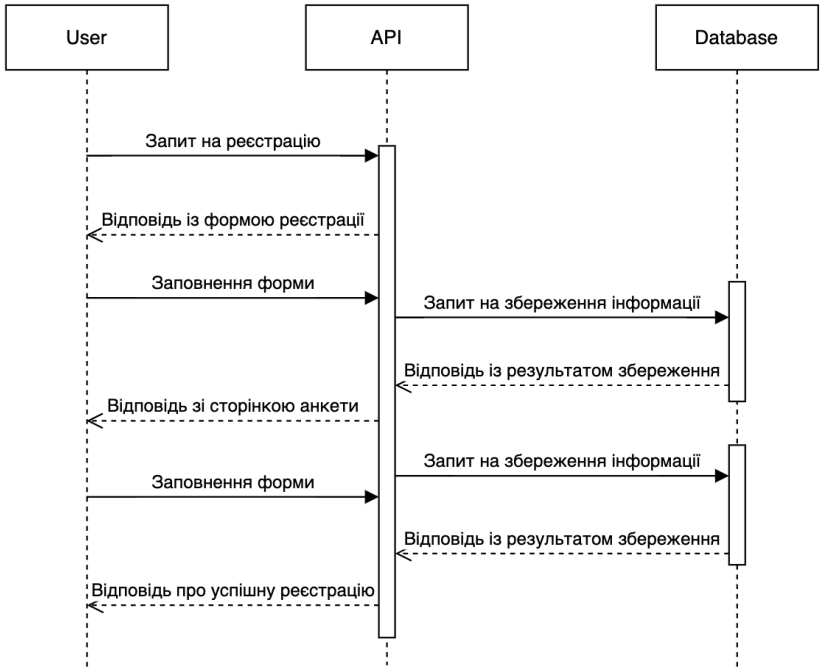

Рис. 4.5. Діаграма послідовності реєстрації користувача

Наступним етапом є аналіз даних користувача та присвоєння йому попереднього профілю для подальшої взаємодії. На рис. 4.6. зображено взаємодію між Customer API, Customer Profiling API та Core Profiling Service. Кроки взаємодії наступні:

- Customer API ініціює запит до іншого сервісу, який містить інформацію про кластери та профілі.
- Customer API передає отриману інформацію про користувача до Customer Profiling API.
- Customer Profiling API перевіряє існуючі профілі відповідного кластеру та, використовуючи потужності Core Profiling Service, призначає користувачеві відповідний профіль.
- Customer Profiling API надсилає інформацію про призначений профіль користувача до системи.
- Customer API отримує інформацію про призначений профіль користувача та зберігає цю інформацію в базі даних. Процес присвоєння профілю користувачеві на основі інформації з іншого сервісу завершено.

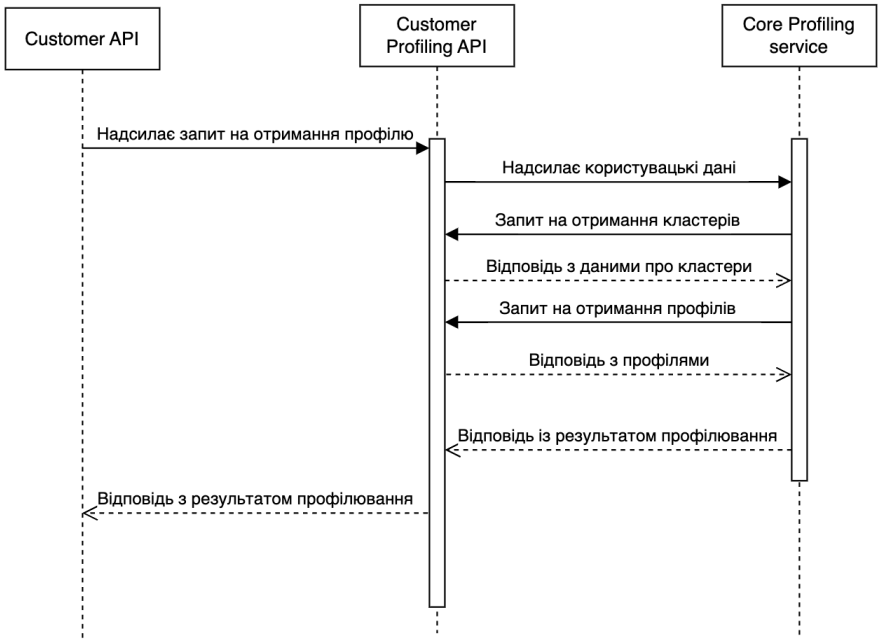

Рис.4.6. Діаграма послідовності призначення профілю новому користувачу

Останнім процесом, який необхідно детально розглянути є сам процес кластеризації даних. Процес відбувається раз у певний період, оскільки динаміка збільшення кількості нових користувачів є повільною.

- Виявлення та видалення викидів: Core Profiling Service використовує статистичні методи та алгоритми для виявлення викидів у даних користувачів. Це дозволяє ідентифікувати аномалії та видаляти їх з даних, щоб забезпечити чистоту та точність профілювання.
- Обробка пропущених значень та заповнення даних: Core Profiling Service перевіряє наявність пропущених значень у вхідних даних та застосовує алгоритми для їх обробки. Зокрема, сервіс може заповнювати пропущені значення доходу, використовуючи дані з бази даних або застосовуючи встановлені бізнес-правила для заповнення цих значень.
- Виявлення та видалення дублікатів: Core Profiling Service виконує перевірку наявності дублікатів у даних користувачів і видаляє їх, щоб уникнути некоректної обробки та забезпечити точність профілювання.
- Core Profiling Service робить запит на Google Cloud Functions для ініціювання процесу кластеризації та віддає підготовлені дані.
- Google Cloud Functions забезпечує нормалізацію даних, що дозволяє привести їх до стандартного формату або шкали.
- Google Cloud Functions проводить відбір ознак, використовуючи різні методи, такі як аналіз кореляції, статистичні метрики та алгоритми вибору ознак. Це дозволяє визначити найважливіші ознаки, які впливають на профілювання користувачів.
- Google Cloud Functions використовує алгоритми виділення нових ознак, які дозволяють побудувати нові характеристики на основі наявних даних. Це допомагає розширити набір ознак та забезпечити більш повне профілювання користувачів.

- Google Cloud Functions використовує методи зменшення розмірності даних для скорочення кількості ознак та побудови компактних представлень даних.
- Google Cloud Functions застосовує алгоритм кластеризації для створення нових кластерів користувачів та віддає результат Core Profiling Service.
- Core Profiling Service зберігає оновлені кластери, проводить перепрофілювання користувачів та зберігає дані.

На рисунку 4.6. зображено діаграму послідовності виконання процесу кластеризації із допомогою хмарного середовища.

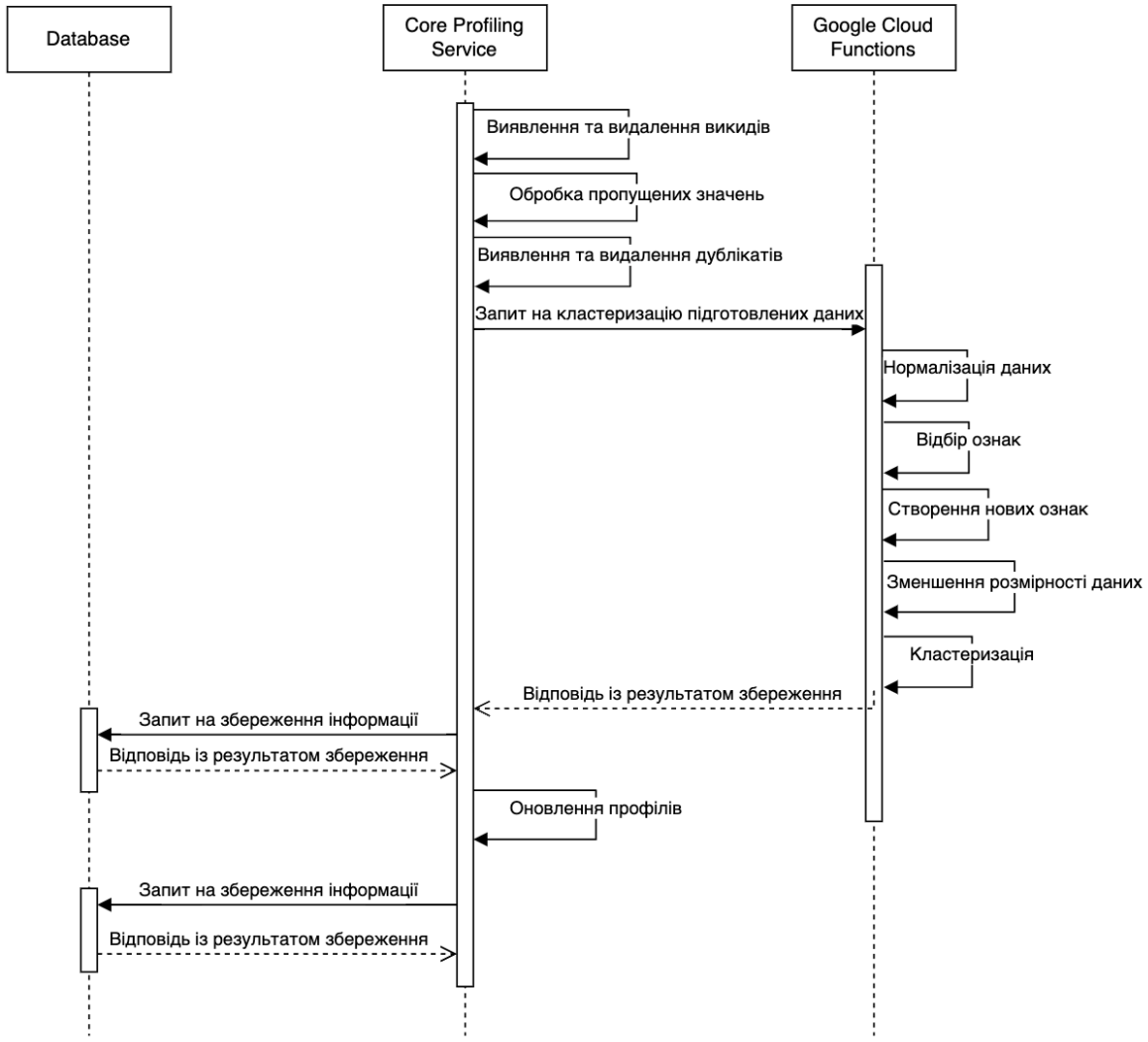

Рис. 4.6. Діаграма послідовності виконання процесу кластеризації

4.4. Оптимізація вартості розгортання системи. Аналіз витрат

Під час розробки архітектури системи було проведено детальний аналіз різних варіантів, зокрема розглянуто можливість використання сторонніх обчислювальних сервісів [93], таких як Google Cloud Functions. Однак, з урахуванням потреб та особливостей проекту, було прийнято рішення про комбінацію власних потужностей з безсерверними сервісами. Цей підхід дозволив досягти оптимального балансу між ефективністю та вартістю розгортання системи. Використання власних потужностей дозволяє зберігати контроль над обчислювальними ресурсами та забезпечити високу продуктивність у межах системи, тоді як безсерверні сервіси надають можливість гнучко масштабувати систему та знизити загальну вартість її експлуатації. Такий підхід дозволяє оптимізувати витрати та забезпечити високу якість та швидкість роботи системи.

Обчислення вартості розгортання системи можна провести на прикладі. Ціна за 1 мільйон запитів на місяць становить $0.40, ціна за ГБ-секунд становить $0.000008120. Ціна 1 ГБ-секунд відповідає одній секунді реального часу з наданою пам'яттю 1 ГБ.

База даних користувачів складає близько 720 тисяч записів. Після видалення дублікатів, на виході отримується приблизно 700 тисяч записів. Вага даних складає приблизно 200 МБ. Завантаження даних в систему становить 38 секунд, тобто 38 ГБ-секунд.

Підготовчі процеси перед кластеризацією (нормалізація, створення нових ознак, зменшення розмірності даних) тривали приблизно 37 секунд. Сама кластеризація в середньому триває 945 секунд. Сумарно кластеризація зайняла 1020 ГБ-секунд. Тут число можна заокруглити до 1100 ГБ-секунд через те, що система може перебувати і сплячому режимі і “холодний старт” додає часу на ініціалізацію.

Кластеризація відбувається на щоденній основі. Кількість запитів у системі в середньому становить 75 (близько 2250 запитів на місяць). Підсумувавши все, буде отримано наступне:

$$10000Gb \cdot s \times 0.00008120USD/Gb \cdot s \times 30 + 0.4\,USD = 2.836USD$$

Отже, використання стороннього сервісу обходиться менше 3 доларів на місяць. Враховуючи побічні ефекти, цифру можна заокруглити до 3 доларів на місяць.

Тепер можна поглянути на стандартний підхід із розгортання системи - це купівля виділеного сервера, або дроплета у хмарному середовищі. В середньому по ринку, ціна сервера з такими характеристиками складатиме приблизно 12 доларів на місяць.

Отже, при розгортанні системи з використанням сторонніх сервісів економія складатиме приблизно 9 доларів на місяць, або вчетверо менше порівняно зі стандартним підходом, що розглядає купівлю виділеного сервера або дроплета у хмарному середовищі. Така різниця у вартості може значно сприяти зниженню загальних витрат і забезпеченню більш економічного розгортання системи.

4.5. Апробація результатів

Для оцінки методу кластеризації було застосовано Індекс Данна [94] та метод силуету [95] для деяких поширених методів кластеризації даних. Індекс Данна розраховується як відношення найменшої відстані між кластерами до найбільшої внутрішньої відстані в кластері. Вище значення індексу Данна вказує на кращу кластеризацію, оскільки це означає, що кластери добре відокремлені один від одного і добре компактні.

Таблиця 4.1. відображає Індекс Данна для 2-7 кластерів різних методів кластеризації.

Таблиця 4.1

Порівняння методів кластеризації методом Індексу Данна

	Кількість кластерів					
Алгоритм	2	3	4	5	6	7
Ієрархічна кластеризація	0.4711	0.3723	0.3721	0.4051	0.3307	0.3401
K-MEANS	0.1457	0.2323	0.422	0.3251	0.2621	0.283
K-MEDOIDS	0.131489	0.214537	0.3932	0.316452	0.275616	0.2795
K-MEANS MINI-BATCH	0.1281	0.2011	0.4167	0.3063	0.2754	0.2591
Модифікований алгоритм	0.1157	0.2375	0.438	0.3167	0.2613	0.26
DBSCAN	0.0857	0.2723	0.3762	0.2521	0.2321	0.2415

На рис. 4.7. зображено графік порівняння Індексу Данна для різних методів кластеризації.

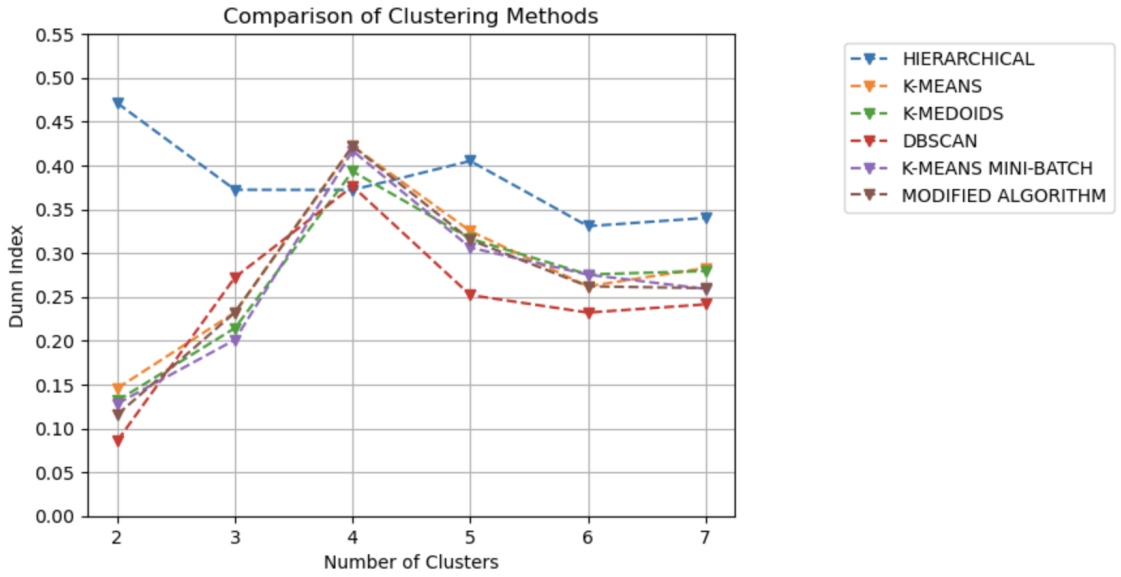

Рис.4.7. Графік порівняння методів кластеризації методом Індексу Данна

Метод силуету вимірює як добре кожен об'єкт був призначений своєму кластеру порівняно з іншими кластерами. Значення силуету варіюються від -1 до +1. Значення близькі до +1 вказують на добре відокремлені кластери, значення близькі до 0 вказують на те, що об'єкт на межі між двома кластерами, а значення близькі до -1 вказують на те, що об'єкт був неправильно призначений своєму кластеру.

У таблиці 4.2. відображено результати вимірів методом силуету.

Таблиця 4.2

Результати вимірів методів кластеризації даних методом силуету

	Кількість кластерів					
Алгоритм	2	3	4	5	6	7
Ієрархічна кластеризація	0.2591	0.1490	0.1575	0.2295	0.2147	0.1913
K-MEANS	0.2593	0.2127	0.2849	0.2656	0.2616	0.2112
K-MEDOIDS	0.2587	0.2103	0.2736	0.2597	0.2421	0.2099
K-MEANS MINI-BATCH	0.2554	0.2103	0.2772	0.2544	0.2245	0.2082
Модифікований алгоритм	0.2583	0.2116	0.2917	0.2611	0.2486	0.1979
DBSCAN	0.2579	0.2287	0.2676	0.1709	0.1566	0.1505

На рисунку 4.8. зображено графік порівняння вимірів методом силуету для різних методів кластеризації.

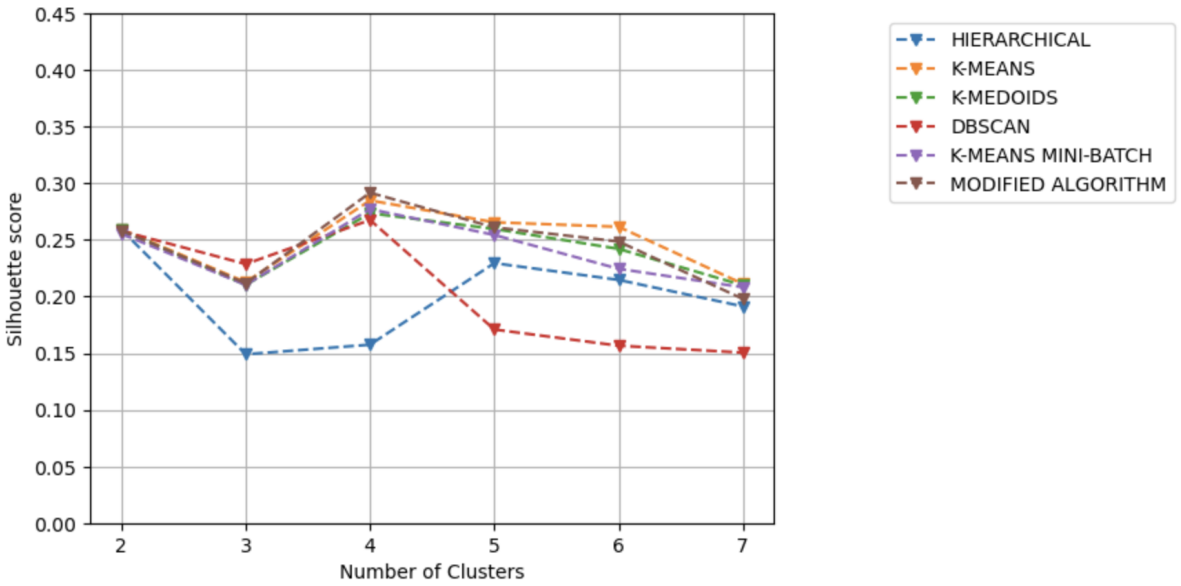

Рис.4.8. Графік порівняння методів кластеризації методом Силуету

Під час стандартної роботи сервера, навантаження на CPU та на оперативну пам’ять є стабільним - система працює у штатному режимі та без проблем може виконувати стандартні операції з базою даних та обробляти запити з клієнтської частини (рис.4.9. – 4.10.).

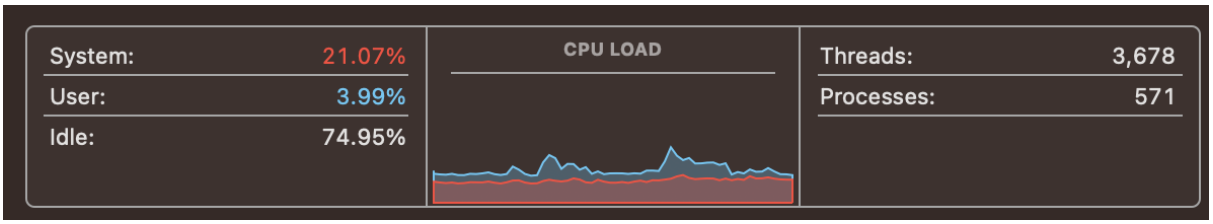

Рис. 4.9. Навантаження на процесор під час штатної роботи сервера

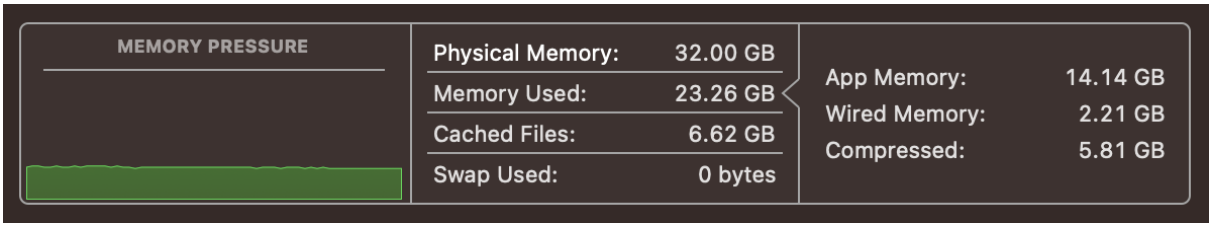

Рис. 4.10. Використання пам’яті під час штатної роботи сервера

При проведенні всіх кроків створення профілів користувачів (витягування даних, препроцесинг, кластеризація і тд.), навантаження на процесор та пам’ять суттєво збільшується (рис. 4.11. – 4.12.).

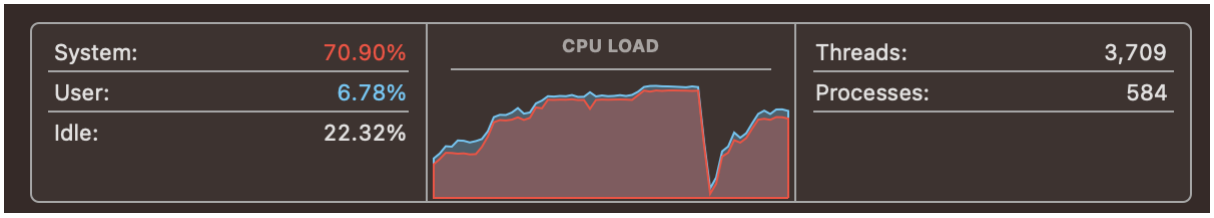

Рис.4.11. Навантаження на процесор під час кластеризації та формування звітів

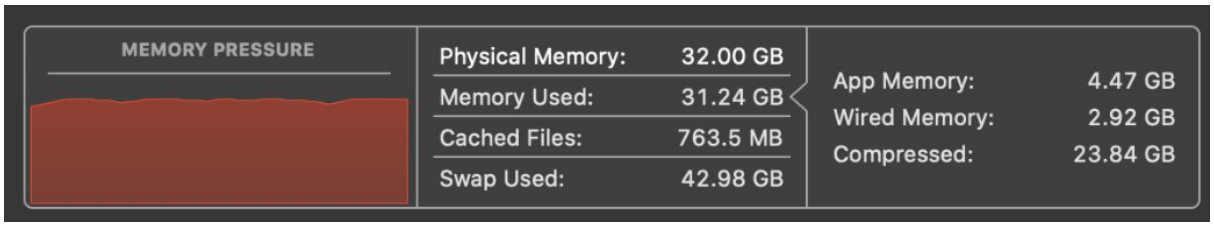

Рис.4.12. Використання пам’яті під час кластеризації та формування звітів

Попри те, що система має невеликий запас потужностей – вона залишається вразливою, якщо під час виконання обчислень буде спостерігатись аномальна активність на клієнтській частині, або у цей момент буде запрограмована масова розсилка сповіщень. Це передуватиме неконтрольованим відмовам під час роботи системи.

Після переведення складних обчислень на безсерверні технології, використання потужностей стало суттєво меншим, оскільки система, в основному надсилає запити на отримання необхідних даних, виконує підготовчі процеси та надсилає інформацію на Google Cloud Functions, які виконують усю необхідну роботу та надсилають оброблені дані назад. У результаті, основною роботою поточної системи є надсилання та отримання запитів та подальшим зберіганням отриманої інформації (рис.4.13. – 4.14).

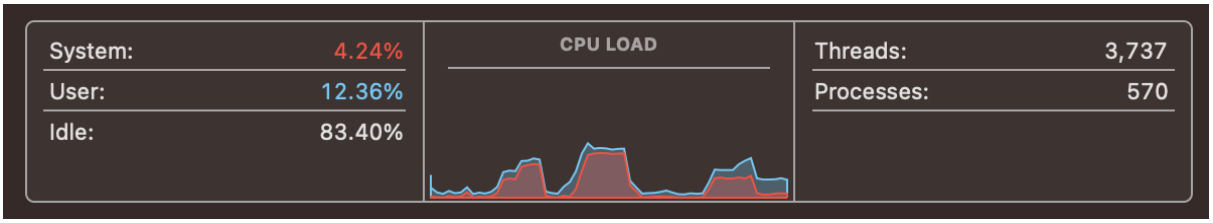

Рис.4.13. Навантаження на процесор із використанням хмарних технологій

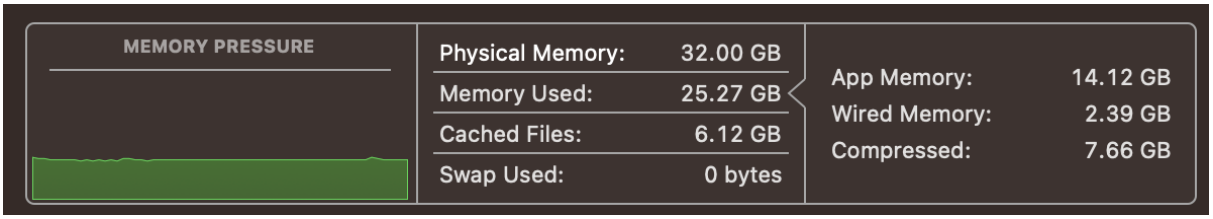

Рис.4.14. Використання пам’яті із використанням хмарних технологій

4.5.1. Порівняння швидкодії алгоритмів

Було проведено 50 тестів над набором даних. Час виконання для 4 кластерів з набором даних для модифікованого методу був майже постійний та склав у середньому 1424 секунди. У порівнянні з трьома іншими методами (k-means – 2859 секунд, kmeans++ – 2550 секунд, Mini-Batch k-means – 1907 секунд), нова модель перевершує k-means у приблизно 2 рази, kmeans++ у приблизно 1.79 разу, Mini-Batch k-means у приблизно 1.34 разу (рис.4.15.).

Під час застосування запропонованого методу, було отримано сталу кількість ітерацій у кожному тесті. Але існуючі методи k-means, kmeans++ та Mini-Batch k-means мають випадкову ітерацію. Через це, час виконання стандартних алгоритмів був нестабільним та більшим у більшості випадків.

Постійна оптимальна ітерація в порівнянні з існуючою моделлю k-means, kmeans++ та Mini-Batch k-means та найкоротший час виконання свідчать про те, що запропонована модель краще працює для великих наборів даних.

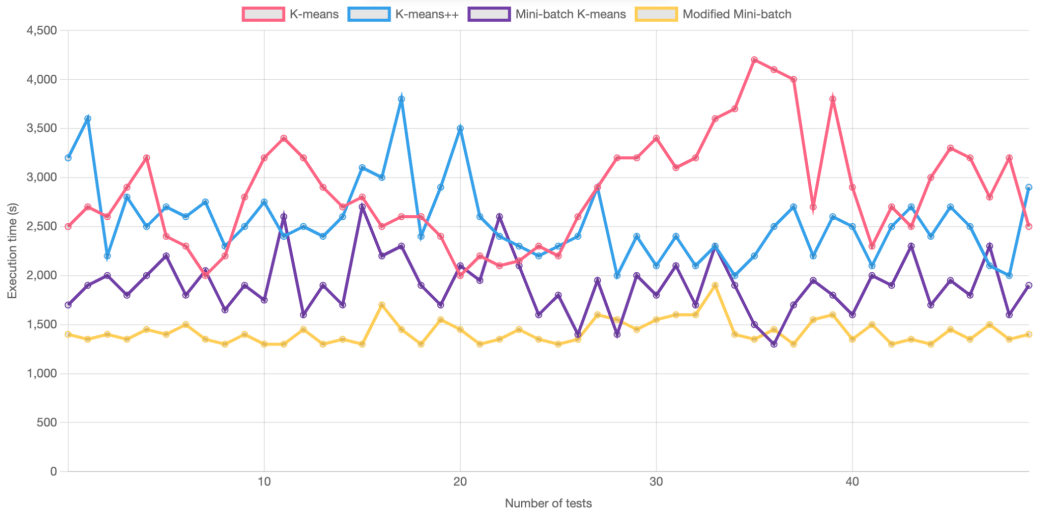

Рис.4.15. Використання пам’яті із використанням хмарних технологій

На основі результатів можна стверджувати, що модель модифікованого алгоритму краще себе показує в реальних застосуваннях та зменшує обчислювальну потужність алгоритму кластеризації K-means.

4.5.2. Результати роботи системи

Основною метою роботи була імплементація системи, яка зможе швидко генерувати профілі користувачів на основі наявної інформації та присвоювати ці профілі кожному користувачу для кращого розуміння бажань та вимог кожного. Профілі користувачів повинні допомогти менеджерам системи краще співпрацювати із клієнтами, у результаті чого рівень задоволеності кожного клієнта повинен стати вищим. Відповідно, оцінки та відгуки користувачів повинні збільшитись (кількісно та якісно).

Після інтеграції методу кластеризації у систему проводилось спостереження щодо кількості залишених відгуків, їх оцінки а також динаміки проведеного часу у системі. Дослідження проводилось із допомогою Google Analytics [96] та власної системи візуалізації інформації. Збір інформації проводився пасивно із допомогою медіатора Google Tag Manager.

Протягом року роботи системи для створення профілів користувачів було зауважено, що кількість відгуків збільшилась приблизно на 30.62% порівняно із попереднім роком. Потрібно відзначити, що низька активність у першому кварталі року та тенденція до зниження активності у четвертому кварталі пов’язана із особливістю ринку нерухомості (на цей період припадають свята, велика кількість користувачів притримує рішення до настання весни чи літа).

На рис 4.16. графічно представлено динаміку зміни кількості користувачів.

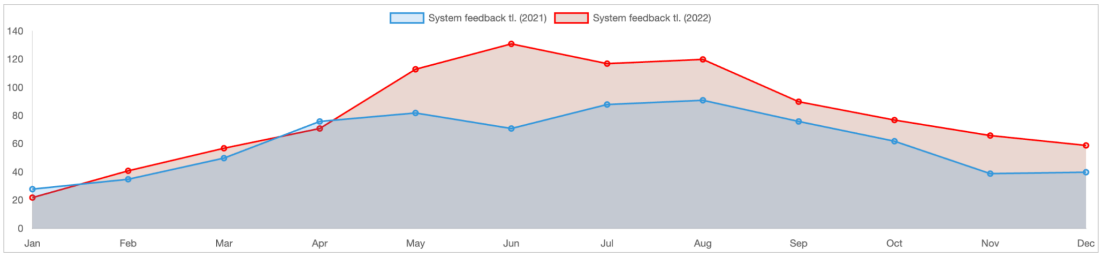

Рис. 4.16. Динаміка кількості відгуків користувачів

Далі було проаналізовано співвідношення позитивних, нейтральних та негативних відгуків у системі. На рис. 4.17. зображено щомісячне порівняння кількості таких відгуків.

Рис. 4.17. Співвідношення позитивних, нейтральних та негативних відгуків

Якщо порівняти їх у відсотковому співвідношенні, то можна побачити, що протягом року кількість позитивних відгуків збільшилась на 5.21% з 54.33% до 59.54%. Кількість нейтральних зменшилась приблизно на 4.06% з 10.43% до 6.36%. Кількість негативних відгуків також зменшилась на 1.15% з 35.23% до 34.08%. Візуально співвідношення представлені на рис.4.18.

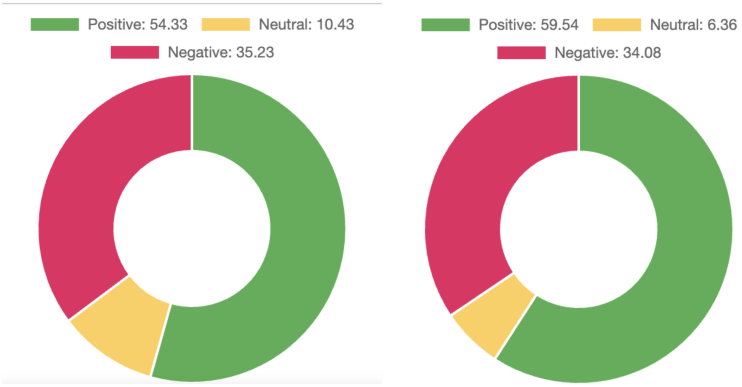

Рис.4.18. Відсоткове співвідношення відгуків

З графіків та даних можна зробити висновки, що кількість нейтральних відгуків зменшилась на користь позитивних відгуків. Це означає, що користувачі, стали більш активними та позитивно оцінюють зміни. Це говорить про те, що система профілювання має потенціал та її можна далі розвивати та вдосконалювати.

Щодо тривалості проведення часу у системі – спостереження показали, що тривалість збільшилась на декілька хвилин (4 хвилини 37 секунд) (рис. 4.19. – 4.20.). Щодо коефіцієнту відмов – невелике збільшення сигналізує, що відсоток користувачів які користуються системою швидше знаходили те що їм потрібно (рис. 4.19. – 4.20.).

Рис.4.19. Середній час сесії та коефіцієнт відмов до впровадження системи

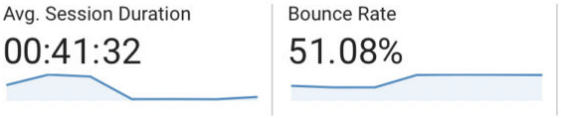

Рис.4.20. Середній час сесії та коефіцієнт відмов після впровадження системи

4.6. Висновки до розділу 4

В цьому розділі дисертаційного дослідження була розроблена архітектура інформаційної системи для здійснення автоматизованого профілювання користувачів на основі різнотипових даних клієнтів.

Для зменшення вартості розгортання такої системи, при побудові використовується безсерверний підхід на основі стороннього сервісу Google Cloud Functions.

Впровадження запропонованої архітектури призвело до зниження кінцевої вартості системи в 4 рази порівняно зі стандартною архітектурою, що використовує дроплети.

ВИСНОВКИ

У дисертаційній роботі було розв’язано актуальну наукове завдання розроблення методів та засобів кластеризації різнотипових даних та створення персоналізованих профілів користувачів.

1. Проведено аналіз літератури, які включають у себе порівняльний аналіз методів та засобів кластеризації даних. Проведено аналіз роботи існуючих онлайн-платформ з ринку нерухомості та виявлено ряд нерозв’язаних задач по персоналізованому підходу користувачів, що дало змогу виявити актуальні проблеми та зробити постановку задачі.
2. Побудовано модель профілю клієнта, яка включає поведінкові та психографічні характеристики та дає можливість визначити рівень задоволеності клієнта.
3. Проведено порівняльний аналіз методів кластеризації різнотипових даних на основі набору даних користувачів, зацікавлених у сфері нерухомості.
4. Модифіковано метод кластеризації різнотипових даних, який дозволяє працювати зі структурованими та напівструктурованими даними на основі поділу на пакети, застосування перцентилів та врахування зважування характеристик.
5. Розроблено класифікатор профілів клієнтів, який, на відміну від існуючих рішень, застосовує зважування відгуків на першому етапі та кластеризацію на другому, що у підсумку формує профілі, які дають можливість пришвидшити обслуговування клієнтів.
6. Розроблено архітектуру інформаційної системи та зменшено вартість розгортання системи за допомогою безсерверної архітектури, що дало змогу знизити обчислювальні витрати у 4 рази.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] Galpaya, H. N. Financial Services and Data Heterogeneity: Does Standardization Work?. MIT Sloan School of Management, 2000.

[2] Bleiholder J., Naumann F. Data fusion. ACM Computing Surveys. 2009. Vol. 41, no. 1. P. 1–41. DOI: https://doi.org/10.1145/1456650.1456651.

[3] Han J., Kamber M., Pei J. Data Mining. Elsevier, 2012. DOI: https://doi.org/10.1016/c2009-0-61819-5.

[4] Spotfire | Structured Data: A Foundation for Data Analysis and Business Intelligence. Spotfire. [Онлайновий] Available: https://www.spotfire.com/glossary/what-is-structured-data [Дата звернення: 20 01 2022]

[5] Spotfire | Structured Data: A Foundation for Data Analysis and Business Intelligence. Spotfire. [Онлайновий] Available: https://www.spotfire.com/glossary/what-is-unstructured-data [Дата звернення: 20 01 2022]

[6] Kelleher J. D., MacNamee B., D'Arcy A. Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies. MIT Press, 2015. 624 p.

[7] Zaki M. J., Jr M. W. Data Mining and Analysis: Fundamental Concepts and Algorithms. Cambridge University Press, 2018.

[8] An industrial study on the risk of software changes / E. Shihab et al. the ACM SIGSOFT 20th International Symposium, Cary, North Carolina, 11–16 November 2012. New York, New York, USA, 2012. DOI: https://doi.org/10.1145/2393596.2393670.

[9] Guide to Intelligent Data Analysis / M. R. Berthold et al. London : Springer London, 2010. DOI: https://doi.org/10.1007/978-1-84882-260-3.

[10] Adomavicius G., Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE

Transactions on Knowledge and Data Engineering. 2005. Vol. 17, no. 6. P. 734–749. DOI: https://doi.org/10.1109/tkde.2005.99.

[11] Toward Privacy-Preserving Personalized Recommendation Services / C. Wang et al. Engineering. 2018. Vol. 4, no. 1. P. 21–28. DOI: https://doi.org/10.1016/j.eng.2018.02.005.

[12] Adomavicius G., Tuzhilin A. Context-Aware Recommender Systems. Recommender Systems Handbook. Boston, MA, 2015. P. 191–226. DOI: https://doi.org/10.1007/978-1-4899-7637-6_6.

[13] Berkhin P. A Survey of Clustering Data Mining Techniques. Grouping Multidimensional Data. Berlin/Heidelberg. P. 25–71. DOI: https://doi.org/10.1007/3-540-28349-8_2.

[14] Rokach L., Maimon O. Clustering Methods. Data Mining and Knowledge Discovery Handbook. New York. P. 321–352. DOI: https://doi.org/10.1007/0-387-25465-x_15.

[15] Hamed Taherdoost. Data Collection Methods and Tools for Research; A Step-by-Step Guide to Choose Data Collection Technique for Academic and Business Research Projects. International Journal of Academic Research in Management (IJARM), 2021, 10 (1), pp.10-38. hal-03741847

[16] Dillman D. A., Smyth J. D., Christian L. M. Internet, Phone, Mail, and Mixed-Mode Surveys: The Tailored Design Method. Wiley & Sons, Incorporated, John, 2014.

[17] Kumar V., Tan P.-N., Steinbach M. Introduction to Data Mining. Pearson Education, 2006.

[18] Duda R. O. Pattern classification. 2nd ed. New York : Wiley, 2001. 654 p.

[19] Miradi M., Molenaar A. A. A., van de Ven M. F. C. Knowledge Discovery and Data Mining Using Artificial Intelligence to Unravel Porous Asphalt Concrete in the Netherlands. Intelligent and Soft Computing in Infrastructure Systems Engineering. Berlin, Heidelberg, 2009. P. 107–176. DOI: https://doi.org/10.1007/978-3-642-04586-8_5.

[20] A distribution-based clustering algorithm for mining in large spatial databases / Xiaowei Xu et al. 14th International Conference on Data Engineering, Orlando, FL, USA. DOI: https://doi.org/10.1109/icde.1998.655795.

[21] Kuchaki Rafsanjani M., Asghari Varzaneh Z., Emami Chukanlo N. A Survey Of Hierarchical Clustering Algorithms. Journal of Mathematics and Computer Science. 2012. Vol. 05, no. 03. P. 229–240. DOI: https://doi.org/10.22436/jmcs.05.03.11.

[22] Jain A. K. Data clustering: 50 years beyond K-means. Pattern Recognition Letters. 2010. Vol. 31, no. 8. P. 651–666. DOI: https://doi.org/10.1016/j.patrec.2009.09.011.

[23] Mensouri D., Azmani A., Azmani M. K-Means Customers Clustering by their RFMT and Score Satisfaction Analysis. International Journal of Advanced Computer Science and Applications. 2022. Vol. 13, no. 6. DOI: https://doi.org/10.14569/ijacsa.2022.0130658.

[24] Arthur D., Vassilvitskii S. K-Means++: The Advantages of Careful Seeding. Conference: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, 7–9 January 2007. 2007.

[25] Bouveyron C., Brunet-Saumard C. Model-based clustering of high-dimensional data: A review. Computational Statistics & Data Analysis. 2014. Vol. 71. P. 52–78. DOI: https://doi.org/10.1016/j.csda.2012.12.008.

[26] Bezdek J. C., Ehrlich R., Full W. FCM: The fuzzy c-means clustering algorithm. Computers & Geosciences. 1984. Vol. 10, no. 2-3. P. 191–203. DOI: https://doi.org/10.1016/0098-3004(84)90020-7.

[27] Hwang S., Thill J.-C. Using fuzzy clustering methods for delineating urban housing submarkets. the 15th annual ACM international symposium, Seattle, Washington, 7–9 November 2007. New York, New York, USA, 2007. DOI: https://doi.org/10.1145/1341012.1341031.

[28] Fischer G. User Modeling and User-Adapted Interaction. 2001. Vol. 11, no. 1/2. P. 65–86. DOI: https://doi.org/10.1023/a:1011145532042.

[29] Ahmad A., Dey L. A k-mean clustering algorithm for mixed numeric and categorical data. Data & Knowledge Engineering. 2007. Vol. 63, no. 2. P. 503–527. DOI: https://doi.org/10.1016/j.datak.2007.03.016.

[30] Li C., Biswas G. Unsupervised learning with mixed numeric and nominal data. IEEE Transactions on Knowledge and Data Engineering. 2002. Vol. 14, no. 4. P. 673–690. DOI: https://doi.org/10.1109/tkde.2002.1019208.

[31] Hsu C.-C., Lin S.-H., Tai W.-S. Apply extended self-organizing map to cluster and classify mixed-type data. Neurocomputing. 2011. Vol. 74, no. 18. P. 3832–3842. DOI: https://doi.org/10.1016/j.neucom.2011.07.014.

[32] Hsu C.-C., Chen Y.-C. Mining of mixed data with application to catalog marketing. Expert Systems with Applications. 2007. Vol. 32, no. 1. P. 12–23. DOI: https://doi.org/10.1016/j.eswa.2005.11.017.

[33] Huang Z. Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values. Data Mining and Knowledge Discovery. 1998. Vol. 2, no. 3. P. 283–304. DOI: https://doi.org/10.1023/a:1009769707641.

[34] Chatzis S. P. A fuzzy c-means-type algorithm for clustering of data with mixed numeric and categorical attributes employing a probabilistic dissimilarity functional. Expert Systems with Applications. 2011. Vol. 38, no. 7. P. 8684–8689. DOI: https://doi.org/10.1016/j.eswa.2011.01.074.

[35] David G., Averbuch A. SpectralCAT: Categorical spectral clustering of numerical and nominal data. Pattern Recognition. 2012. Vol. 45, no. 1. P. 416–433. DOI: https://doi.org/10.1016/j.patcog.2011.07.006.

[36] Flach P. A. Machine Learning: The Art and Science of Algorithms That Make Sense of Data. Cambridge University Press, 2012. 409 p.

[37] Reed R. The relationship between house prices and demographic variables. International Journal of Housing Markets and Analysis. 2016. Vol. 9, no. 4. P. 520–537. DOI: https://doi.org/10.1108/ijhma-02-2016-0013.

[38] Profiling and Segmenting Clients with the Use of Machine Learning Algorithms / P. Rymarczyk et al. EUROPEAN RESEARCH STUDIES JOURNAL. 2021. Vol. XXIV, Special Issue 2. P. 513–522. DOI: https://doi.org/10.35808/ersj/2281.

[39] Le-Hoang P. V. Behavior Intention To Purchase Real Estate: An Empirical Study In Ho Chi Minh City. Independent Journal of Management & Production. 2021. Vol. 12, no. 1. P. 080–094. DOI: https://doi.org/10.14807/ijmp.v12i1.1262.

[40] Tilford, Michael Burr. Developing for demand : an analysis of demand segmentation methods and real estate development. 2009.

[41] Altman D. G., Bland J. M. Statistics Notes: Quartiles, quintiles, centiles, and other quantiles. BMJ. 1994. Vol. 309, no. 6960. P. 996. DOI: https://doi.org/10.1136/bmj.309.6960.996.

[42] Sculley D. Web-scale k-means clustering. the 19th international conference, Raleigh, North Carolina, USA, 26–30 April 2010. New York, New York, USA, 2010. DOI: https://doi.org/10.1145/1772690.1772862.

[43] Efficient BackProp / Y. A. LeCun et al. Lecture Notes in Computer Science. Berlin, Heidelberg, 2012. P. 9–48. DOI: https://doi.org/10.1007/978-3-642-35289-8_3.

[44] Bejar, J. «K-means vs Mini Batch K-means: a comparison». 2013 [Онлайновий]. Available: https://upcommons.upc.edu/bitstream/handle/2117/23414/R13-8.pdf [Дата звернення: 18 03 2022]

[45] Jarman Angur Mahmud. Hierarchical cluster analysis: Comparison of single linkage, complete linkage, average linkage and centroid linkage method. Georgia Southern University, 2020.

[46] Ros F., Guillaume S. A hierarchical clustering algorithm and an improvement of the single linkage criterion to deal with noise. Expert Systems with

Applications. 2019. Vol. 128. P. 96–108. DOI: https://doi.org/10.1016/j.eswa.2019.03.031.

[47] Seifoddini H. K. Single linkage versus average linkage clustering in machine cells formation applications. Computers & Industrial Engineering. 1989. Vol. 16, no. 3. P. 419–426. DOI: https://doi.org/10.1016/0360-8352(89)90160-5.

[48] Roux M. Basic Procedures in Hierarchical Cluster Analysis. Eurocourses: Chemical and Environmental Science. Dordrecht, 1991. P. 115–135. DOI: https://doi.org/10.1007/978-94-011-3198-8_4.

[49] Vijaya, Sharma S., Batra N. Comparative Study of Single Linkage, Complete Linkage, and Ward Method of Agglomerative Clustering. 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 14–16 February 2019. 2019. DOI: https://doi.org/10.1109/comitcon.2019.8862232.

[50] Krznaric D., Levcopoulos C. Optimal algorithms for complete linkage clustering in d dimensions. Theoretical Computer Science. 2002. Vol. 286, no. 1. P. 139–149. DOI: https://doi.org/10.1016/s0304-3975(01)00239-0.

[51] Krznaric D., Levcopoulos C. Fast Algorithms for Complete Linkage Clustering. Discrete & Computational Geometry. 1998. Vol. 19, no. 1. P. 131–145. DOI: https://doi.org/10.1007/pl00009332.

[52] Ester, Martin, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Knowledge Discovery and Data Mining. 1996.

[53] Merk A., Cal P., Woźniak M. Distributed DBSCAN Algorithm – Concept and Experimental Evaluation. Advances in Intelligent Systems and Computing. Cham, 2017. P. 472–480. DOI: https://doi.org/10.1007/978-3-319-59162-9_49.

[54] Li Y., Wu H. A Clustering Method Based on K-Means Algorithm. Physics Procedia. 2012. Vol. 25. P. 1104–1109. DOI: https://doi.org/10.1016/j.phpro.2012.03.206.

[55] Ahmed M., Seraj R., Islam S. M. S. The k-means Algorithm: A Comprehensive Survey and Performance Evaluation. Electronics. 2020. Vol. 9, no. 8. P. 1295. DOI: https://doi.org/10.3390/electronics9081295.

[56] Chong B. K-means clustering algorithm: a brief review. Academic Journal of Computing & Information Science. 2021. Vol. 4, no. 5. DOI: https://doi.org/10.25236/ajcis.2021.040506.

[57] Ahmed M., Choudhury N., Uddin S. Anomaly Detection on Big Data in Financial Markets. ASONAM '17: Advances in Social Networks Analysis and Mining 2017, Sydney Australia. New York, NY, USA, 2017. DOI: https://doi.org/10.1145/3110025.3119402.

[58] Ahmed M., Mahmood A. N., Islam M. R. A survey of anomaly detection techniques in financial domain. Future Generation Computer Systems. 2016. Vol. 55. P. 278–288. DOI: https://doi.org/10.1016/j.future.2015.01.001.

[59] Comparative analysis of k-nearest neighbor and modified k-nearest neighbor algorithm for data classification / Okfalisa et al. 2017 2nd International Conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE), Yogyakarta, 1–2 November 2017. 2017. DOI: https://doi.org/10.1109/icitisee.2017.8285514.

[60] Sun S., Chen Q. Hierarchical distance metric learning for large margin nearest neighbor classification. International journal of pattern recognition and artificial intelligence. 2011. Vol. 25, no. 07. P. 1073–1087. DOI: https://doi.org/10.1142/s021800141100897x.

[61] Zhang Z. Introduction to machine learning: k-nearest neighbors. Annals of Translational Medicine. 2016. Vol. 4, no. 11. P. 218. DOI: https://doi.org/10.21037/atm.2016.03.37.

[62] Gajanova L., Nadanyiova M., Moravcikova D. The Use of Demographic and Psychographic Segmentation to Creating Marketing Strategy of Brand Loyalty. Scientific Annals of Economics and Business. 2019. Vol. 66, no. 1. P. 65–84. DOI: https://doi.org/10.2478/saeb-2019-0005.

[63] Mensouri D., Azmani A., Azmani M. K-Means Customers Clustering by their RFMT and Score Satisfaction Analysis. International Journal of Advanced Computer Science and Applications. 2022. Vol. 13, no. 6. DOI: https://doi.org/10.14569/ijacsa.2022.0130658.

[64] Zhou J., Wei J., Xu B. Customer segmentation by web content mining. Journal of Retailing and Consumer Services. 2021. Vol. 61. P. 102588. DOI: https://doi.org/10.1016/j.jretconser.2021.102588.

[65] Zhu, D.S., Lin, C.T., Tsai, C.H., Wu, J.F. A Study on the Evaluation of Customers' Satisfaction-The perspective of Quality, International Journal for Quality Research. 2010. Vol. 4, no. 2, pp. 105-116,

[66] Baquero A. Net Promoter Score (NPS) and Customer Satisfaction: Relationship and Efficient Management. Sustainability. 2022. Vol. 14, no. 4. P. 2011. DOI: https://doi.org/10.3390/su14042011.

[67] Agag G., Eid R. Which consumer feedback metrics are the most valuable in driving consumer expenditure in the tourism industries? A view from macroeconomic perspective. Tourism Management. 2020. Vol. 80. P. 104109. DOI: https://doi.org/10.1016/j.tourman.2020.104109.

[68] Palm P. Measuring customer satisfaction: a study of the Swedish real estate industry. Property Management. 2016. Vol. 34, no. 4. P. 316–331. DOI: https://doi.org/10.1108/pm-08-2015-0041.

[69] Automated variable weighting in k-means type clustering / J. Z. Huang et al. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2005. Vol. 27, no. 5. P. 657–668. DOI: https://doi.org/10.1109/tpami.2005.95.

[70] Jadhav A., Pramod D., Ramanathan K. Comparison of Performance of Data Imputation Methods for Numeric Dataset. Applied Artificial Intelligence. 2019. Vol. 33, no. 10. P. 913–933. DOI: https://doi.org/10.1080/08839514.2019.1637138.

[71] A survey on missing data in machine learning / T. Emmanuel et al. Journal of Big Data. 2021. Vol. 8, no. 1. DOI: https://doi.org/10.1186/s40537-021-00516-9.

[72] Review: A gentle introduction to imputation of missing values / A. R. T. Donders et al. Journal of Clinical Epidemiology. 2006. Vol. 59, no. 10. P. 1087–1091. DOI: https://doi.org/10.1016/j.jclinepi.2006.01.014.

[73] Baraldi A. N., Enders C. K. An introduction to modern missing data analyses. Journal of School Psychology. 2010. Vol. 48, no. 1. P. 5–37. DOI: https://doi.org/10.1016/j.jsp.2009.10.001.

[74] Missing data imputation using statistical and machine learning methods in a real breast cancer problem / J. M. Jerez et al. Artificial Intelligence in Medicine. 2010. Vol. 50, no. 2. P. 105–115. DOI: https://doi.org/10.1016/j.artmed.2010.05.002.

[75] Huang, Yu and Fei Chiang. Refining Duplicate Detection for Improved Data Quality. TDDL/MDQual/Futurity@TPDL. 2017.

[76] Elmagarmid A. K., Ipeirotis P. G., Verykios V. S. Duplicate Record Detection: A Survey. IEEE Transactions on Knowledge and Data Engineering. 2007. Vol. 19, no. 1. P. 1–16. DOI: https://doi.org/10.1109/tkde.2007.250581.

[77] Ahmed S. T., George L. E. Lightweight hash-based de-duplication system using the self detection of most repeated patterns as chunks divisors. Journal of King Saud University - Computer and Information Sciences. 2021. DOI: https://doi.org/10.1016/j.jksuci.2021.04.005.

[78] «function MD 5,» [Онлайновий]. Available: https://www.md5.cz/. [Дата звернення: 20 12 2022].

[79] «SECURE HASH STANDARD,» [Онлайновий]. Available: https://csrc.nist.gov/files/pubs/fips/180-2/final/docs/fips180-2.pdf/. [Дата звернення: 20 12 2022].

[80] Singh D., Singh B. Investigating the impact of data normalization on classification performance. Applied Soft Computing. 2020. Vol. 97. P. 105524. DOI: https://doi.org/10.1016/j.asoc.2019.105524.

[81] Aksu G., Güzeller C. O., Eser M. T. The Effect of the Normalization Method Used in Different Sample Sizes on the Success of Artificial Neural Network Model. International Journal of Assessment Tools in Education. 2019. P. 170–192. DOI: https://doi.org/10.21449/ijate.479404.

[82] Z-Score Normalization, Hubness, and Few-Shot Learning / N. Fei et al. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021. 2021. DOI: https://doi.org/10.1109/iccv48922.2021.00021.

[83] A Review on Data Preprocessing Techniques Toward Efficient and Reliable Knowledge Discovery From Building Operational Data / C. Fan et al. Frontiers in Energy Research. 2021. Vol. 9. DOI: https://doi.org/10.3389/fenrg.2021.652801.

[84] Feature dimensionality reduction: a review / W. Jia et al. Complex & Intelligent Systems. 2022. DOI: https://doi.org/10.1007/s40747-021-00637-x.

[85] Class definition in discriminant feature analysis / J. Duchateau et al. 7th European Conference on Speech Communication and Technology (Eurospeech 2001). ISCA, 2001. DOI: https://doi.org/10.21437/eurospeech.2001-197.

[86] Singular Value Decomposition (SVD). Data-Driven Science and Engineering. 2019. P. 3–46. DOI: https://doi.org/10.1017/9781108380690.002.

[87] Mahmud M. S., Rahman M. M., Akhtar M. N. Improvement of K-means clustering algorithm with better initial centroids based on weighted average. 2012 7th International Conference on Electrical & Computer Engineering (ICECE), Dhaka, Bangladesh, 20–22 December 2012. 2012. DOI: https://doi.org/10.1109/icece.2012.6471633.

[88] Dotsika F. Semantic APIs: Scaling up towards the Semantic Web. International Journal of Information Management. 2010. Vol. 30, no. 4. P. 335–342. DOI: https://doi.org/10.1016/j.ijinfomgt.2009.12.003.

[89] Al-Debagy O., Martinek P. A Comparative Review of Microservices and Monolithic Architectures. 2018 IEEE 18th International Symposium on Computational Intelligence and Informatics (CINTI), Budapest, Hungary, 21–22 November 2018. 2018. DOI: https://doi.org/10.1109/cinti.2018.8928192.

[90] Stonebraker M., Rowe L. A., Hirohama M. The implementation of POSTGRES. IEEE Transactions on Knowledge and Data Engineering. 1990. Vol. 2, no. 1. P. 125–142. DOI: https://doi.org/10.1109/69.50912.

[91] Vidhya R., Vadivu G. Research Document Search using Elastic Search. Indian Journal of Science and Technology. 2016. Vol. 9, no. 37. DOI: https://doi.org/10.17485/ijst/2016/v9i37/102108.

[92] Towards Scalable and Reliable In-Memory Storage System: A Case Study with Redis / S. Chen et al. 2016 IEEE Trustcom/BigDataSE/ISPA, Tianjin, China, 23–26 August 2016. 2016. DOI: https://doi.org/10.1109/trustcom.2016.0255.

[93] Exposito Jimenez V. J., Zeiner H. Serverless Cloud Computing : A Comparison Between "Function as a Service" Platforms. 7th International Conference on Information Technology Convergence and Services. 2018. DOI: https://doi.org/10.5121/csit.2018.80702.

[94] Dunn’s index for cluster tendency assessment of pharmacological data sets / O. M. Rivera-Borroto et al. Canadian Journal of Physiology and Pharmacology. 2012. Vol. 90, no. 4. P. 425–433. DOI: https://doi.org/10.1139/y2012-002.

[95] Rousseeuw P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics. 1987. Vol. 20. P. 53–65. DOI: https://doi.org/10.1016/0377-0427(87)90125-7.

[96] Galbraith S. Google Analytics. Journal of the Canadian Health Libraries Association / Journal de l'Association des bibliothèques de la santé du Canada. 2014. Vol. 34, no. 2. P. 119. DOI: https://doi.org/10.5596/c13-022.

[97] Бойко Н.І., Ткачик О.А. Оцінка методів кластеризації різнотипових даних. Journal Automation of technological and business –processes, 2023. Vol. 15(1), pp. 1-12. https://doi.org/10.15673/atbp.v15i1.2508.

[98] Boyko N., Tkachyk O. Model for Finding Frequent Sets in FP-growth for Multimodal Data, in: Proceedings of The Fifth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2022), Zaporizhzhia, Ukraine, May 12, 2022, pp.126-143. https://doi.org/10.327 82/cmis/3137-11

[99] Mytnyk B., Tkachyk O., Shakhovska N., Fedushko S., Syerov Yu. Application of Artificial Intelligence for Fraudulent Banking. Big Data Cogn. Comput. 2023, 7(2), 93. https://doi.org/10.3390/bdcc7020093 (квартиль Q1 у НМБД Scopus)

[100] Havano B., Kytsun H., Tkachyk O. Web-server cross-site request forgery protection, in: VII International Youth Conference "Perspectives of Science and Education", 2020 New York, USA, pp. 9–16. https://doi.org/10.29013/VII-Conf-USA-7-9-16.

[101] Boyko N., Tkachyk O. Frequency pattern growth algorithm (FP) for multimodal data extraction, in: Proceedings of the 3rd International Workshop on Intelligent Information Technologies & Systems of Information Security Khmelnytskyi, Ukraine, March 23–25, 2022, pp. 72-82.

[102] Ткачик О. Застосування методів кластеризації даних для створення цільових груп користувачів на ринку нерухомості. Вісник Хмельницького національного університету, 2023. Том 2 (319), С. 300-307. https://www.doi.org/10.31891/2307-5732-2023-319-1-300-307

[103] Бойко Н. І., Ткачик О. А. Алгоритми та методи кластеризації для різноманітних даних. Науковий вісник Ужгородського університету. Серія «Математика і інформатика» / редкол.: М. М. Маляр (гол. ред.) та інші.

Ужгород: Видавництво УжНУ «Говерла», 2023. Т. 42, No 1. С. 131-150. DOI: https://www.doi.org/10.24144/2616-7700.2023.42(1).129-147

[104] State of the Connected Customer Report,» [Онлайновий]. Available: https://www.salesforce.com/resources/research-reports/state-of-the-connected-customer/. [Дата звернення: 20 12 2022].

[105] Hermann E. Artificial intelligence and mass personalization of communication content–An ethical and literacy perspective. New Media & Society. 2021. P. 146144482110227. DOI: https://doi.org/10.1177/14614448211022702.

[106] Cappella J. N. Vectors into the Future of Mass and Interpersonal Communication Research: Big Data, Social Media, and Computational Social Science. Human Communication Research. 2017. Vol. 43, no. 4. P. 545–558. DOI: https://doi.org/10.1111/hcre.12114.

[107] A Machine Learning Approach to Personalize Computerized Cognitive Training Interventions / M. Vladisauskas et al. Frontiers in Artificial Intelligence. 2022. Vol. 5. DOI: https://doi.org/10.3389/frai.2022.788605.

[108] Gao Y., Liu H. Artificial intelligence-enabled personalization in interactive marketing: a customer journey perspective. Journal of Research in Interactive Marketing. 2022. P. 1–18. DOI: https://doi.org/10.1108/jrim-01-2022-0023.

ДОДАТОК А. АКТИ ВПРОВАДЖЕННЯ