

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

факультет інформатики та обчислювальної техніки

(повна назва інституту/факультету)

кафедра автоматика та управління в технічних системах

(повна назва кафедри)

«На правах рукопису»

УДК 004.93

«До захисту допущено»

Завідувач кафедри

О. І. Ролік

(підпис) (ініціали, прізвище)

“ ” _____ 2018 р.

Магістерська дисертація

зі спеціальності (спеціалізації) 126 «Інформаційні системи та технології»

(код і назва спеціальності)

на тему: Інтелектуальна система компаративної оцінки клієнтських уподобань

Виконав : студент 6 курсу, групи ІА-73мп

(шифр групи)

Бельдяга Кирило Дмитрович

(прізвище, ім'я, по батькові)

(підпис)

Науковий керівник доцент, к.т.н. Дорогий Я.Ю.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Консультант _____

(назва розділу)

(науковий ступінь, вчене звання, прізвище, ініціали)

(підпис)

Рецензент _____

(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____

(підпис)

Київ – 2018 року

Національний технічний університет України
“Київський політехнічний інститут
імені Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки
(повна назва)
Кафедра автоматичного управління в технічних системах
(повна назва)
Ступінь вищої освіти – другий (магістерський)
(код, назва)
Спеціальність 126 «Інформаційні системи та технології»
(код, назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ О. І. РОЛІК
(підпис) (ініціали, прізвище)

“ ___ ” _____ 2018 р.

ЗАВДАННЯ

на магістерську дисертацію студенту

Бельдязі Кирилу Дмитрович
(прізвище, ім'я, по батькові)

1. **Тема дисертації** Інтелектуальна система компаративної оцінки клієнтських уподобань

Науковий керівник дисертації Дорогий Ярослав Юрійович, к.т.н., доцент

затверджені наказом по університету від “ 29 ” жовтня 2018 р. № _____

2. Строк подання студентом дисертації “ 4 “ грудня 2018 р.

3. Об'єкт дослідження: системи аналізу клієнтських уподобань а

4. Зміст пояснювальної записки: а) огляд існуючого стану предметної області; б) синтез моделі та алгоритму кластеризації; в) технічне рішення та реалізація; г) оцінка якості системи; д) розробка стартап-проекту.

5. Перелік завдань, які потрібно розробити: проаналізувати предметну область, існуючі рішення для побудови систем аналізу клієнтів, розробити моделі та обрати алгоритми, розробити архітектуру системи, провести оцінку роботи системи, протестувати розроблену систему, провести маркетинговий аналіз стартап проекту.

6. Перелік графічного (ілюстративного) матеріалу: структурна схема моделі для аналізу клієнтських уподобань, варіації моделі, діаграма варіантів використання, структурна схема, функціональна схема, схема алгоритмів роботи, діаграма послідовності, діграма розгортання.

4. Орієнтовний перелік публікацій: _____

5. Консультанти розділів проекту:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

6. Дата видачі завдання “ 29 ” жовтня 2018 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів проекту	Примітка
1	Огляд існуючого стану проблемної області	10.11.2018	
2	Синтез моделі та вибір алгоритму кластеризації	15.11.2018	
3	Технічне рішення та реалізація	20.11.2018	
4	Оцінка якості системи	25.11.2018	
5	Розробка стартап-проекту	29.11.2018	
6	Оформлення текстової та графічної документації	2.12.2018	
7	Представлення до захисту	4.12.2018	

Студент

(підпис)

Бельдяга К.Д.
(ініціали, прізвище)

Науковий керівник дисертації

(підпис)

Дорогий Я.Ю.
(ініціали, прізвище)

РЕФЕРАТ

Магістерська дисертація: 115 с., 25 рис., 50 табл., 9 додатків, 19 джерел.

Тема магістерської дисертації «Інтелектуальна система компаративної оцінки клієнтських уподобань».

Підґрунтям актуальності теми даної магістерської дисертації є швидкі темпи впровадження компаніями у свої системи аналітики та управління відносинами з клієнтами для формування прихованих груп клієнтів та прогнозування ознак та уподобань систем на базі алгоритмів машинного навчання, щоб збільшити прибуток та оптимізувати витрати.

Об'єктом дослідження є системи аналізу клієнтських уподобань.

Предметом дослідження є алгоритми та нейронні мережі, що використовуються для аналізу клієнтських уподобань.

Метою роботи є підвищити ефективність пошуку прихованих груп клієнтів та збільшити точність прогнозування уподобань і ознак ймовірних клієнтів та запропонувати підходи для вибору алгоритмів та моделей для вирішення проблем прогнозування.

Результатом виконання даної кваліфікаційної роботи є запропонована математична модель клієнта на основі нейронної мережі, що дає можливість прогнозувати уподобання та ознаки клієнта на основі прихованих особливостей та навпаки, а також програмне забезпечення – інтелектуальна система компаративної оцінки клієнтських уподобань.

Результати та дослідження, проведені під час роботи, були використані в системі, що проваджена в експлуатацію, що підтверджує практичну цінність одержаних результатів.

**НЕЙРОННІ МЕРЕЖІ, ПРОГНОЗУВАННЯ, КЛІЄНТСЬКІ УПОДОБАННЯ,
МАШИННЕ НАВЧАННЯ, АНАЛІЗ КЛІЄНТІВ, АНАЛІЗ СПОЖИВАЧІВ.**

ABSTRACT

Master's thesis: 115 p., 25 figures, 50 tables, 9 appendixes, 19 sources.

Theme of the master's thesis “Intelligent system for comparative assessment of client’s preferences”.

The reason for the relevance of the topic of this master's thesis is the rapid pace of implementing solutions and services based on machine learning algorithms that can predict user’s preferences and features in order to increase profit and optimize costs. These are getting implemented by companies for their customer relationship management and analytic systems.

The object of research: client’s preferences analysis systems.

The subject of the research are the algorithms and neural networks that are used in client’s preferences analysis.

The aim of the work is to increase the effectiveness of search of hidden groups of clients and accuracy of preferences and features prediction and to offer an approach for choosing algorithms and models for solving prediction problems.

The result of this qualification work is the proposed mathematical model of the client based on neural network that allows to predict client preferences and features with hidden features and vice versa, and the software system – an intellectual system for comparative assessment of client’s preferences. The results of development and research work were used in the development of the system being put into operation, which confirms the practical value of the results.

NEURAL NETWORKS, PREDICTION, CLIENT PREFERENCES, MACHINE LEARNING, CLIENT ANALYSIS, CUSTOMER ANALYSYS.

ЗМІСТ

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ	8
ВСТУП	9
1 ОГЛЯД ІСНУЮЧОГО СТАНУ ПРОБЛЕМНОЇ ОБЛАСТІ	10
1.1 Аналіз предметної області	10
1.2 Огляд існуючих рішень	13
1.3 Формування вимог до системи	14
1.4 Теоретичні відомості	15
1.4.1 Нейронні мережі	15
1.4.2 Кластеризація	22
1.5 Висновки	22
2 СИНТЕЗ МОДЕЛІ ТА ВИБІР АЛГОРИТМУ КЛАСТЕРИЗАЦІЇ	23
2.1 Моделювання клієнта	24
2.1.1 Синтез моделі	24
2.1.2 Варіації моделі клієнта	34
2.2 Кластеризація клієнтів	37
2.3 Висновки	40
3 ТЕХНІЧНЕ РІШЕННЯ ТА РЕАЛІЗАЦІЯ	41
3.1 Обґрунтування вибору архітектури	41
3.2 Сценарії використання системи	43
3.3 Розробка структурної схеми	64
3.4 Розробка функціональної схеми	68
3.5 Інструменти, мови та засоби реалізації	71
3.5.1 Вибір мови програмування системи	71
3.5.2 Вибір мови програмування для досліджень та навчання	73
3.5.3 Вибір сховища для зберігання даних	73
3.5.4 Вибір брокера повідомлень	74
3.6 Алгоритми роботи	75
3.7 Діаграма послідовності	76
3.8 Розгортання системи	77
3.9 Тестування системи	78

3.10 Висновки	82
4 ОЦІНКА ЯКОСТІ СИСТЕМИ	83
4.1 Технічна оцінка	83
4.2 Оцінка опитуванням	85
4.3 Висновки	85
5 РОЗРОБКА СТАРТАП ПРОЕКТУ	86
5.1 Опис ідеї проекту	86
5.2 Технологічний аудит ідеї проекту	88
5.3 Аналіз ринкових можливостей запуску	90
5.4 Розроблення ринкової стратегії проекту	97
5.5 Розроблення маркетингової програми	99
5.6 Висновки	103
ВИСНОВКИ	104
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	105
ДОДАТОК А	107
ДОДАТОК Б	108
ДОДАТОК В	109
ДОДАТОК Г	110
ДОДАТОК Д	111
ДОДАТОК Е	112
ДОДАТОК Ж	112
ДОДАТОК К	114
ДОДАТОК Л	115

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

НМ — нейронна мережа являє собою систему, що складається з певної кількості нейронів та моделює принцип сприйняття інформації мозком.

ШНМ — штучна нейронна мережа є штучною моделлю нейронної мережі, яка була створена неприродним шляхом.

ЕМ — енергетичні моделі є особливим видом нейронних мереж, в основі яких лежить поняття енергії системи при її конфігурації.

МБ — машина Больцмана є різновидом енергетичних систем, відмінністю від яких є наявність шару прихованих вузлів.

ОМБ — обмежена машина Больцмана являє собою машину Больцмана, у якої немає зв'язків між елементами одного шару.

CD (англ. Contrastive Divergence) — порівняльна розбіжність представляє собою алгоритм оптимізації функції втрат і застосовується в тренуванні обмежених машин Болцьмана.

ВСТУП

З кожним днем зростає кількість компаній. В той самий час, компанії пропонують свої товари, продукти та послуги задля того, щоб збільшити прибуток, і, таким чином, заохочуючи все більше і більше клієнтів. Однак, через надмірну кількість клієнтів виникають нові групи, що не приносять прибутку компанії, проте остання витрачає ресурси на неї.

Клієнти обирають продукти та послуги, які вони потребують. Таким чином, можна зазначити, що клієнтські уподобання – це тип продукту або послуги, якому клієнт віддає перевагу. Отже, вирішити проблему появи нерентабельних груп та пошуку нових можна базуючись на уподобаннях клієнта. Вони утворюються внаслідок визначення схожих ознак, уподобань. Проте існують приховані особливості, що характеризують клієнта як особистість, що не можливо визначити традиційними способами сегментації користувачів. Тому існує потреба в розробці системи, яка розв'язала задачу розбиття вибірки клієнтів на групи, базуючись на латентних факторах і, в той самий час, дозволяла порівнювати клієнтів за цими факторами. Такий підхід дасть можливість компаніям досліджувати своїх споживачів та визначати ймовірні групи, що принесуть більше прибутку і будуть зацікавлені в продуктах або послугах компанії. Завдяки можливості прогнозування уподобань, така система також може бути використана в якості рекомендаційної системи.

Аналіз клієнтів є одним із розділів аналітики споживачів, мета якої прийняти рішення щодо, наприклад, стратегії розвитку бізнесу, базуючись на поведінці клієнтів. Використовувати суміжні підходи для досягнення мети.

1 ОГЛЯД ІСНУЮЧОГО СТАНУ ПРОБЛЕМНОЇ ОБЛАСТІ

1.1 Аналіз предметної області

Аналіз клієнтських уподобань, ознак та груп відноситься до розділу аналітики споживачів[1], що є процесом, яким поведінка споживачів допомагає виконувати рішення та прогнозувати події стосовно бізнесу. Цей підхід використовується в роздрібній торгівлі, фінансах, онлайн-маркетплейсах. Мета аналітики користувачів розробити єдиний точний вид споживача, для рішень про залучення та пошук нових сегментів споживачів.

Компаративна оцінка клієнтських уподобань означає порівняння клієнтів за ознаками та уподобаннями. Виконати компаративну оцінку можна багатьма способами, найпростішим з яких є порівняння уподобань клієнтів між собою, на основі даних про те. Дані про продукти чи послуги можуть бути представлені багатьма способами. Наприклад, кількістю покупок того чи іншого товару або частотою використання послуг. Якщо розглядати уподобання як, наприклад, медіа-контент, то тут вони можуть бути як і частотою прослуховування так і рейтинговим оцінюванням. Однак, інформація про уподобання не розкриває поведінку та характер клієнта. Тому, для порівняння необхідно використовувати декілька підходів одночасно.

Оскільки основною сферою застосування досліджуваної системи є впровадження в існуючі системи аналітики[2] та управління ресурсами компанії. Це означає, що необхідно визначити серед всієї вибірки клієнтів схожих, як за ознаками, так і уподобаннями.

Пошук схожих об'єктів є завданням кластерного аналізу[3]. Область його застосувань широка: маркетинг, виявлення спаму та шахрайства в грошових переказах, аналіз текстів, повідомлень, сегментація зображень, прогнозування, виявлення шаблонів, прогнозування, тощо. Основним підходом в роботі кластеризації є утворення кластерів, що являють собою згустки об'єктів, які відносяться до спільного класу або групи, усередині кожної повинні виявитися схожі об'єкти, а різні повинні бути якомога більш відмінні. В контексті даної системи, застосування такого підходу дозволяє

виділяти сегменти клієнтів[4], споживачів за їхніми уподобаннями, ознаками та розроблювати окрему маркетингову стратегію для підвищення прибутку.

Дані, що необхідно кластеризувати мають наступні особливості:

- висока розмірність(тисячі атрибутів) і великий об'єм (мільйони записів) баз та сховищ даних;
- дані можуть бути неоднорідними, тобто представлені, як кількісними, так і якісними значеннями.

Вхідними даними можуть бути як матриця відстані, що описує відстані до всіх об'єктів метричного простору, так і матриця зіставлення, яка враховує ступінь збігу об'єкта з іншими в метричному просторі.

Застосування кластерного аналізу в загальному вигляді зводиться до наступних етапів:

- відбір вибірки об'єктів для кластеризації;
- визначення безлічі змінних, за якими будуть оцінюватися об'єкти у вибірці. При необхідності – нормалізація значень змінних;
- обчислення значень міри схожості між об'єктами.
- застосування методу кластерного аналізу для створення груп схожих об'єктів (кластерів).
- представлення результатів аналізу.

Для порівняння об'єктів використовується міра відстані, яка характеризує ступінь схожості. Існує велика кількість метрик, основні з яких:

- евклідова відстань є геометричною відстанню в багатовимірному просторі між точками;
- квадрат евклідової відстані застосовується для додання більшої ваги більш віддаленим один від одного об'єктів;
- відстань міських кварталів є середньою різницею по координатах, тому великі різниці по осям менше впливають на відстань;
- відстань Чебишева визначає різницю з однією координатою;
- степенева відстань застосовується в разі, коли необхідно збільшити або зменшити вагу ознаки, для якої відповідні об'єкти сильно відрізняються.

Вищеописані метрики застосовуються для порівняння об'єктів з кількісними атрибутами, а для категоріальних поширена міра схожості Чекановського-Серенсена та Жакара.

$$(|t_1 \cap t_2|/|t_1 \cup t_2|). \quad (1.1)$$

Вибір алгоритму кластеризації залежить від типу задачі, однак їх можна класифікувати за деякими критеріями. За можливістю застосування в обмеженій пам'яті: масштабовані та немасштабовані. За способом розбиття на кластери – ієрархічні та неієрархічні. Перші забезпечують високу якість групування та не вимагають попереднього задання кількості класів, будуючи дерево вкладеності кластерів. Останні базуються на оптимізації цільової функції, що визначає оптимальне розбиття множини. Використання генетичних алгоритмів значно збільшує складність алгоритму. До них відносять: ітеративні, щільні, модельні, концептуальні та мережеві. Прикладами є k-means, EM. Проблему з неоднорідними ознаками дозволяють вирішити різні модифікації алгоритмів такі як, k-modes[5], k-prototypes[6]. В свою чергу, scalable k-means і scalable EM, CUDA, CACTUS потребують декількох разів для звернення до сховища даних для отримання кластеризації. Таким чином, можна прийти до висновку, що не існує єдиного універсального алгоритму кластеризації.

Компаративну оцінку можна виконати також і за допомогою кластеризації, визначивши найближчу групу або клієнта, можна чим відрізняються розглядувані клієнти.

Частково або повністю розв'язують задачу рекомендаційні системи та колаборативна фільтрація, порівнюючи об'єкт з усіма наявними в вибірці, однак не вирішують проблему компаративної оцінки.

1.2 Огляд існуючих рішень

Існуючих рішень с аналізу клієнтських уподобань, базуючись на прихованих факторах у відкритому доступі немає. Дослідження ринку показало, що компанії використовують системи управління ресурсами з розширеннями для аналітики споживачів. Однак всі методи базуються на порівнянні або за ознаками, або за уподобаннями, уникаючи аналізу глибоких залежностей. Багато компаній розроблюють власні системи аналізу та не використовують їх як джерело додаткового прибутку. Тому розробка такої системи буде доцільною, аби задовольняти потреби багатьох компаній.

Оскільки аналіз клієнтів є розділом аналізу споживачів, було досліджено його ринок. В результаті було встановлено[7], що на ринку аналізу споживачів серед лідерів можна виділити декілька рішень та засобів. Adobe Analytics[8], Google Analytics 360, IBM Watson Customer Experience Analytics[9], SAP Hybris Marketing Cloud[10], SAS Customer Intelligence 360.

Особливості Adobe Analytics:

- платформа для відео та рекламної взаємодії;
- середовище прогнозування за допомогою машинного навчання;
- мобільний додаток для компаній;
- середовище для роботи з даними.

Особливості SAP Hybris Marketing Cloud:

- підсилює розуміння споживачів;
- категоризує споживачів;
- формує цільові сегменти споживачів;

Також виділити рішення CustomerAnalytics[11], що може бути інтегровано в систему компаній та аналізувати дані про споживачів за допомогою машинного навчання та інших алгоритмів.

Проте всі перелічені рішення мають надмірно велику кількість функцій та не виконують аналіз клієнтів за уподобаннями в повній мірі.

1.3 Формування вимог до системи

Провівши аналіз предметної області, проаналізувавши існуючі рішення, дані клієнтів, які буде аналізовано в системі, а також оцінивши вимоги систем, в сервіси якої буде впроваджено дану систему, як частину системи аналітики та системи управління ресурсами, необхідно сформувати вимоги до системи, що забезпечить її якість та конкурентноспроможність.

Функціональні вимоги:

- а) менеджмент користувачів;
- б) авторизацію;
- в) менеджмент ролей;
- г) менеджмент клієнтів;
- г) можливість отримувати інформацію про клієнтів;
- д) можливість отримувати інформацію про утворені групи клієнтів;
- е) можливість компаративну оцінку клієнтів;
- є) можливість прогнозування ймовірних ознак клієнта, маючи інформацію про уподобання і навпаки;
- ж) можливість прогнозування ймовірних уподобань, маючи інформацію про ознаки клієнта;
- з) можливість прогнозування прихованих особливостей ймовірного клієнта на основі його уподобань та ознак і навпаки.

Нефункціональні вимоги:

- Масштабованість;
- Швидко реагувати на запити з інших сервісів системи, в яку впроваджено дану інтелектуальну систему;
- Можливість роботи на всіх платформах;
- Простий процес розгортання системи.

1.4 Теоретичні відомості

1.4.1 Нейронні мережі

1.4.1.1 Функції активації

Основною складовою штучної нейронної мережі є нейрон. В залежності від випадків, в яких використовується НМ, визначається функція активації, що обчислює вихід нейрона.

Найпростішою функцією активації є функція Хевісайда, область значень якої $E(f) = \{0,1\}$, яка представляється виразом

$$y = \begin{cases} 0, & S \leq 0 \\ 1, & S > 0 \end{cases} \quad (1.2)$$

де y – вихід нейрона;

S – деяка функція, що обчислює значення нейрону.

Обмежена область значень є основним недоліком нейрона такого типу, що може бути усунений зміною функції активації на деяку функцію сигма, що представлена в формулі

$$\sigma(S) = \frac{1}{(1+e^{-S})}. \quad (1.3)$$

Тобто область значень даної функції є $E(\sigma) = [0; 1]$. При умові $S \gg 0$, то $e^{-S} \approx 0$, а $\sigma(S) \approx 1$, і навпаки при умові, коли $S \ll 0$, то $e^{-S} \approx \infty$, а $\sigma(S) \approx 0$.

В деяких видах НМ застосовується логарифмічна функція, яка виражається за виразом

$$y = \ln(1 + e^S) = \text{softplus}(s). \quad (1.4)$$

1.4.1.2 Обмежена машина Больцмана

Моделі, що базуються на енергії, або енергетичні моделі (ЕМ) визначають стан системи, обраховуючи скалярну енергію до кожної з її конфігурацій. Існує така функція енергія станів системи, при якій система еволюціонує в один з низькоенергетичних станів. Даний принцип бере початок з статистичної механіки, в якій розглядається система, що може знаходитися в одному зі станів з певною ймовірністю, а кожному зі станів відповідає енергія системи. Стани з меншою енергією мають більшу ймовірність переходу в відповідний стан.

Обмежена машина Больцмана[12] (ОМБ) – породжувальна неорієнтована модель, окремий тип машин Больцмана, для якої вводяться обмеження на зв'язки між змінними одного шару, тобто відсутні зв'язки “видимий-видимий” та “прихований-прихований”. Відсутність зв'язків між змінними одного шару дозволяє ефективніше визначати розподіл значень на деякому векторі \mathbf{v} видимих змінних, використовуючи приховані \mathbf{h} .

Енергетичним моделям притаманна енергетична функція. Для ОБМ виражається наступною формулою

$$\begin{aligned} E(\mathbf{v}, \mathbf{h}) &= -\mathbf{b}^T \mathbf{h} - \mathbf{c}^T \mathbf{v} - \mathbf{h}^T \mathbf{W} \mathbf{v} = \\ &= -\sum_j \sum_i W_{ij} h_j v_i - \sum_i c_i v_i - \sum_j b_j h_j, \end{aligned} \quad (1.5)$$

де \mathbf{W} — матриця вагових коефіцієнтів між видимим та прихованим шарами;

\mathbf{b} – зсув прихованого шару;

\mathbf{c} – зсув видимого шару.

Ймовірнісний розподіл у моделі ОМБ визначається за виразом

$$p(\mathbf{v}, \mathbf{h}) = \frac{e^{-E(\mathbf{v}, \mathbf{h})}}{Z}, \quad (1.6)$$

де Z – статична сума визначена як

$$Z = \sum_{\mathbf{v}, \mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}. \quad (1.7)$$

Проаналізувавши вирази (1.5) та (1.6) можна зробити певні висновки. Для того, щоб ймовірність конфігурації була великою, тобто, енергія системи була нижчою, необхідно аби кожен із членів виразу (1.5) прагнув до одиниці. Тобто, у випадку, коли \mathbf{c}^T набуває від'ємних значень \mathbf{v} має дорівнювати одиниці і, навпаки, коли значення \mathbf{c}^T більше нуля змушує \mathbf{v} бути нулем. Аналогічні правила застосовуються і для виразу $\mathbf{b}^T \mathbf{h}$. Для останнього члена цього виразу, бажане значення застосовується до обох значеннях \mathbf{h} та \mathbf{v} одночасно.

Для ОМБ також притаманна функція вільної енергії, яка означає енергію по всім можливим конфігураціям \mathbf{h} , тобто енергію, яка не залежить від стану прихованої частини моделі, що визначається виразом

$$F(\mathbf{v}) = -\mathbf{b}^T \mathbf{v} - \sum_i \log \sum_{h_i} e^{h_i(c_i + W_i \mathbf{v})}. \quad (1.8)$$

Завдяки своїй структурі, ОМБ дозволяє шарам бути умовно незалежними один від одного, як це показано на виразах

$$p(\mathbf{h}|\mathbf{v}) = \prod_i p(h_i|\mathbf{v}), \quad (1.9)$$

$$p(\mathbf{v}|\mathbf{h}) = \prod_j p(v_j|\mathbf{h}). \quad (1.10)$$

Легко помітити, що при умові $h_j = 0$, умовний розподіл ймовірностей спростовується, як показано виразом (1.9), а при умові $h_j = 1$ визначається за виразом (1.10).

$$p(h_j = 0|\mathbf{v}) = \frac{1}{1 + e^{w_j v + b_j}}, \quad (1.11)$$

$$p(h_j = 1|\mathbf{v}) = \frac{e^{w_j v + b_j}}{1 + e^{w_j v + b_j}} = \frac{1}{1 + e^{-w_j v - b_j}} = \sigma(\mathbf{w}_j \mathbf{v} + b_j). \quad (1.12)$$

Таким чином отримано формули (1.13) та (1.14) для обрахунку ймовірності, що прихована змінна h_j чи v_i при умові \mathbf{v} або \mathbf{h} , відповідно, буде дорівнювати одиниці.

$$p(h_j = 1|\mathbf{v}) = \sigma(\mathbf{w}_j\mathbf{v} + b_j), \quad (1.13)$$

$$p(v_i = 1|\mathbf{h}) = \sigma(\mathbf{w}_i\mathbf{h} + c_i). \quad (1.14)$$

Двійкові значення змінних дає змогу ефективно обчислювати ймовірнісний розподіл серед змінних видимого шару \mathbf{v} , за виразом

$$p(\mathbf{v}) = \sum_{\mathbf{h} \in \{0,1\}^H} p(\mathbf{v}, \mathbf{h}) = \frac{e^{\mathbf{c}^T\mathbf{v} + \sum_{j=1}^H \log(1 + e^{b_j + \mathbf{w}_j\mathbf{v}})}}{Z} = \frac{e^{-F(\mathbf{v})}}{Z}. \quad (1.15)$$

Таким чином, обчислення $F(\mathbf{v})$ має вигляд виразу:

$$F(\mathbf{v}) = \mathbf{c}^T\mathbf{v} + \sum_{j=1}^H \log(1 + e^{b_j + \mathbf{w}_j\mathbf{v}}) = \mathbf{c}^T\mathbf{v} + \sum_{j=1}^H \text{softplus}(b_j + \mathbf{w}_j\mathbf{v}). \quad (1.16)$$

Як бачимо функція вільної енергії є функцією активації по всіх нейронах видимого шару, у вигляді *softplus* функції, що зсунута вектором видимого шару.

Процес навчання полягає у виконанні стохастичного градієнтного спуску на обрхованій негативній функції правдоподібності, яка визначається формулою

$$L(\theta, D) = \sum_{\mathbf{v} \in D} \log p(\mathbf{v}), \quad (1.17)$$

де D — множина всіх можливих конфігурацій системи;

θ — параметри моделі.

Тому функція втрат стає негативною функцією правдоподібності і обчислюється за виразом

$$l(\theta, D) = -L(\theta, D) = -\sum_{\mathbf{v} \in D} \log p(\mathbf{v}). \quad (1.18)$$

Метою тренування ОМБ є знаходження таких параметрів системи, при якому ймовірність того, що згенерований вектор $\widetilde{\mathbf{v}}_t$ тотожній тренувальним даним \mathbf{v}_t , є максимальною. Функція правдоподібності визначається наступним чином

$$\frac{\partial(-\log p(\mathbf{v}))}{\partial\theta} = \frac{\partial F(\mathbf{v})}{\partial\theta} - \frac{\partial(-\log Z)}{\partial\theta}, \quad (1.19)$$

де $\frac{\partial F(\mathbf{v})}{\partial\theta}$ – позитивна фаза, що змушує модель надавати перевагу прихованим станам, які сумісні з видимим тренувальними даними;

$\frac{\partial(-\log Z)}{\partial\theta}$ - негативна фаза, що зменшує енергію станів, яким модель надає перевагу.

Вираз (1.19) спрощується в (1.20), оскільки обчислити аналітично другу частину за поліноміальний час неможливо, тому що воно є математичним очікуванням $E_{\mathbf{v},\mathbf{h}} \left[\frac{\partial E(\mathbf{v},\mathbf{h})}{\partial\theta} \right]$ по всім можливих значеннях \mathbf{v} та \mathbf{h} . Таким чином, щоб оцінити значення виразу $\frac{\partial(-\log Z)}{\partial\theta}$, слід виконати t -крокове випробування Гіббса, починаючи з \mathbf{v}_t , внаслідок чого градієнт визначається виразом (1.20), в результаті чого утворюється алгоритм CD-k

$$\frac{\partial(-\log p(\mathbf{v}_t))}{\partial\theta} = \frac{\partial F(\mathbf{v}_t)}{\partial\theta} - \frac{\partial F(\widetilde{\mathbf{v}}_t)}{\partial\theta}, \quad (1.20)$$

де $\widetilde{\mathbf{v}}_t$ — вектор, що має найбільшу подібність з \mathbf{v}_t і є результатом відбору проб за Гіббсом.

Відбір проб за Гіббсом – це алгоритм, який реалізує ланцюг Маркова – послідовність із станів, ймовірність переходу в кожен з яких залежить тільки від значення попереднього стану, а не від всієї послідовності станів. Для ОМБ станами виступають спостережувані та приховані шари, що чергуються. Ланцюг Маркова для ОМБ представлений на рисунку 1.1.

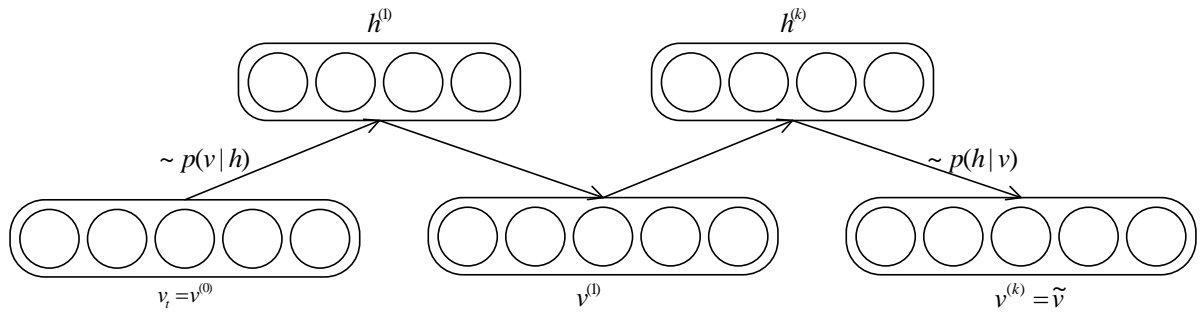


Рисунок 1.1 – Відбір проб за Гіббсом

Використовуючи вирази (1.13) і (1.14) для обчислення умовного розподілу отримуємо вирази для одержання значень змінних ланцюга Маркова

$$\mathbf{h}_{t+1} \sim \sigma(\mathbf{W}_{vh}\mathbf{v}_t + \mathbf{b}), \quad (1.21)$$

$$\mathbf{v}_{t+1} \sim \sigma(\mathbf{W}_{hv}\mathbf{h}_{t+1} + \mathbf{c}). \quad (1.22)$$

Таким чином, виконавши відбір проб k разів, можна отримати вектор значень видимого шару $\tilde{\mathbf{v}}_t$.

Часткові похідні функції втрат обчислюються наступними формулами (1.23) – (1.25), що дозволяє оновити параметри системи виразами (1.26) – (1.28), чим закінчується крок тренування ОМБ.

$$-\frac{\partial \log p(\mathbf{v})}{\partial W_{ij}} = E_v[p(h_i|\mathbf{v})v_j] - \tilde{v}_j^t \sigma(W_i v^t + c_i), \quad (1.23)$$

$$-\frac{\partial \log p(\mathbf{v})}{\partial c_i} = E_v[p(h_i|\mathbf{v})] - \sigma(W_i v^t), \quad (1.24)$$

$$-\frac{\partial \log p(\mathbf{v})}{\partial b_j} = E_v[p(v_j|\mathbf{h})] - \tilde{v}_j^t, \quad (1.25)$$

$$\mathbf{W} = \mathbf{W} + \alpha(\mathbf{h}(\mathbf{v}_t)\mathbf{v}_t^T - \mathbf{h}(\tilde{\mathbf{v}}_t)\tilde{\mathbf{v}}_t^T), \quad (1.26)$$

$$\mathbf{b} = \mathbf{b} + \alpha(\mathbf{h}(\mathbf{v}_t) - \mathbf{h}(\tilde{\mathbf{v}}_t)), \quad (1.27)$$

$$\mathbf{c} = \mathbf{c} + \alpha(\mathbf{v}_t - \tilde{\mathbf{v}}_t). \quad (1.28)$$

1.4.1.3 Глибинна мережа переконань

Глибинна мережа переконань (ГМП) – це глибока нейронна мережа, утворена об'єднанням декількох простих моделей, таких як ОМБ або автокодувачів. У випадку коли ГМП формується з двох ОМБ, прихований шар нижньої є видимим верхньої. Для кількості n більше двох, підхід повторюється $n - 2$ разів. Таким чином, утворюється модель з n прихованих шарів та 1 видимого, а число об'єднаних ОМБ $n - 1$. Для ГМП характерний спільний розподіл ймовірностей між векторами спостережуваного \mathbf{v} та n прихованих $\mathbf{h}^{(n)}$ рівнів. В загальному випадку, він обчислюється наступною формулою:

$$P(\mathbf{v}, \mathbf{h}^{(1)}, \dots, \mathbf{h}^{(n)}) = \left(\prod_{k=0}^{n-2} P(\mathbf{h}^{(k)} | \mathbf{h}^{(k+1)}) \right) P(\mathbf{h}^{(n-1)}, \mathbf{h}^{(n)}), \quad (1.29)$$

де $\mathbf{h}^{(0)} = \mathbf{v}$;

$P(\mathbf{h}^{(k)} | \mathbf{h}^{(k+1)})$ — умовний розподіл для спостережуваного шару k -ї ОМБ при умові «спостережуваного» шару $(k + 1)$ -ї ОМБ;

$P(\mathbf{h}^{(n-1)}, \mathbf{h}^{(n)})$ — ймовірнісний розподіл найвищої ОМБ.

Умовний ймовірнісний розподіл шару знаходиться за наступними виразами:

$$p(h_j^{(k)} = 1 | \mathbf{h}^{(k+1)}) = \sigma(\mathbf{b}_{\mathbf{h}^{(k)}} + \mathbf{W}^{(k+1)T} \mathbf{h}^{(k+1)}), \quad (1.30)$$

$$p(v_i = 1 | \mathbf{h}^{(k)}) = \sigma(\mathbf{c} + \mathbf{W}^{(k)T} \mathbf{h}^{(k)}). \quad (1.31)$$

Принцип жадібного плавного безконтрольного навчання може бути застосовано до ГМБ з ОМБ як будівельних блоків для кожного шару за алгоритмом:

1. тренувати перший шар як ОМБ, в якій вхідний вектор $\mathbf{x} = \mathbf{h}^{(0)}$;
2. використати перший шар для другого за формулами (1.31) – (1.32);
3. тренувати другий шар як ОМБ, як тренувальні дані;
4. повторювати 2-4 кроки для необхідної кількості ОМБ.

1.4.2 Кластеризація

1.4.2.1 Алгоритм k-prototype

Нехай $X = \{X_1, X_2, \dots, X_n\}$ множина об'єктів, яку необхідно розбити. $X_i = [x_{i1}, x_{i2}, \dots, x_{im}]$ об'єкт, представлений вектором атрибутів довжиною m . Задача алгоритму знайти розділ, який розбиває X на k кластерів. Мірою відстані є деяка функція вартості, що визначається виразом

$$E = \sum_{l=1}^k \sum_{i=1}^n y_{il} d(X_i, Q_l), \quad (1.32)$$

де $Q_l = [q_{l1}, q_{l2}, \dots, q_{lm}]$ представляє прототип кластера l ;

y_{il} – елемент матриці розділу, що визначається квадратом Евклідової відстані.

Коли X має категоріальні атрибути, функція вартості спрощується в наступну

$$E_l = \sum_{i=1}^k \sum_{i=1}^n y_{il} \sum_{j=1}^{m_r} (x_{ij}^r - q_{ij}^r)^2 + \gamma_l \sum_{i=1}^n y_{il} \sum_{j=1}^{m_c} \delta(x_{ij}^c, q_{ij}^c), \quad (1.33)$$

де x_{ij}^r, q_{ij}^r – числові атрибути;

x_{ij}^c, q_{ij}^c – категоріальні атрибути;

m_c – число категоріальних атрибутів;

m_r – кількість числових атрибутів;

γ_l – вага категоріального атрибуту для кластеру l .

1.5 Висновки

В даному розділі було проаналізовано предметну область, існуючі рішення, в результаті чого були сформувані вимоги до системи, для виконання яких наведено та використано теоретичні відомості

2 СИНТЕЗ МОДЕЛІ ТА ВИБІР АЛГОРИТМУ КЛАСТЕРИЗАЦІЇ

На сьогоднішній день більшість компанії мають достатню кількість даних про користувачів та клієнтів, аби передбачати їх поведінку та уподобання задля того, щоб пропонувати продукти чи послуги або ж рекламу. Для вирішення такої проблеми може використовуватися рекомендаційні системи, побудовані за допомогою колаборативної фільтрації чи тематичної фільтрації. Також можна кластеризувати всіх користувачів звичайними алгоритмами кластеризації.

Задля цього систему було розбито на дві частини. Перша – це моделювання клієнта на основі його ознак та уподобань, а друга – кластеризація клієнтів за їх ознаками та уподобаннями. Таким чином, обидві частини можуть незалежно один від одної виконувати свої функції. Заміна алгоритму кластеризації ніяк не вплине на роботу моделі клієнта та навпаки.

Дані про клієнтів представлені у вигляді двох відношень. В першому знаходиться інформація, що характеризує, власне, клієнта, а в другому та, що описує його поведінку.

Проаналізувавши фрагменту даних про клієнтів, що наведений у додатку А, впливає, що їх можна розділити на дві множини. Розглянемо першу з них. Вона представляє собою набір відповідності між ознакою та її значенням, що характеризує даного клієнта. Такими ознаками є, наприклад, рівень заробітку, сімейний статус, інтереси, типи темпераменту, або вага, що може необхідним в деяких випадках. Друга описує поведінку клієнта стосовно продуктів чи послуг, яка визначається частотою їх користування, яку визначає компанія. В даному випадку це вказано шкалою від 0 до 10, де максимальне значення означає найменший період.

2.1 Моделювання клієнта

2.1.1 Синтез моделі

Для отримання найбільш точного визначення ознак необхідно розробити модель користувача, тобто клієнта, якого можна представити у вигляді вектору ознак, що мають, як якісний, так і кількісний тип. В свою чергу, клієнт на будь-якому ресурсі має уподобання, що також представлені у вигляді вектору. Отже, клієнта можна описати, як об'єкт, що має на вході дані, у вигляді вектору ознак та уподобань, а на виході – вектор латентних уподобань. Виходячи з цього, можна прийти до висновку, що вихідний вектор досліджуваної моделі залежить від ознак і виникає відношення, яке можна описати формулою (2.1).

$$Y = F(X, P), \quad (2.1)$$

де Y – вектор латентних уподобань;

X – вектор ознак;

P – вектор уподобань.

Проаналізувавши дані про обрані уподобання клієнтів, можна прийти до висновку, що клієнт не завжди надає перевагу тому чи іншому продукту або послугі, тобто виникає необхідність ввести ймовірність того, що клієнт має прихований факт або уподобання. Таким чином, клієнт описується наступною залежністю.

$$p(Y) = F(X, P), \quad (2.2)$$

де $p(Y)$ – ймовірнісний розподіл, в якому кожен елемент вектору представляє собою ймовірність того, що клієнт має дане приховане уподобання.

Оскільки, клієнт описується залежністю ймовірності від ознаки, можна прийти до висновку, що для його моделювання, можна використати нейронні мережі, що ма-

ють на виході системи ймовірнісний розподіл. Таким видом ШНМ є моделі, що базуються на енергії системи. Тобто, існує деяка енергія, що описує стан системи, в залежності від її параметрів та конфігурації, а ймовірність перейти в стан з меншою енергією є більшим. Одним із видів таких мереж є машина Больцмана, яка може розглядатися, як стохастичний генеративний варіант мережі Хопфілда. Однак, через необмежену зв'язність вона не може бути використана в практичному підході. В такому випадку можна використати обмежену машину Больцмана (ОМБ), яка має головну відмінність – наявність прихованих та видимих шарів та відсутністю зв'язків між нейронами одного шару. У випадку моделі клієнта, така НМ дозволяє знаходити приховані зв'язки між ознаками та уподобаннями і визначати приховані зв'язки, що характеризують клієнта і визначають його особливість. З цього випливає, що модель повинна складатися з двох, перша з яких визначає приховані ознаки, а інша – латентні зв'язки між обраними продуктами. Після чого, обидві моделі, об'єднуючись, представляють собою модель клієнта.

Розглянемо модель для роботи з ознаками, утворену на основі ОМБ. Як видно з рисунку 2.1, НМ складається з прихованого та видимого шарів, де перший представляє собою ймовірнісний розподіл по уподобанням, а другий – ознаки. Елементи як видимого, так і прихованого шарів не мають зв'язків між собою. M – кількість уподобань, N – кількість ознак. $\mathbf{h} \in \{0, 1\}^M$, $\mathbf{W} \in \mathbb{R}^{M \times N}$, $\mathbf{v} \in \{0, 1\}^N$. Зміщення нейронів видимого шару $\mathbf{c} \in \mathbb{R}^N$, прихованого – $\mathbf{b} \in \mathbb{R}^M$.

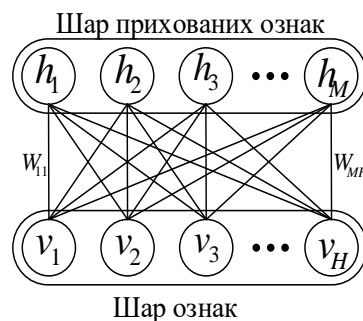


Рисунок 2.1 – Модель клієнта, на основі ОМБ

Навчання ОМБ відбувається за, описаним в розділі 1.4.1.5, алгоритмам, що має назву порівняльної розбіжності у поєднанні з відбором проб за Гіббсом, що дає можливість обраховувати ймовірнісний розподіл за час T без обчислень очікування по всім можливим конфігураціям моделі. Умовний ймовірнісний розподіл обчислюється за формулами (1.13), (1.14), а енергія системи – (1.5).

Розглянемо вхідні дані нейронної мережі. Елементи видимого шару приймають значення одиниці або нуля, тому така НМ носять назву бінарна обмежена машина Больцмана. Прикладом застосування такого виду може бути рекомендаційна система на основі переглянутих фільмів користувачем[13]. Кожен з елементів видимого шару визначає переглядав користувач відповідний фільм, чи ні, а прихованого шару – ймовірності того, що клієнт надає перевагу певному жанру.

Проаналізувавши дані про ознаки клієнта, можна прийти до висновку, що їх можна розділити на дві категорії – кількісні та якісні. Кількісні – це ті, що можуть бути виміряні та представлені числом, а якісні – ті, які можна чітко розрізнити. Таким чином, можна прийти до висновку, що використання такого типу НМ для описання вхідних ознак не підходить, тому що деякі з ознак є реальними значеннями, а інші відносяться до якісних. Отже, необхідно привести до однієї форми.

Розглянемо випадок, коли кожен з елементів видимого шару відображує наявність кожного значення в ознаках клієнта. Нехай, клієнт має K ознак і деяку ознаку p , що характеризується числом, $r \in \mathbb{R}$. Тоді кількість значень дорівнює потужності множини \mathbb{R} . Це означає, що випадку кількісних даних, число значень може бути необмеженим, що буде дорівнювати кількості елементів в групі видимого шару, а це неможливо, навіть, якщо r обмежена деякою областю допустимих значень. У випадку якісних даних, ознака p може бути представлена цілим числом z , яке відповідає порядку в множині значень ознаки $\mathbb{Z}_1 \in \mathbb{Z}$, потужністю L , яка обмежена областю значень, тобто $z \in \mathbb{Z}_1$. Такий спосіб дозволяє вирішити проблему наявності декількох значень ознак у клієнта, який зображено на рисунку 2.2, з якого видно, що видимий шар має дві групи нейронів, які мають L та H , відповідно.

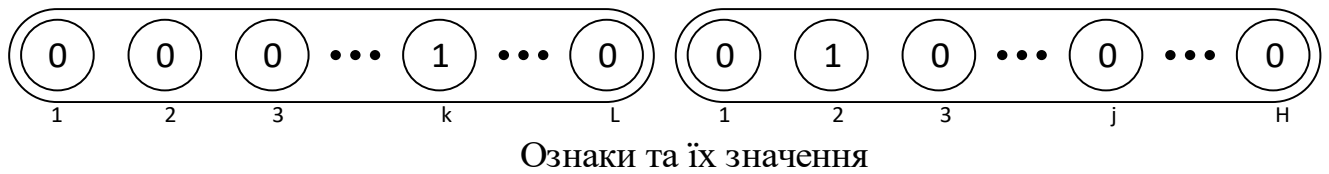


Рисунок 2.2 – Групи нейронів видимого шару

Групи нейронів видимого шару являють собою значення відповідної ознаки, а кожен з елементів показує яке саме значення воно приймає. В такому випадку, значенням є категорія, до якої відноситься дане значення.

Перевагами такого зображення видимого шару є можливість використовувати як і бінарні, так і визначені вищезазначеним способом. До недоліків можна віднести високу залежність довжини видимого шару від кількості значень кожного з нейронів. Для додавання нових значень, модель має бути повністю перебудована.

Таким чином, модель набуває вигляду, зображеному на рисунку 2.3.

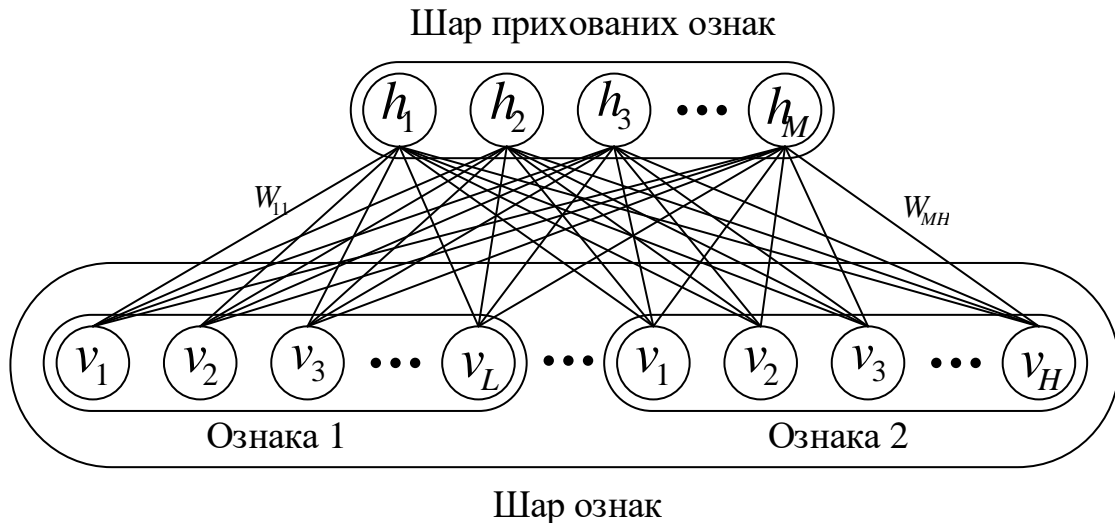


Рисунок 2.3 – Початковий вигляд моделі

Як видно з рисунку 2.3, на якому зображено початковий вигляд моделі, вона представляє собою звичайну ОМБ. Ймовірнісний розподіл обчислюється за формулою (1.6). Виходячи з того, що ОМБ має властивість породжувальної моделі, можна прийти до висновку, що маючи вектор значень елементів прихованого шару, можна

обрахувати ймовірнісний розподіл для видимого шару за допомогою відбору проб за Гіббсом. З чого випливає, що знаючи приховані уподобання можна передбачити, якими будуть ознаки клієнта.

Інакше модель може бути побудована шляхом перетворення вектору видимого шару \mathbf{v} в деяку матрицю \mathbf{V} розмірністю $K \times N$, де K – максимальна кількість значень ознаки по всіх ознаках клієнта. $\mathbf{h} \in \{0, 1\}^M$, $\mathbf{W} \in \mathbb{R}^{M \times NK}$, $\mathbf{V} \in \{0, 1\}^{NK}$. Зміщення нейронів видимого шару $\mathbf{c} \in \mathbb{R}^{NK}$, прихованого – $\mathbf{b} \in \mathbb{R}^M$. Кожен елемент матриці v_{ik} є індикатором відповідності значення ознаки клієнта відповідній категорії, тобто, якщо клієнт має категорію k , то, відповідно, $v_{ik} = 1$ та $v_{ik} = 0$ навпаки. Використання такого підходу дає можливість чіткіше відобразити незалежність довжини видимого шару від кількості значень елементів, а додавання нових значень не дуже вплине на будову НМ.

Перевагами такого зображення видимого шару є можливість використовувати категоріальні, вищезазначеним способом. До недоліків можна віднести те, що довжини векторів значень ознак мають бути рівними, проте в дійсності, категорії можуть не мати таку кількість. Таким чином утворена k -нарна ОМБ вигляд якої зображений на рисунку 2.4.

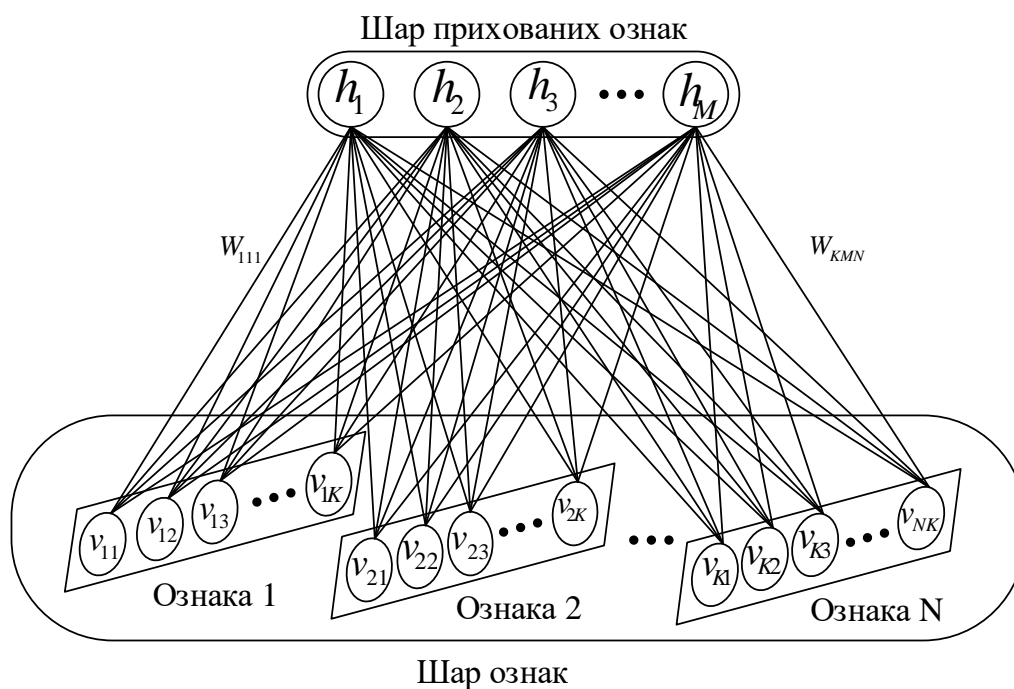


Рисунок 2.4 – ОМБ з матричним видимим шаром

Аналогічно до звичайної ОМБ, енергія системи обчислюється за формулою (2.3), а ймовірнісний розподіл – за виразом (2.4).

$$E(\mathbf{V}, \mathbf{h}) = -\sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^K W_{ijk} h_j v_{ik} - \sum_{i=1}^N \sum_{k=1}^K v_{ik} c_{ik} - \sum_{j=1}^M h_j b_j, \quad (2.3)$$

$$p(\mathbf{V}, \mathbf{h}) = \frac{e^{-E(\mathbf{V}, \mathbf{h})}}{Z}. \quad (2.4)$$

Граничний розподіл по видимому шару \mathbf{V} визначений наступним твердженням, що є ймовірністю того, що мережа призначить видимому шару, сумуючи по всім можливим прихованим векторам.

$$p(\mathbf{V}) = \sum_{\mathbf{h}} \frac{e^{-E(\mathbf{V}, \mathbf{h})}}{\sum_{\mathbf{V}' \mathbf{h}'} e^{-E(\mathbf{V}', \mathbf{h})}}. \quad (2.5)$$

В даній НМ використовується умовний мультиноміальний розподіл для моделювання кожного стовпця видимого шару та умовний розподіл Бернуллі для моделювання прихованого шару. Аналогічно до звичайної ОМБ, ймовірність того, що $h_j = 0$, при умові \mathbf{v} обчислюється за формулою (1.11), а $h_j = 1$, при умові \mathbf{v} – за формулою (1.12). Головною відмінністю є розмірність вектору \mathbf{v} та матриці \mathbf{W} . Отже, вираз для обчислення набуває вигляду формули 2.6.

$$p(h_j = 1 | \mathbf{V}) = \sigma(b_j + \sum_{k=1}^K \mathbf{v}_k \mathbf{W}_k), \quad (2.6)$$

Однак обрахувати розподіл видимого шару $v_i = 1$, при умові \mathbf{h} неможливо, оскільки, він представляється у вигляді матриці. Тому, визначається вираз для обрахунку розподілу для кожного зі стовпців \mathbf{v} .

$$p(v_{ik} = 1 | \mathbf{h}) = \frac{e^{c_{ik} + \mathbf{h}^T \mathbf{W}_{ik}}}{\sum_{l=1}^K e^{c_{il} + \mathbf{h}^T \mathbf{W}_{il}}}, \quad (2.7)$$

що можна довести з виходячи з перетворень, наведених у формулі (2.8).

$$\begin{aligned}
p(\mathbf{V}|\mathbf{h}) &= \frac{p(\mathbf{V}, \mathbf{h})}{\sum_{\mathbf{V}'} p(\mathbf{V}', \mathbf{h})} = \left[p(\mathbf{V}, \mathbf{h}) = \frac{e^{-E(\mathbf{V}, \mathbf{h})}}{Z} \right] = \\
&= \frac{\frac{e^{-E(\mathbf{V}, \mathbf{h})}}{Z}}{\frac{\sum_{\mathbf{V}'} e^{-E(\mathbf{V}', \mathbf{h})}}{Z}} = \frac{e^{-E(\mathbf{V}, \mathbf{h})}}{\sum_{\mathbf{V}'} e^{-E(\mathbf{V}', \mathbf{h})}} = \\
&= \left[= - \sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^K W_{ijk} h_j v_{ik} - \sum_{i=1}^N \sum_{k=1}^K v_{ik} c_{ik} - \sum_{j=1}^M h_j b_j \right] = \\
&= \frac{e^{hb^T + c^T \mathbf{V} + h^T \mathbf{W} \mathbf{V}}}{\sum_{\mathbf{V}'} e^{hb^T + c^T \mathbf{V}' + h^T \mathbf{W} \mathbf{V}'}} = \frac{e^{hb^T} e^{c^T \mathbf{V} + h^T \mathbf{W} \mathbf{V}}}{e^{hb^T} \sum_{\mathbf{V}'} e^{c^T \mathbf{V}' + h^T \mathbf{W} \mathbf{V}'}} = \\
&= \frac{e^{\sum_i c_i v_i + h^T \mathbf{W}_i v_i}}{\sum_{\mathbf{V}'} e^{\sum_i c_i v'_i + h^T \mathbf{W}_i v'_i}} = \frac{\prod_i e^{c_i v_i + h^T \mathbf{W}_i v_i}}{\sum_{\mathbf{V}'} \prod_i e^{c_i v'_i + h^T \mathbf{W}_i v'_i}} = \\
&= \prod_{i=1}^N \frac{e^{c_i v_i + h^T \mathbf{W}_i v_i}}{\sum_{v'_i} e^{c_i v'_i + h^T \mathbf{W}_i v'_i}} = \\
&= \prod_{i=1}^N \frac{e^{c_i v_i + \sum_{j=1}^M h_j w_{ij} v_i}}{\sum_i \sum_{l=1}^K e^{c_{il} + \sum_{j=1}^M h_j w_{ijl}}}, \tag{2.8}
\end{aligned}$$

і таким чином,

$$p(v_{ik} = 1|\mathbf{h}) = \frac{e^{c_{ik} + \sum_{j=1}^M h_j w_{ijk}}}{\sum_{l=1}^K e^{c_{il} + \sum_{j=1}^M h_j w_{ijl}}}. \tag{2.9}$$

Навчання такого виду моделі вимагає виконання градієнтного спуску на функції правдоподібності для оновлення параметрів, що показано у виразі (2.10).

$$\Delta w_{ijk} = \frac{\partial \log p(\mathbf{V})}{\partial w_{ijk}} = E_{v_{ijk}h_j|data} - E_{v_{ijk}h_j|model}, \quad (2.10)$$

де $E_{v_{ijk}h_j|data}$ – очікування, що визначає частоту, з якою i -та ознака, k -те значення ознаки та j -те уподобання зустрічаються разом, а $E_{v_{ijk}h_j|model}$ – очікування по відношенню до розподілу, визначеного моделлю.

Останній член цього виразу не може бути обчислено аналітично за менш, ніж експоненціальний час [14], тому для уникнення такого виду проблеми, застосовується принцип контрастивної дивергенції, або порівняльної розбіжності, що показано у виразі (2.11).

$$\Delta w_{ijk} = E_{v_{ijk}h_j|data} - E_{v_{ijk}h_j|K}, \quad (2.11)$$

де $E_{v_{ijk}h_j|K}$ – очікування, що представляє собою розподіл, отриманого способом відбору проб за Гіббсом, аналогічно до звичайної ОМБ.

Згідно до властивостей ОМБ, існує можливість передбачити ознаку клієнта, використовуючи видимий шар $\tilde{\mathbf{V}}$. Для цього необхідно обчислити ймовірність того, що $h_j = 1$ при умові $\tilde{\mathbf{V}}$, а потім $v_{qk} = 1$ при умові \hat{h} , отриманого на попередньому кроці.

Такий підхід дозволяє обрахувати розподіл за поліноміальний час, однак втрачається точність у порівнянні з методом, що базується на енергії системи. Отже,

$$\hat{h}_j = p(h_j = 1 | \tilde{\mathbf{V}}) = \sigma(b_i + \sum_{i=1}^M \sum_{k=1}^K v_{ik} w_{ijk}), \quad (2.12)$$

$$p(v_{qk} = 1 | \tilde{\mathbf{h}}) = \frac{e^{c_{qk} + \sum_{j=1}^M \tilde{h}_j w_{qjk}}}{\sum_{l=1}^K e^{c_{ql} + \sum_{j=1}^M \tilde{h}_j w_{qjl}}}. \quad (2.13)$$

Таким чином було синтезовано модель, здатну обчислювати приховані ознаки клієнта, що базується на ОМБ.

Проаналізуємо дані про уподобання клієнта, інформація про яких записана у вигляді частоти користування продуктом або послугою. Частота представляється у

вигляді шкали від 0 до 10. Виникає необхідність представити таку інформацію у вигляді вектора \mathbf{o} , елементи якого визначають ступінь використання. Тобто, значення, в цьому контексті також, ознаки визначає який елемент вектора має бути рівним 1. Таким чином, розглянуту ознаку можна представити аналогічно до попередньої моделі. Оскільки клієнт має N уподобань, виходить, що утворюється матриця уподобань, що у контексті моделі є видимим шаром іншої ОМБ, тобто, матрицю \mathbf{V} розмірністю $K \times N$, де K – максимальна кількість значень ознаки по всіх ознаках клієнта. $\mathbf{h} \in \{0, 1\}^M$, $\mathbf{W} \in \mathbb{R}^{M \times NK}$, $\mathbf{V} \in \{0, 1\}^{NK}$. Зміщення нейронів видимого шару $\mathbf{c} \in \mathbb{R}^{NK}$, прихованого – $\mathbf{b} \in \mathbb{R}^M$. Кожен елемент матриці v_{ik} є індикатором відповідності значення частоти використання продукту клієнтом індексу в векторі \mathbf{o} , якщо клієнт має частоту використання уподобання, що знаходиться за індексом k , то, відповідно, $v_{ik} = 1$ та $v_{ik} = 0$ навпаки. Навчання та ймовірнісний розподіл обчислюються ідентично до першої моделі.

Недоліком такої моделі є те, що клієнт може не завжди мати всі ознаки. Рішенням є використати умовну ОМБ. Проте таких вимог до системи не має та всі клієнти за наведених даних їх мають.

Таким чином, було отримано модель для визначення прихованих зв'язків по відношенню до уподобань, яка зображена на рисунку 2.5.

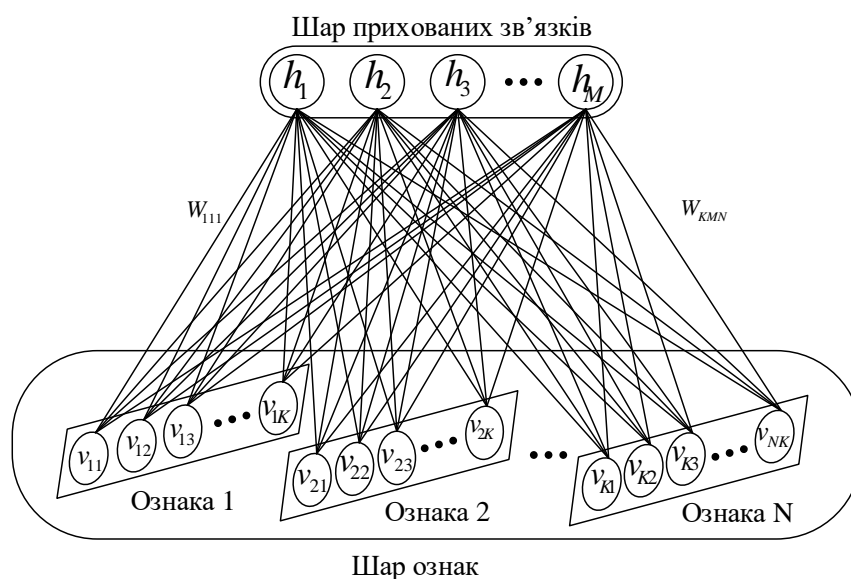


Рисунок 2.5 – Модель для визначення прихованих зв'язків

Останнім кроком у синтезі моделі клієнта є об'єднання обох моделей в одну для цілісності. Така модель здатна обчислювати та визначати приховані зв'язки, що клієнта одночасно і характеризують його і описують поведінку. Їх можна назвати латентними уподобаннями. Об'єднана загальна модель показана на рисунку 2.6.

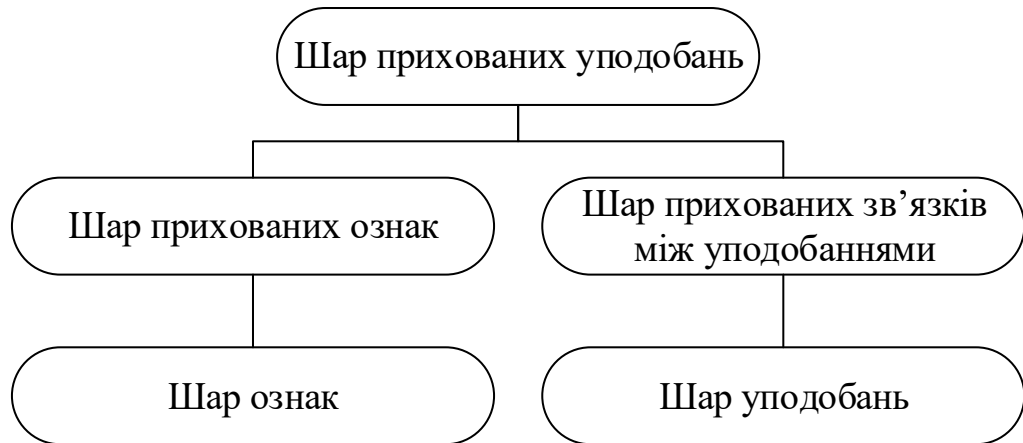


Рисунок 2.6 – Загальна модель клієнта

Як видно з вище наведеного рисунку, дві моделі було об'єднано в одну шляхом представлення прихованих шарів видимим для ОМБ вищого шару, як показано на рисунку 2.7.

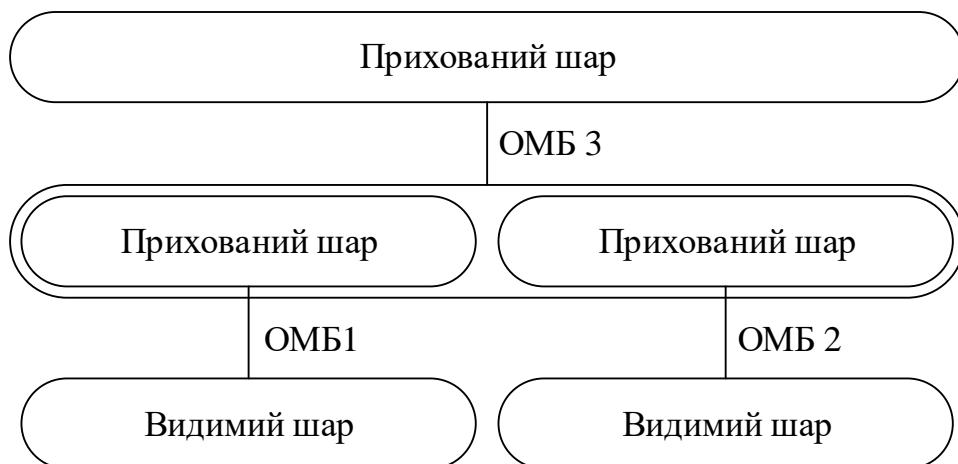


Рисунок 2.7 – Структурна схема моделі

Утворена ОМБ є звичайною бінарною ОМБ, оскільки $\mathbf{h}^{(1)} \in \{0,1\}^{M_1}$ та $\mathbf{h}^{(2)} \in \{0,1\}^{M_2}$, де $\mathbf{h}^{(1)}$ та $\mathbf{h}^{(2)}$ вектори прихованих шарів ОМБ1 та ОМБ2, відповідно. Таким чином, $\mathbf{v}^{(3)} = [\mathbf{h}^{(1)} \mathbf{h}^{(2)}]$.

Ймовірнісний розподіл визначається за формулою (1.6), а навчання відбувається за вищеописаним алгоритмом порівняльної розбіжності.

Структурна схема моделі наведена в додатку Б.

2.1.2 Варіації моделі клієнта

Розглянемо варіації моделі клієнта. Завдяки своїй ієрархічній структурі, утвореній внаслідок об'єднання декількох ОМБ, ГМП[15] дозволяє обрахувати приховані глибокі зв'язки. Перші декілька шарів вибирають низько-рівневі особливості, в той час, як верхні уточнюють раніше визначені шаблони для визначення більш складних ознак. Останній вихідний шар визначає ймовірності тих чи інших глибоких зв'язків. Виходячи з того, що модель клієнта складається з трьох ОМБ, можна утворити 4 варіації, використовуючи ГМП. Тобто, утворюються наступні послідовності архітектур: (ГМП+ОМБ)+ОМБ, (ОМБ+ГМП)+ОМБ, (ОМБ+ОМБ)+ГМП та (ГМП+ГМП)+ГМП.

Вигляд моделей представлений на рисунках 2.8 – 2.11 та в додатку В.

У всіх варіаціях видимими шарами є матриці $\mathbf{V}_1, \mathbf{V}_2$, розмірності $N_1 \times K_1$ та $N_2 \times K_2$, відповідно, де N_1 і K_1 – кількість ознак та їх категорій, а N_2 та K_2 кількість уподобань клієнта та частота їх користування. У випадку, коли модель для визначення прихованих ознак реалізована першим способом, видимий шар є вектором \mathbf{v}_1 розмірністю $N_1 K_1$.

Перша варіація може використовуватися, коли необхідно визначити більш глибокі зв'язки в ознаках клієнта.

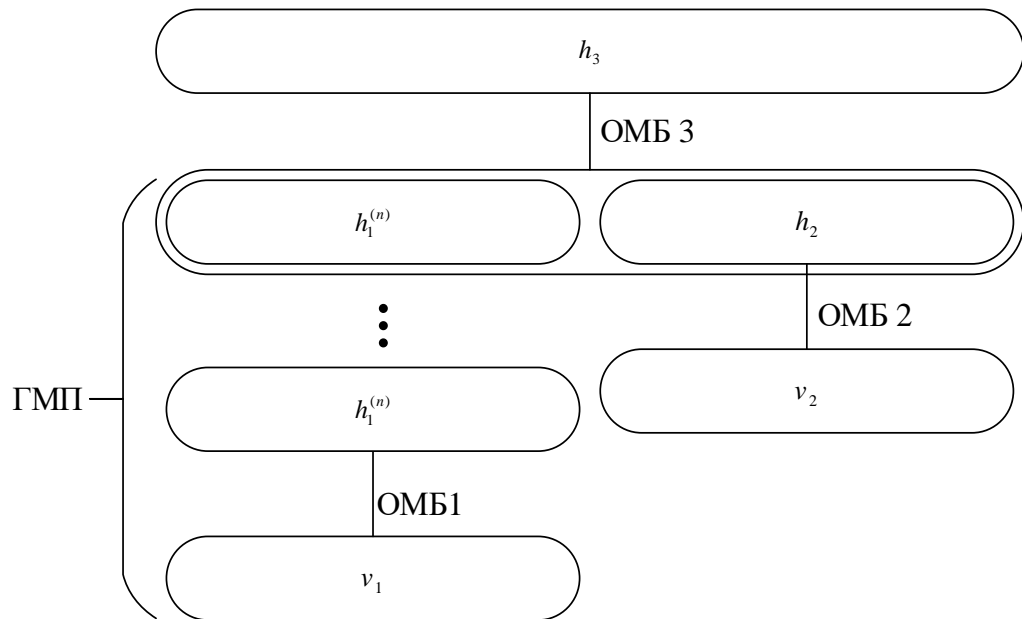


Рисунок 2.8 – Вигляд варіації (ГМП+ОМБ)+ОМБ

Друга застосовується для знаходження особливостей між уподобаннями клієнта.

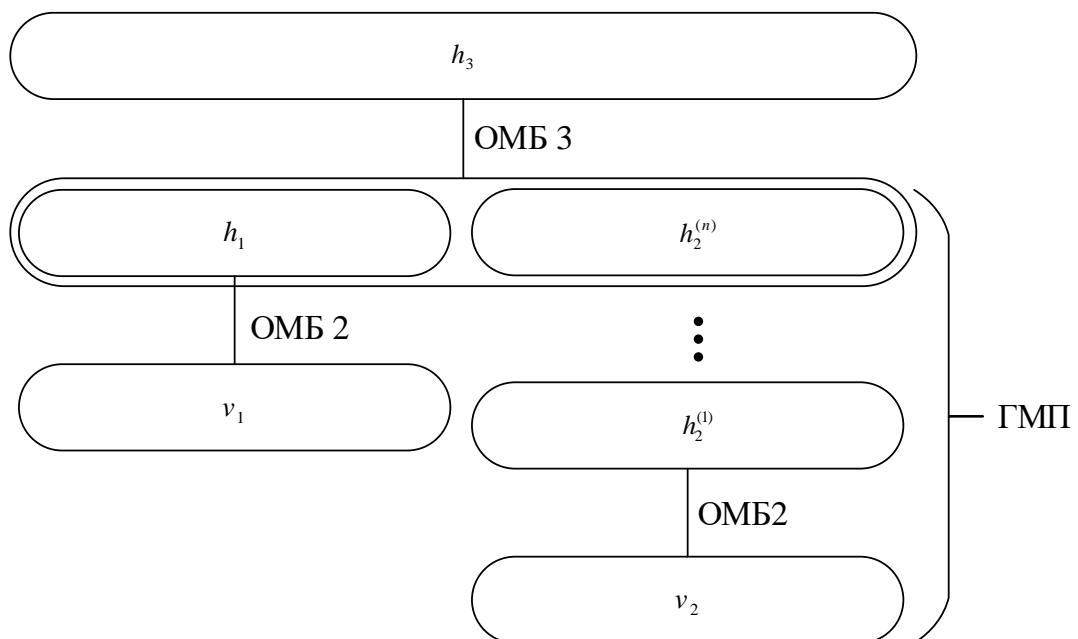


Рисунок 2.9 – Вигляд варіації (ОМБ+ГМП)+ОМБ

Третя дає можливість уточнити, знайдені нижчими ОМБ, зв'язки.

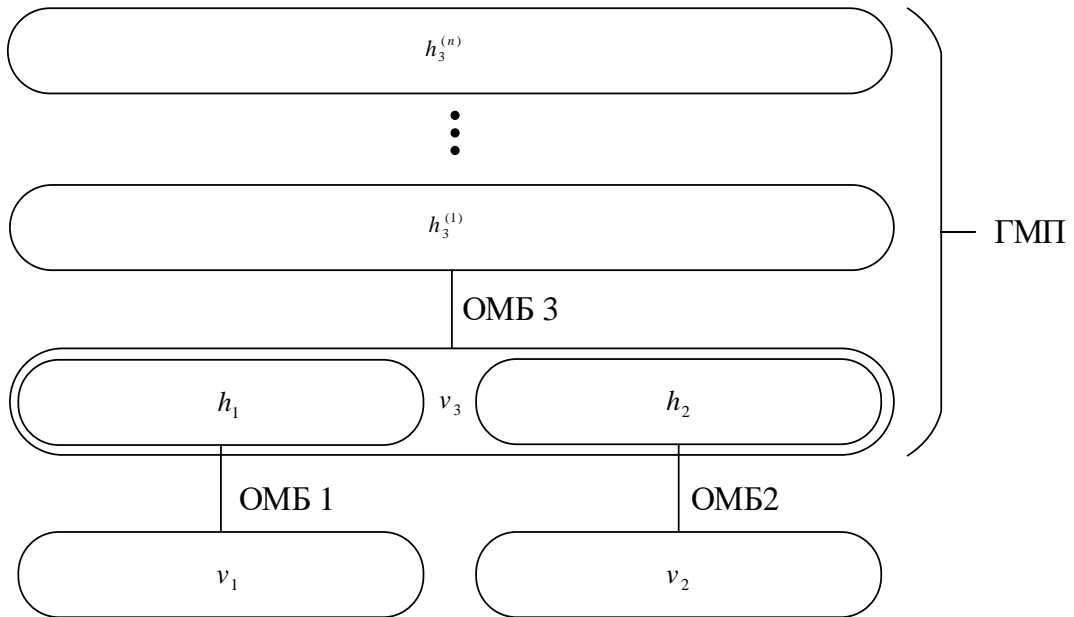


Рисунок 2.10 – Вигляд варіації (ОМБ+ОМБ)+ГМП

Остання об'єднує вище перераховані варіації в одну.

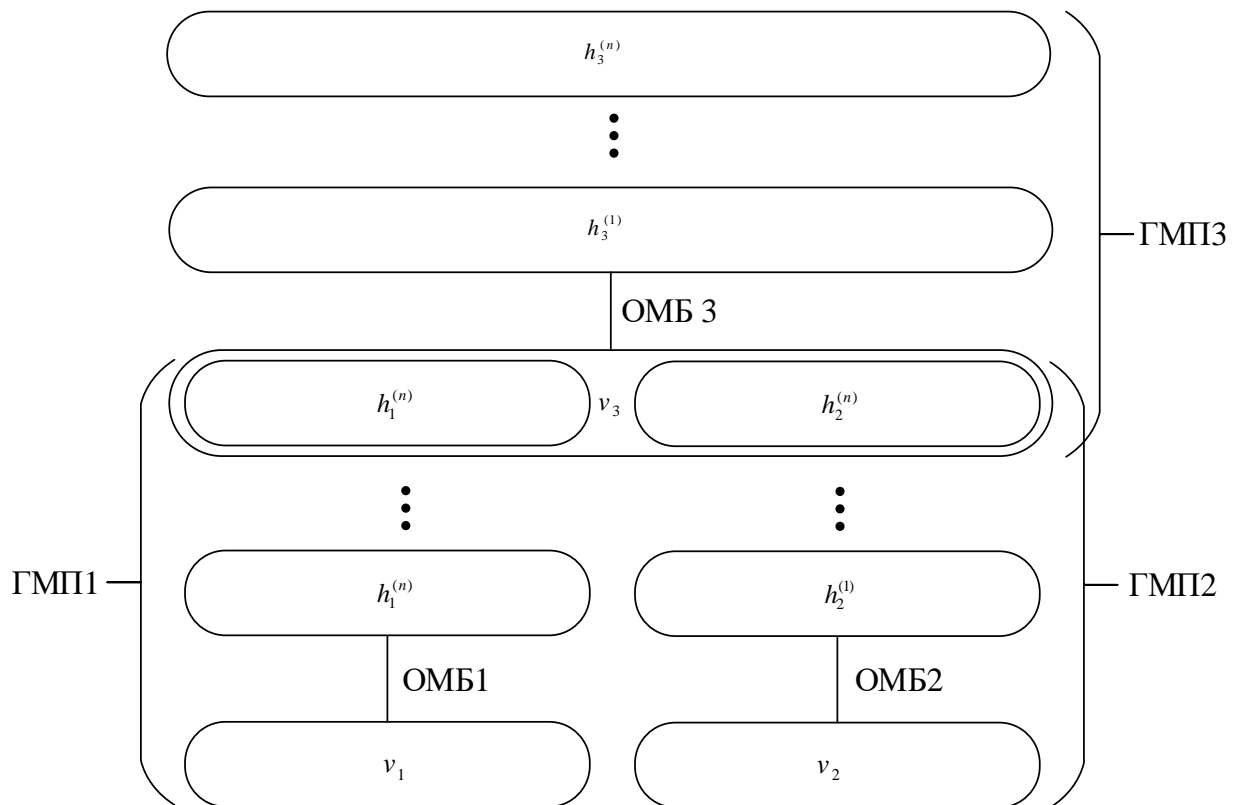


Рисунок 2.11 – Вигляд варіації (ГМП+ГМП)+ГМП

2.2 Кластеризація клієнтів

Для відображення зв'язків між клієнтами, необхідно використати кластеризацію, розбивши множину на підмножини. Тобто, схожі клієнти мають бути об'єднані в групи та бути на відстані з іншими. Як було згадано раніше, клієнт представляється як залежність прихованих уподобань від його ознак та уподобань, чим виражається частотою їх користування. Оскільки, уподобання, ознаки та приховані зв'язки, визначені НМ, є векторами, на одному просторі представити їх неможливо, як єдиного клієнта у вигляді точки, тому було запропоновано виконати розбиття на трьох просторах. Одне для кластеризації клієнтів за їх ознаками, друге – за їх уподобаннями, третє об'єднує за прихованими зв'язками. Такий спосіб дає можливість чітко відокремити клієнтів за різними клієнтів та ефективно проводити аналітику. Наприклад, серед клієнтів, у яких відомі ознаки та наявна інформація про уподобання можна знаходити приховані групи та класи, що неявно існують.

Розглянемо кластеризацію за ознаками. Як було описано вище, наведені дані про клієнта можна поділити на числові та категоріальні. Тому, виникає проблема обчислення відстані між об'єктами з різнотипними атрибутами. Звичайно, якщо розташувати, як і для однорідних, частина інформації втратиться. Якщо визначати відстань на евклідовому просторі, то виникнуть проблеми з категоріальними ознаками. Наприклад, нехай існує категоріальний атрибут з двома протилежними значеннями, то зрозуміло, що відстань між об'єктами з різними значеннями по цій осі дорівнює нулю, а з однаковими – одиниці. У випадку атрибута з набором значень можна використати шкалу, визначивши порядок значення або присвоївши бал. Таким чином, приходимо до висновку, що для вирішення проблеми потрібно застосувати алгоритм, що здатний працювати зі змішаними типами даних. Одним з таких алгоритмів є k-prototype, деталі якого наведені в розділі 1.4.2.1.

Нехай X – множина клієнтів така, що $X = \{X_1, X_2 \dots X_{20}\}$, і $X_i = [c_i, r_{i1}, r_{i2}]$ – i -й клієнт, представлений у вигляді вектору атрибутів, де $c_i \in \{0,1\}$ – категоріальна ознака, а $r_{i1}, r_{i2} \in \overline{1,10}$ – кількісні.

Результат кластеризації показано на рисунку 2.12, з якого видно, що множину клієнтів X було розбито на дві підмножини, групи. До першої входять ті клієнти, у яких $c = 0$ та $r_1, r_2 \in \overline{1,4}$, а до другої – $c = 1, r_1, r_2 \in \overline{5,10}$. Однак, виникли випадки, коли клієнти, що мають категорію протилежну множині, входять до неї. Це пов'язано з тим, що відстань по числовим атрибутам більше впливає, ніж вага категорії. Проаналізувавши даний приклад, приходимо до висновку, що даний алгоритм забезпечить необхідний рівень кластеризації.

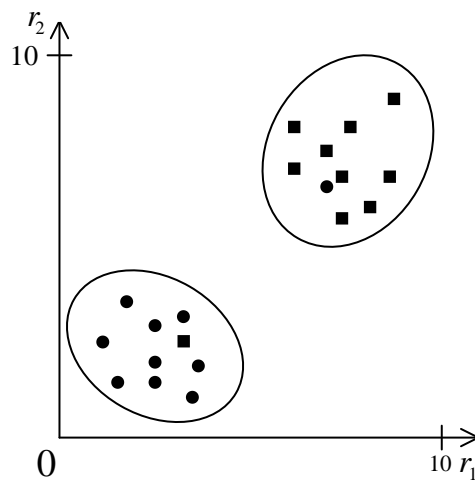


Рисунок 2.12 – Кластеризація по ознакам

Розглянемо кластеризацію за уподобаннями. У клієнта вони представлені у вигляді залежності продукту від частоти використання, що має значення по шкалі. Оскільки вони є числовими, то можна обрахувати значення відстані між об'єктами. Таким чином, приходимо до висновку, що для вирішення проблеми можна застосувати будь-який алгоритм, що здатний працювати зі числовими даними. Одним з таких алгоритмів є k-means.

Нехай X – множина клієнтів така, що $X = \{X_1, X_2 \dots X_{20}\}$, і $X_i = [r_{i1}, r_{i2}]$ – i -й клієнт, представлений у вигляді вектору частот використання уподобань, $r_{i1}, r_{i2} \in \overline{1,10}$.

Результат кластеризації показано на рисунку 2.13, з якого видно, що множину клієнтів X було розбито на підмножини в залежності від частоти використання продуктів. Проаналізувавши даний приклад, приходимо до висновку, що даний алгоритм забезпечить необхідний рівень кластеризації.

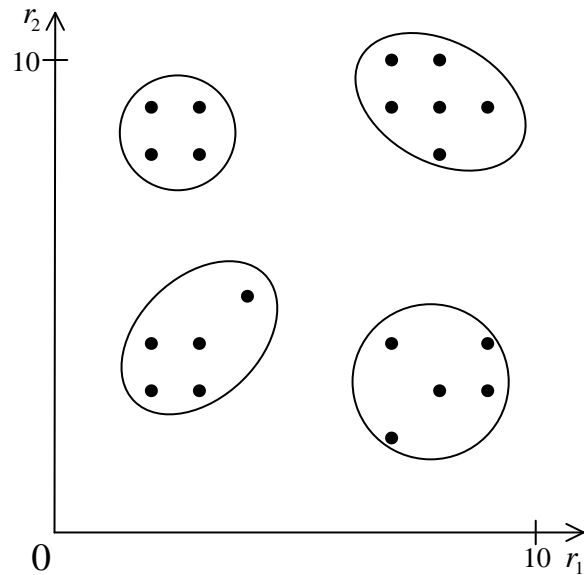


Рисунок 2.13 – Результат кластеризації за уподобаннями

Розглянемо кластеризацію за прихованими особливостями клієнта. У клієнта вони представлені у вигляді залежності продукту від частоти використання, що є дробове число від 0 до 1, оскільки визначаються ймовірнісним розподілом. Оскільки вони є кількісними, то можна обрахувати значення відстані між об'єктами. Таким чином, приходимо до висновку, що для вирішення проблеми можна застосувати будь-який алгоритм, що здатний працювати зі числовими даними. Одним з таких алгоритмів є k-means.

Нехай X – множина клієнтів така, що $X = \{X_1, X_2 \dots X_{20}\}$, і $X_i = [r_{i1}, r_{i2}]$ – i -й клієнт, представлений у вигляді вектору чисел, а $r_{i1}, r_{i2} \in [0; 1]$ – кількісні.

Результат кластеризації показано на рисунку 2.14, якого видно, що множину клієнтів X було розбито в залежності від прихованих зв'язків, визначених НМ на основі ознак і уподобань, на три групи. До першої входять клієнти, які мають $p(r_1) \in [0; 0.5), p(r_2) \in [0.5; 1]$, до другої – $p(r_2) \in [0; 0.5), p(r_1) \in [0.5; 1]$, а до останньої –

ті, у яких ймовірність прояву прихованого зв'язка r_1 та r_2 більше 50%. Проаналізувавши даний приклад, приходимо до висновку, що даний алгоритм забезпечить необхідний рівень кластеризації.

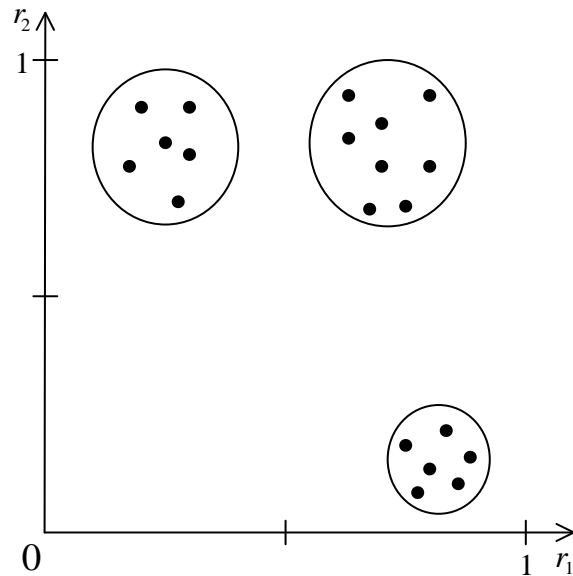


Рисунок 2.14 – Результат кластеризації за прихованими зв'язками

2.3 Висновки

В даному розділі було визначено модель клієнта та алгоритм кластеризації для компаративної оцінки. Для моделі було використано комбінацію з бінарної та кнарної обмеженої машини Больцмана. Алгоритмами кластеризації було обрано k-means за уподобаннями та прихованими особливостями, а за ознаками – k-prototype, що дозволяє працювати з категоріальними та якісними ознаками.

3 ТЕХНІЧНЕ РІШЕННЯ ТА РЕАЛІЗАЦІЯ

3.1 Обґрунтування вибору архітектури

Інтелектуальна система компаративної оцінки клієнтських уподобань являє собою рішення, яке будуть використовувати компанії для передбачення поведінки клієнтів, аналізу їх уподобань та підбору, а також оцінки схожості між клієнтами, шляхом впровадження та інтеграції її в якості стороннього сервісу, у виді виділеного серверу, в інші. З цього випливає, що система має бути побудована у вигляді клієнт-серверної архітектури, а основного компоненту – монолітна. А, оскільки, інформація поступає через REST API, в архітектурі системі було виділено менше уваги інтерфейсу для користувача. В даному випадку, ними, тобто клієнтами цієї системи, а не об'єкти, якими оперує система, виступають, під'єднані до серверу по REST API, засоби взаємодії з системою через інтерфейс, або ж, власне, REST API.

Найпоширенішою схемою архітектури є багатошарова архітектура, що відповідає традиційним ІТ-комунікаціям та організаційним структурам, що існують у більшості компаній, що робить його природним вибором для більшості бізнес-додатків. Шаблон архітектури є загальною схемою, що робить його гарною відправною точкою для більшості додатків. Однак має великий недолік – воронку запитів. Цим пояснюється ситуація, коли запити проходять через кілька шарів архітектури як просту пропускну обробку, з невеликою або без логіки, що виконуються в межах кожного шару. Ще одним важелем у розгляді даної структури архітектури є те, що може бути застосована в монолітних додатках. Хоча це може спричинити певні проблеми з точки зору розгортання, загальної надійності та надійності, продуктивності та масштабованості.

Подіє-орієнтована архітектура є популярною асинхронною архітектурою, що використовується в високо-масштабованих застосунках, що адаптована для використання, як в малих, так і в великих. Дана архітектура складається з декількох відокремлених компонентів, що обробляють події, асинхронно отримуючи та обробляючи їх.

Архітектура мікроядра є природним патерном для реалізації застосунків на основі продуктів. Даний тип архітектури складається з ядра, в якому лежить основна

логіка роботи програми, та декількох модулів або плагінів, що розширюють базовий функціонал.

Мікросервісна архітектура дуже швидко розвивається завдяки своїй здатності розбиття програми на незалежні автономні компоненти, кожен з яких відповідає за свій призначений функціонал. Основною перевагою над іншими прикладами архітектури є здібність до розгортання окремого компоненту в реальному часі без загрози завдати шкоди всій системі.

Таблиця 3.1 – Порівняльна характеристика[18] архітектур ПЗ

	Пристосування до змін	Легкість розгортання	Здатність до тестування	Продуктивність	Масштабованість	Легкість розробки
Багатошарова	Низька	Низька	Висока	Низька	Низька	Висока
Подіє-орієнтована	Висока	Висока	Низька	Висока	Висока	Низька
Мікроядерна	Висока	Висока	Висока	Висока	Низька	Низька
Мікросервісна	Висока	Висока	Висока	Низька	Висока	Висока

Найкращі показники серед описаних вище архітектур має мікросервісна. Однак в силу того, що описувана система складається з невеликої кількості компонентів, що мають швидко реагувати на запити від користувачів, та повинна бути розгорнута на будь-якому сервері, було обрано подіє-орієнтовану архітектуру, як основу для застосування системи.

3.2 Сценарії використання системи

Діаграма варіантів використання системи наведена в додатку В. Опис сценаріїв наведено нижче.

Таблиця 3.2 – Сценарій використання 1

Назва	Авторизувати
ID	1
Опис	Будь-який користувач повинен авторизуватися аби отримати доступ до функцій системи
Актори	Неавторизований користувач
Вигоди компанії	Доступ до системи можуть отримати тільки ті користувачі, що мають відповідні права.
Частота користування	Постійно
Тригери	Неавторизований користувач виконує запит
Передумови	Наявність логіну, паролю користувача
Постумови	Неавторизованому користувачу надано доступ до системи
Основний розвиток	Неавторизований користувач виконує запит і передає дані про користувача.
Альтернативні розвитку	–
Винятки	–

Таблиця 3.3 – Сценарій використання 2

Назва	Зареєструвати користувача
ID	2
Опис	Для отримання доступу до системи будь-який користувач має зареєструватися. Необхідність додавання нового користувача
Актори	Адміністратор
Вигоди компанії	Контроль користувачів, що потребують доступу
Частота користування	Часто
Тригери	Адміністратор виконує запит
Передумови	Користувач авторизований як Адміністратор. Наявність логіну, паролю користувача.
Постумови	Створено неавторизованого користувача в системі
Основний розвиток	Адміністратор виконує запит і передає дані про користувача, після чого новий користувач має можливість отримати доступ до системи
Альтернативні розвитку	–
Винятки	–

Таблиця 3.4 – Сценарій використання 3

Назва	Відновити пароля користувача
ID	3
Опис	Для відновлення доступу користувача. Неавторизованому користувачу необхідно отримати доступ до функцій
Актори	Адміністратор
Вигоди компанії	Не втрачати зареєстрованих користувачів
Частота користування	Рідко
Тригери	Адміністратор виконує запит
Передумови	Користувач авторизований як Адміністратор. Наявність логіну та паролю користувача.
Постумови	Дані користувачі змінено в системі
Основний розвиток	Адміністратор виконує запит і передає дані про користувача, після чого дані користувача змінено
Альтернативні розвитки	–
Винятки	–

Таблиця 3.5 – Сценарій використання 4

Назва	Вийти
ID	4
Опис	Для завершення доступу до системи.
Актори	Авторизований користувач
Вигоди компанії	Контроль одночасно авторизованих користувачів
Частота користування	Часто
Тригери	Користувач виконує запит
Передумови	Користувач авторизований
Постумови	Авторизованого користувача деавторизовано в системі
Основний розвиток	Користувач виконує запит і не має більше можливості використовувати функції системи
Альтернативні розвитку	—
Винятки	—

Таблиця 3.6 – Сценарій використання 5

Назва	Отримати інформацію про користувачів
ID	5
Опис	Для перегляду інформації про користувачів
Актори	Адміністратор
Вигоди компанії	Контроль і аналіз користувачів системи
Частота користування	Часто
Тригери	Адміністратор виконує запит
Передумови	Користувач авторизований як Адміністратор
Постумови	Адміністратор отримує інформацію про користувачів
Основний розвиток	Адміністратор виконує запит, після чого системи повертає інформацію про користувачів
Альтернативні розвитку	—
Винятки	—

Таблиця 3.7 – Сценарій використання 6

Назва	Видалити користувача
ID	6
Опис	Для видалення користувача. Для необхідності заборони доступу до системи або видалення користувача.
Актори	Адміністратор
Вигоди компанії	Контроль користувачів системи
Частота користування	Рідко
Тригери	Адміністратор виконує запит
Передумови	Користувач авторизований як Адміністратор. Наявність даних про користувача, що потребує видалення.
Постумови	Користувача буде видалено
Основний розвиток	Адміністратор виконує запит, передає інформацію про користувача, що необхідно видалити, після чого користувача буде видалено.
Альтернативні розвитку	—
Винятки	—

Таблиця 3.8 – Сценарій використання 7

Назва	Додати нову роль користувача
ID	7
Опис	Для обмеження або розширення можливостей користувачів. Для створення ролі, що об'єднує можливостей декількох ролей.
Актори	Адміністратор
Вигоди компанії	Розширення можливостей користувачів. Обмеження можливостей користувачів. Контроль користувачів системи.
Частота користування	Рідко
Тригери	Адміністратор виконує запит
Передумови	Користувач авторизований як Адміністратор. Наявність даних про нову роль.
Постумови	Додано нову роль
Основний розвиток	Адміністратор виконує запит, передає інформацію про роль, що необхідно додати, після чого роль буде доданою
Альтернативні розвитки	–
Винятки	–

Таблиця 3.9 – Сценарій використання 8

Назва	Видалити роль
ID	8
Опис	Для видалення ролі.
Актори	Адміністратор
Вигоди компанії	Контроль користувачів системи
Частота користування	Рідко
Тригери	Адміністратор виконує запит
Передумови	Користувач авторизований як Адміністратор. Наявність даних про роль, що необхідно видалити.
Постумови	Роль буде видалено
Основний розвиток	Адміністратор виконує запит, передає інформацію про роль, що необхідно видалити, після чого роль буде видалено
Альтернативні розвитки	–
Винятки	–

Таблиця 3.10 – Сценарій використання 9

Назва	Тренувати НМ
ID	9
Опис	Для тренування НМ
Актори	Адміністратор
Вигоди компанії	Зміна точності та якості системи прогнозування уподобань клієнта
Частота користування	Рідко
Тригери	Адміністратор виконує запит
Передумови	Користувач авторизований як Адміністратор
Постумови	Тренування НМ буде розпочато
Основний розвиток	Адміністратор виконує запит, після чого тренування НМ буде почато
Альтернативні розвитки	–
Винятки	–

Таблиця 3.11 – Сценарій використання 10

Назва	Отримати інформацію про НМ
ID	10
Опис	Для перегляду інформації про НМ. Застосовується для отримання інформації про конфігурації НМ та статус.
Актори	Адміністратор
Вигоди компанії	Перегляд актуальної інформації точності та якості системи прогнозування уподобань клієнта.
Частота користування	Рідко
Тригери	Адміністратор виконує запит
Передумови	Користувач авторизований як Адміністратор
Постумови	Система поверне інформацію про НМ
Основний розвиток	Адміністратор виконує запит, після чого система поверне інформацію про НМ
Альтернативні розвитки	–
Винятки	–

Таблиця 3.12 – Сценарій використання 11

Назва	Додати клієнта
ID	11
Опис	Для додавання нового клієнта до системи
Актори	Авторизований користувач
Вигоди компанії	<p>Додавання нової інформації.</p> <p>Збільшення якості і точності системи прогнозування та порівняння.</p> <p>Пошук нових прихованих груп клієнтів.</p> <p>Визначення прихованих особливостей клієнта.</p>
Частота користування	Постійно
Тригери	Авторизований користувач виконує запит
Передумови	<p>Користувач авторизований.</p> <p>Наявність даних про ознаки клієнта.</p> <p>Наявність даних про уподобання клієнта.</p>
Постумови	Клієнта буде додано, визначено його приховані особливості, додано до групи
Основний розвиток	Адміністратор виконує запит, після чого буде додано клієнта, визначено його приховані особливості, додано до групи
Альтернативні розвитку	–
Винятки	–

Таблиця 3.13 – Сценарій використання 12

Назва	Видалити клієнта
ID	12
Опис	Для видалення клієнта з системи
Актори	Авторизований користувач
Вигоди компанії	Видалення клієнтів з системи. Контроль клієнтів. Фільтрування помилкових та клієнтів, що не розглядаються, або не входять в цільову аудиторію.
Частота користування	Рідко
Тригери	Авторизований користувач виконує запит
Передумови	Користувач авторизований. Наявність даних про клієнта, що необхідно видалити.
Постумови	Клієнта буде видалено, перераховано групи
Основний розвиток	Адміністратор виконує запит, після чого буде видалено клієнта
Альтернативні розвитки	–
Винятки	–

Таблиця 3.14 – Сценарій використання 13

Назва	Змінити дані про клієнта
ID	13
Опис	Для зміни даних клієнта системи
Актори	Авторизований користувач
Вигоди компанії	Зміна інформації про клієнтів системи. Контроль клієнтів. Зміна існуючих уподобань. Зміна існуючих ознак.
Частота користування	Часто
Тригери	Авторизований користувач виконує запит
Передумови	Користувач авторизований. Наявність даних про клієнта, що необхідно змінити.
Постумови	Дані клієнта буде змінено, перераховано групи
Основний розвиток	Адміністратор виконує запит, після чого буде змінено дані клієнта
Альтернативні розвитки	–
Винятки	–

Таблиця 3.15 – Сценарій використання 14

Назва	Отримати інформацію про клієнтів
ID	14
Опис	Для перегляду інформації про клієнта
Актори	Авторизований користувач
Вигоди компанії	Контроль та аналіз клієнтів. Перегляд особливостей клієнта. Перегляд ознак клієнта.
Частота користування	Постійно
Тригери	Авторизований користувач виконує запит
Передумови	Користувач авторизований. Наявність даних про клієнта.
Постумови	Повернуто інформацію про клієнта
Основний розвиток	Адміністратор виконує запит, після чого буде повернуто системою
Альтернативні розвитки	–
Винятки	–

Таблиця 3.16 – Сценарій використання 15

Назва	Порівняти клієнтів
ID	15
Опис	Для порівняння клієнтів
Актори	Авторизований користувач
Вигоди компанії	Збільшення продажів. Зменшення витрат. Пошук нових прихованих груп клієнтів. Визначення прихованих особливостей клієнта.
Частота користування	Постійно
Тригери	Авторизований користувач виконує запит
Передумови	Користувач авторизований. Наявність даних про клієнтів.
Постумови	Отримано порівняльну оцінку ознак та уподобань клієнтів
Основний розвиток	Адміністратор виконує запит, після чого буде повернуто порівняльну оцінку ознак і уподобань клієнтів
Альтернативні розвитку	–
Винятки	–

Таблиця 3.17 – Сценарій використання 16

Назва	Отримати інформацію про утворені групи клієнтів
ID	16
Опис	Для перегляду інформації про групи клієнтів
Актори	Авторизований користувач
Вигоди компанії	Збільшення продажів. Зменшення витрат. Пошук нових прихованих груп клієнтів. Аналіз утворених груп клієнтів. Аналіз клієнтських уподобань.
Частота користування	Постійно
Тригери	Авторизований користувач виконує запит
Передумови	Користувач авторизований.
Постумови	Отримано інформацію про утворені групи клієнтів
Основний розвиток	Адміністратор виконує запит, після чого буде повернуто інформацію про утворені групи клієнтів
Альтернативні розвитки	–
Винятки	–

Таблиця 3.18 – Сценарій використання 17

Назва	Прогнозувати ознаки та уподобання ймовірного клієнта
ID	17
Опис	Для прогнозування інформації про ймовірного клієнта
Актори	Авторизований користувач
Вигоди компанії	Збільшення продажів. Зменшення витрат. Збільшення якості і точності системи прогнозування та порівняння. Пошук нових прихованих груп клієнтів. Визначення прихованих особливостей клієнта.
Частота користування	Постійно
Тригери	Авторизований користувач виконує запит
Передумови	Користувач авторизований. Наявність прихованих особливостей клієнта.
Постумови	Прогнозовано уподобання та ознаки ймовірного клієнта
Основний розвиток	Адміністратор виконує запит, після чого прогнозовано уподобання та ознаки і повернуто системою
Альтернативні розвитку	–
Винятки	–

Таблиця 3.19 – Сценарій використання 18

Назва	Прогнозувати приховані особливості клієнта
ID	18
Опис	Для прогнозування інформації про клієнта
Актори	Авторизований користувач
Вигоди компанії	Збільшення продажів. Зменшення витрат. Збільшення якості і точності системи прогнозування та порівняння. Пошук нових прихованих груп клієнтів. Визначення прихованих особливостей клієнта.
Частота користування	Постійно
Тригери	Авторизований користувач виконує запит
Передумови	Користувач авторизований. Наявність ознак клієнта. Наявність уподобань клієнта
Постумови	Прогнозовано приховані особливості ймовірного клієнта
Основний розвиток	Адміністратор виконує запит, після чого прогнозовано приховані особливості і повернуто системою
Альтернативні розвитки	–
Винятки	–

Таблиця 3.20 – Сценарій використання 19

Назва	Прогнозувати ймовірні уподобання клієнта
ID	19
Опис	Для прогнозування інформації про клієнта
Актори	Авторизований користувач
Вигоди компанії	Збільшення продажів. Зменшення витрат. Збільшення якості і точності системи прогнозування та порівняння. Пошук нових прихованих груп клієнтів. Визначення прихованих особливостей клієнта.
Частота користування	Постійно
Тригери	Авторизований користувач виконує запит
Передумови	Користувач авторизований. Наявність ознак клієнта.
Постумови	Прогнозовано уподобання ймовірного клієнта
Основний розвиток	Адміністратор виконує запит, після чого буде прогнозовано уподобання і повернуто системою
Альтернативні розвитки	–
Винятки	–

Таблиця 3.21 – Сценарій використання 20

Назва	Прогнозувати ймовірні ознаки клієнта
ID	20
Опис	Для прогнозування інформації про клієнта
Актори	Авторизований користувач
Вигоди компанії	Збільшення продажів. Зменшення витрат. Збільшення якості і точності системи прогнозування та порівняння. Пошук нових прихованих груп клієнтів. Визначення прихованих особливостей клієнта.
Частота користування	Постійно
Тригери	Авторизований користувач виконує запит
Передумови	Користувач авторизований. Наявність уподобань клієнта
Постумови	Прогнозовано ознаки ймовірного клієнта
Основний розвиток	Адміністратор виконує запит, після чого буде прогнозовано ознаки і повернуто системою
Альтернативні розвитки	–
Винятки	–

Для оцінки якості діаграми варіантів використання необхідно проаналізувати покриття вимог до інтелектуальної системи сценаріями використання.

В таблиці 3.22 наведено покриття вимог до системи сценаріями використання. Рядки таблиці представляють собою функціональні вимоги, наведені у підрозділі 1.4, а стовпці – описані вище сценарії використання, що представлено на діаграмі варіантів використання в додатку Г.

Таблиця 3.22 – Покриття вимог до системи сценаріями використання

Сценарій \ Вимога	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
а		+	+		+	+														
б	+			+																
в							+	+												
г											+	+	+							
г														+						
д																+				
е															+					
є																			+	+
ж																			+	
з																	+	+		
и									+	+										

Як видно з вищенаведеної таблиці видно, що всі функціональні вимоги до системи покрито сценаріями використання. Декілька вимог покриваються одночасно декількома сценаріями використання, оскільки такі вимоги були неточними і потребували деталізації. Один сценарій покриває одночасно дві вимоги, через те, що він реалізує одну і ту ж функцію для обох вимог.

3.3 Розробка структурної схеми

Структурна схема інтелектуальної системи наведена в додатку Д.

Основним компонентом системи є серверний застосунок. Незважаючи на те, що система побудована на монолітній структурі, його було розділено на декілька ключових складових, що є незалежними один від одного.

В ході проектування системи було встановлено необхідність в відокремленні деталей реалізації складових від основної логіки, а також для досягнення максимальної швидкодії. Це пов'язано з вимогами до системи та уникненням затримки від моменту запиту та повернення відповіді. Тому для реалізації цього компоненту було використано подіє-орієнтовану архітектуру. В основі цього лежить використання однієї шини для обміну внутрішніх повідомлень. Основними об'єктами для роботи з шиною є відправник та підписник. В залежності від кількості останніх, в шині створюється група очікування, асоційована з певною подією. В момент відправки повідомлення в шині воно дублюється, підкріплюється міткою підписника і «передається». Завдяки вбудованій підтримці асинхронності, реалізувати це можна за допомогою, так званих каналів, що поєднують паралельні задачі. Для забезпечення послідовності подій, ініціатор створює ідентифікатор ланцюга, що додається в подію. Такий підхід дає змогу слідкувати за багатьма подіями одного типу одночасно.

Всі компоненти системи побудовані таким чином, щоб можна було використовувати як і виділений сервер, так і локальний для кожного з компонентів. Це підвищує масштабованість та захищеність системи.

Файл конфігурації системи дозволяє змінювати доступи до різних компонентів програми чи обмеження деяких функцій. Оскільки всі модулі є незалежними, їх можна вимикати у вказаному файлі.

Обмін інформацією в реальному часі відбувається за допомогою окремого брокера повідомлень, що також дає можливість використовувати віддалену одиницю. Необхідний для забезпечення користувачів та сервісів актуальною інформацією в реальному часі задля того, щоб вони могли виконувати свої алгоритми в той саме час,

коли було завершена якась дія в даній системі. Доступи до брокеру повідомлень вказуються в файлі конфігурації системи.

Дані про клієнтів, уподобання та ознаки зберігаються в базу даних, розташованому на тому ж сервері, однак є можливість вказати віддалену, якщо, наприклад, компанія надає перевагу тримати всі компоненти к окремому місці, що вказується в файлі конфігурації системи.

Головний модуль забезпечує загальну логіку роботи системи, вирішує який алгоритм роботи треба виконати в залежності від події, що відбулась, чи після обробки запиту від сервісів системи впровадження. В деяких алгоритмах роботи, система продукує повідомлення до брокеру повідомлення, оскільки підключені користувачі повинні отримувати інформацію в реальному часі, таким чином, даний модуль з'єднаний з брокером повідомлень і сам, наразі, є виробником. В залежності від алгоритму роботи, в системі використовується тип комунікації «вентилятор», який розсилає повідомлення одразу всім підключеним «споживачам», а сам сервер, на разі, є «виробником» та прямий тип, відправляє повідомлення тільки тому споживачу, ідентифікатор якого буде в повідомленні. З'єднаний з шиною. Реагує на події, асоційовані з його підписниками, а в кінці роботи функцій чи алгоритмів роботи породжує нові подію. Оскільки завжди є ініціатором ланцюга повідомлень, зберігає ідентифікатора в своєму стані, з яким асоційовано також, ідентифікатор запиту. Включає в себе такі компоненти, як сервер, маршрутизатор, налаштування можна змінити в файлі конфігурації системи.

Модуль журналювання призначений для обробки журналів, які продукує система в процесі роботи. Це може відбуватися під час тренування нейронних мереж, в момент отримання та після обробки запиту, перед записом до бази даних чи прогнозування інтелектуальною частиною системи. Кожен з модулів, що має необхідність в журналюванні відсилає свій ідентифікатор для того, щоб в системі журналювання можна було розрізнити джерело. В системі розрізняють чотири рівні журналювання: «Вимкнено», «Консоль», «Зовнішній», «Всі». Найнижчий вимикає журналювання, «Консоль» відтворює журнали в консоль, в якій запускалася система. «Зовнішній»

відсилає журнали в віддалену систему для зберігання. При обраному рівні «Всі» журнали одночасно і відтворюються в консоль і відправляються. Для зберігання журналів, які продукує система, було використано рішення, яке побудоване за клієнт-серверною архітектурою та має можливість отримувати їх в реальному часі. Змінювати рівні та доступи до системи журналювання можуть змінюватися в файлі конфігурації системи. З'єднаний з шиною. Реагує на подію, асоційованою з його підписником, а в кінці роботи функції породжує нову подію.

Модуль веб-інтерфейсу дає можливість використовувати вбудований веб-інтерфейс, який може використовуватися адміністратором або звичайний користувачем, що грає роль клієнта та обмінюється інформацією через REST API та брокер повідомлень. Завдяки тому, що API для користувача базової ролі доступне також без веб-інтерфейсу, компанія-замовник має можливість реалізувати власний клієнт для доступу до сервісу або безпосередньо інтегрувати в систему аналітики або маркетингу. Реагує на подію, асоційованою з його підписником, а в кінці роботи функції породжує нову подію.

Веб-інтерфейс реалізує базові функції системи, тобто відтворює варіанти використання, наведені в додатку Г, а також отримує актуальну інформацію та статуси з системи через брокер повідомлень. Для підтримки даної функції було розроблено розширення, що забезпечує комунікацію між браузером та брокером повідомлень. З'єднаний з шиною.

Модуль взаємодії з базою даних системи впровадження дає можливість отримати доступ до сховища даних, де зберігаються первинні дані про клієнтів. Підтримує доступи до найбільш популярних баз даних, проте може розширено шляхом додавання необхідного адаптера в код програми. З'єднаний з базою даних інтелектуальної системи, оскільки отримані дані про клієнтів зберігає у сховищі системи. З'єднаний з шиною. Реагує на подію, асоційованою з його підписником, а в кінці роботи функції породжує нову подію. Доступи до баз даних системи впровадження можуть бути змінені в файлі конфігурації системи.

Модуль для взаємодії з системою навчання нейронних мереж для аналізу клієнтських уподобань необхідний для постійного навчання інтелектуального забезпечення системи для підвищення якості та точності системи. З'єднаний через віддалений виклик процедур з системою навчання нейронних мереж. Реагує на подію, асоційовану з його підписником, а в кінці роботи функції породжує нову подію. Конфігурації можуть бути змінені у відповідному файлі.

Система для навчання нейронних мереж представляє собою виділений сервіс для того, щоб відокремити процес навчання від основної роботи системи. Може бути розташована на сервері застосунку або на окремому, аби не заважати роботі системи, що розробляється. Такий підхід дозволяє одночасно виконувати дві функції. З'єднаний з базою даних системи, оскільки потребує тестові та тренувальні дані для навчання. З'єднаний з модулем взаємодії з системою навчання через віддалений виклик процедур. Складається з механізму обробки виклику віддалених процедур, моделі клієнта та адаптеру для доступу до бази даних.

Модуль для аналізу клієнтських уподобань є основним інтелектуальним забезпеченням, в якому реалізовано модель та механізм аналізу клієнтських уподобань та ознак, на основі яких система прогнозує приховані особливості притаманні клієнту. Використовується для прогнозування, як і ймовірних ознак, так і ймовірних уподобань клієнта, виходячи з наявних даних у користувача. Зчитує та записує в базу дані про клієнтів, у випадку прогнозування прихованих особливостей. Для оновлення моделі клієнта може бути перезавантажений, що дає можливість не перезавантажувати всю систему в цілому. З'єднаний з шиною. Реагує на події, асоційовані з його підписниками, а в кінці роботи ініціює нову подію.

Модуль для компаративної оцінки використовується для утворення груп клієнтів та порівняльної оцінки між ними. Взаємодіє з базою даних для отримання такої інформації про клієнтів як ознаки, уподобання чи приховані особливості, що використовуються в кластеризації, порівнянні клієнтів та отриманні найближчих схожих клієнтів. З'єднаний з шиною. Реагує на події, асоційовані з його підписниками, а в кінці роботи ініціює нову подію.

3.4 Розробка функціональної схеми

Функціональна схема інтелектуальної системи наведена в додатку Е.

Інтелектуальна система представляє собою складну систему з підтримкою масштабованості та асинхронності. Кожне із складових системи виконує свої функції та забезпечує виконання вимог, наведених в підрозділі 1.3.

В основі обробки запитів лежить використання швидкого маршрутизатора. Кожен з модулів при ініціалізації при необхідності чи наявності реєструє власні маршрути, таким чином розмежуючи функції модулів. Кожен маршрут відноситься до маршрутної групи, яка є набором посередників, утворюючи ланцюг. Перед обробкою запита безпосередньо, він проходить ланцюг посередників групи, до якої відноситься маршрут і викликає обробника. У випадку помилки повертає відповідний код. Кожен із маршрутів може мати окремі посередники, надані модулем, а також права, що дозволяють або забороняють виконання його деяким ролям або користувачам.

Для доступу з різних джерел до системи, була впроваджена авторизація. Система розрізняє наступні ролі: «Адміністратор», «За замовчуванням» та «Сервіс». Користувачі типу «Адміністратор» мають можливість додавати нові ролі і закріплювати за ними необхідні права, реєструвати інших користувачів в системі. Особливостями групи «Адміністратор» є можливість моніторингу поточного стану всіх компонентів системи. Користувачі типу «За замовчуванням» мають права такі, як додавання нового клієнта, визначення його вподобань, а також прогнозування ймовірних ознак можливого клієнта, виходячи з обраних уподобань. Останньою вбудованою роллю є «Сервіс». Користувачі, що входять до її складу, є сервісами компанії, яка використовує дану систему у вигляді інтегрованого, що дозволяє визначати уподобання в поточний момент часу. Такий спосіб розмежування ролей дозволяє відокремити департамент аналітики або маркетингу від технічного.

Шина грає основну роль в обміні інформації між модулями. Кожен із модулів для передачі даних формує необхідну подію, а у випадку отримання створює підписника. В результаті чого, повідомлення відправляється одночасно всім підписникам.

Модуль журналювання виконує функції зберігання інформації про дії системи в цілому для віддаленого моніторингу системи. Також може бути використаний для моніторингу помилок, що трапляються під час роботи програмного забезпечення. Після реагування підписника на відповідну подію, відправляє дані в повідомленні до системи журналювання.

Модуль взаємодії з базою даних системою впровадження виконує роль інтегратора та дає можливість отримувати дані про клієнтів напряму. Після реагування підписником на подію, визначається база даних системи впровадження за схеми таблиць, у випадку реляційних СУБД, або колекцій, у випадку нереляційних. Виконує запит та отримує множину клієнтів, після чого переформовує в необхідний для системи формат, зберігаючи відносні ідентифікатори, тощо. Після чого записує її до бази даних і формує подією в шині.

Модуль веб-інтерфейсу дає можливість виконувати базові операції і є вбудованим в систему. Однак, як було зазначено вище, користувач, авторизований через вбудований інтерфейс відноситься до ролі «За Замовчуванням», яка дає можливість доступу до функцій системи, тому компанія-замовник може реалізувати власний веб-інтерфейс або інтегрувати в існуючу систему аналітики чи систему управління ресурсами. Після реагування на подію, модуль ініціалізує маршрут, що повертає контент веб-інтерфейсу.

Модуль для взаємодії з системою навчання НМ має можливість накопичення тренувальних та тестових даних, необхідних для тренування НМ. Реагує на подію, після чого відправляє запит на систему для навчання НМ. У випадку отримання відповіді, зберігає конфігурацію НМ і формує подію в шині.

Система навчання НМ виконує роль обгортки над тестовою моделлю клієнта. Після реагування на запит, отримує тестові та тренувальні дані з бази даних та тренує модель, після закінчення чого, формує відповідь.

Модуль для аналізу клієнтських уподобань виконує основну інтелектуальну роль, маючи можливість прогнозування прихованих особливостей або ознак, в залежності від випадків і запису результатів в базу даних. Має такі функції, як: прогнозу-

вання прихованих особливостей за ознаками і навпаки, додавання нових клієнтів, визначивши необхідні показники. Перед початком прогнозування виконує категоризацію ознак клієнта, формуючи необхідні вхідні дані для НМ. Також має можливість перезавантаження.

Модуль компаративної оцінки виконує роль компаратора між клієнтами. Дає можливість знаходити групи клієнтів за ознаками, прихованими особливостями та уподобаннями, або виконувати порівняльну оцінку клієнтів. Має функції визначення найближчого клієнта по уподобаннях, ознаках або прихованих особливостей, додавання клієнта до визначеної групи, виконання розбиття та нове розбиття вже існуючих клієнтів та оновлення необхідних даних в базі даних. У випадку кластеризації по ознакам виконується категоризація і визначаються координатні прямі та площини у просторі

Головний модуль виконує функцію об'єднання функцій модулів шляхом виконання алгоритму роботи. Вбудований сервер обробляє запити від користувачів, та передає до ланцюга посередників, після якої визначається алгоритм роботи, в результаті якого формується відповідь. Підписники, створені модулем, реагуючи на подію, викликають обробник події. Внаслідок чого визначається і виконується алгоритм роботи, а після – формується подія в шині. Авторизація та валідація реалізована шляхом додавання посередника. Під час виконання роботи використовує брокер повідомлень для відправки статусу системи. Якщо, наприклад, модуль для аналізу клієнтських уподобань перезавантажується, відправляється відповідний статус, додається посередник, що забороняє обробку запитів, а після вдалого перезавантаження відправляє новий статус і відновлює роботу системи. Такий підхід дає можливість користувачам раціонально використовувати ресурси. Забезпечує менеджмент користувачів, клієнтів та ролей. В разі необхідності видалити або змінити інформацію про клієнта, формує події на виконання нового часткового розбиття та оновлення запису в базі даних.

3.5 Інструменти, мови та засоби реалізації

3.5.1 Вибір мови програмування системи

Основною мовою програмування даної системи та її компонентів було обрано Go, розроблену компанією Google у 2009 році [16]. Вона є статично типізованою, компільованою та має такі вигоди, як вбудований збірник сміття, безпекою пам'яті та структурованої типізації. Відноситься до таких парадигм, як функціональна, процедурна, імперативна та паралельна. Основною перевагою є наявність вбудованої підтримки виконання великої кількості задач, що потребують незначних ресурсів. Наприклад, у порівнянні з мовою програмування Java, де необхідно для створення нового потоку виділити 1 Мб з кучі, дана мова потребує 2 Кб нової задачі, та ініціалізується за менший час. Виходячи з того, що Go є компільованою, вона не залежить від середовищ, наприклад, віртуальної машини або фреймворку, у випадку з C#, Java, JavaScript, Python, тощо. А завдяки вбудованій можливості перевіряти код на невикористовувані змінні, вона не виділяє зайву непотрібну пам'ять. На відміну від таких мов програмування, як C#, Go доступна на трьох основних платформах, таких як, Mac OS, Windows та Linux, а також, для мобільних пристроїв – Android та iOS, з використанням необхідного пакету програмного забезпечення.

Недоліком мови програмування Go є відсутність ООП, хоча в ній присутнє наслідування інтерфейсів структурами, на противагу чому є висока продуктивність та обробка багатозадачності. Проаналізувавши порівняння продуктивності мов програмування, таких як, JavaScript, Python, C++, Java, Go на обчисленні різних алгоритмах, фрагменти якого наведені в таблиці 3.20, можна визначити, що Go має істотну перевагу над Python та JavaScript, в деяких алгоритмах має однакову з Java продуктивність та відстає, у порівнянні з C++. Однак слід зазначити, що швидкість розробки системи на мові C++ істотно перевищує за обрану в даній системі, що є досить впливовим важелем у виборі та зважаючи на те, що описувана система є серверною архітектурою, рідше використовується. Порівняння [17] проводилось на обчислювальному пристрої з такими характеристиками:

- процесор з 4-ма ядрами Intel Q6600 з частотою 2.4 ГГц;
- об'єм оперативної пам'яті – 4 Гб;
- жорсткий диск SATA II об'ємом 250 Гб;
- операційна система Ubuntu™ 18.10 Linux x64 4.18.0-10-generic.

Таблиця 3.20 – Порівняльна характеристика серверних мов програмування

Мова	t , с	C , Кб	R , б	Завантаження ядер процесора, %			
				1	2	3	4
Алгоритм pidigits							
Go	2.04	8.38	603	2	1	1	100
NodeJs	14.27	63.108	530	65	1	1	36
Java	3.13	37.376	938	1	98	4	6
C++	1.89	4.272	513	1	3	100	2
Python 3	3.38	10.352	386	1	1	0	100
Алгоритм mandelbrot							
Go	5.47	31.004	905	100	99	99	99
NodeJs	18.18	632,008	748	88	94	95	83
Java	6.96	76.316	796	97	97	97	97
C++	1.81	29.072	1002	97	98	98	100
Python 3	263.04	51.780	688	100	100	100	100

Де t – час виконання, C – об'єм використовуваної пам'яті, а R – розмір стиснутого коду у форматі gzip.

Ще одним істотним важелем у виборі, вищеприписаної мови програмування, є наявність її у вимогах до системи.

3.5.2 Вибір мови програмування для досліджень та навчання

Для досліджень над нейронними мережами та в компонентах системи, де проходить процес навчання було використано мову програмування Python з використанням додаткового фреймворку для машинного навчання Keras. Вибір обумовлений тим, що його застосовують у поєднанні з TensorFlow або Theano у вигляді обгортки, що полегшує процес розробки та дослідження, що проводяться стосовно системи, перейнявши переваги. При розробці даної системи було обрано TensorFlow, в якості бекенду для роботи з нейронними мережами, а також, у зв'язку з тим, що натреновані моделі можна завантажити в додаток, написаний на мові програмування Go, що є істотним важелем в описуваній системі.

3.5.3 Вибір сховища для зберігання даних

Сховищем для зберігання даних було обрано нереляційну базу даних MongoDB, оскільки дані про клієнтів, тобто ознаки, представляються нечіткими структурами, як і уподобання, аналогічно до систем слідкування в реальному часі, зберігання журналів, аналітики тощо, а для зберігання інформації про ознаки можна зберігати у раціональній базі даних, проте у випадку даної системи це є недоцільним.

В такому типі бази даних записи зберігаються у вигляді документів, а відношення, на відміну від реляційних, де це представляється таблицями, виражається дочірнім піддокументом. Кожен рядок представляє собою запис подібний до формату JSON. Нереляційні бази даних мають більшу швидкість читання та запису, оскільки, в реляційних базах даних є етап відповідності типу.

Оскільки ознаки клієнта є неточними та можуть постійно змінюватися в процесі розробки системи або навчання її інтелектуальної частини, було обрано саме такий тип бази даних. Зміна структури документа є однією з її основних переваг, а у випадку реляційних для вирішення цього завдання необхідно було б змінювати схеми таблиць, що свідчить про затрати часу в процесі розробки.

3.5.4 Вибір брокера повідомлень

Ще одним засобом реалізації системи був обраний брокер повідомлень RabbitMQ, вибір якого обумовлений простотою, певною швидкістю до 10000 повідомлень на секунду, чого достатньо, для задоволення потреб та відповідності вимогам системи.

Наявність такого засобу в системі є необхідним у зв'язку з відтворенням інформації в реальному часі без суттєвих затримок. Система виступає в ролі відправника повідомлень, а підключені користувачі – отримувачами. Аналогічно до цього принципу, можна було реалізувати окремий компонент для обміну повідомлень, однак, в такому випадку виникає технічний бар'єр, обумовлений відсутністю можливостей реалізації такого підходу, або ж обмеженнями, які мають підключені користувачі. Тому, доцільно використати готове рішення, беручи до уваги існування імплементації під'єднання до брокера повідомлень на багатьох мовах, таких як, NodeJS, C#, Java, Go, C++, Python, тощо.

Обмін інформацією між підключеними пристроями та, власне, брокером повідомлень відбувається по здійснюється згідно протоколу AMQP 0-9-1(з англ – Advanced Message Queuing Protocol). Відповідно до цього протоколу, повідомлення публікуються в деяких обмінниках, які часто порівнюються з поштовими відділеннями. Обмінники потім поширюють копії повідомлень у черги за допомогою правил, які називаються прив'язками. Тоді брокери AMQP або надсилають повідомлення споживачам, що підписалися на черги, або споживачі отримують повідомлення з черг на вимогу. Відправляючий клієнт має назву «виробник», а той хто приймає повідомлення – «споживачем». У екосистемі AMQP існує велика кількість можливих структур для побудови комунікації. Для реалізації описуваної системи слід розглянути декілька з них, а саме – «прямий» та «вентилятор». Обидва підходи складаються з виробника, обмінника, черги та споживачів. Головною відмінністю є наявність ключа, що зв'язує чергу з деякою міткою, що характеризує споживача, а також вказання роутинг ключа при відправці повідомлення, яке співпадає з пов'язуючим ключем.

3.6 Алгоритми роботи

В залежності від запиту користувача в системі розрізняють декілька алгоритмів роботи, частина з яких показана в додатку Ж.

Розглянемо декілька алгоритмів роботи.

Додавання клієнта починається з того, що ознаки клієнта необхідно категоризувати, тобто визначити яке значення зі шкали категорії відповідає ознаці клієнта. Таким чином буде отримано вектор ознак. Наступним кроком є формування входу до НМ, де об'єднується в матриця уподобань та вектору ознак і передається на НМ. Далі по наданим значенням, НМ прогнозує приховані особливості клієнта, після чого клієнт розміщується в трьох просторах – уподобань, ознак та прихованих особливостей. В результаті клієнт закріплюється за визначеною групою.

Прогнозування ознак та уподобань на основі прихованих особливостей, використовуючи надані значення, починається з того, що формується вхід до НМ, після чого, він використовується як прихований шар, що використовує НМ для оберненого прогнозування видимого шару. Перед кінцем роботи алгоритму визначається найближчий клієнт з простору прихованих особливостей.

Прогнозування прихованих особливостей є частиною алгоритму додавання клієнту з відмінністю в тому, що не виконується додавання до простору нового клієнта.

Прогнозування уподобань починається з категоризації ознак, внаслідок чого обраховується через алгоритми кластеризації найближчий клієнт, що використовується як основа для твердження про ймовірні уподобання.

Порівняння клієнтів також з категоризації ознак, після чого обраховується відстань між клієнтами та визначаються спільні і відмінні критерії, тобто ознаки, уподобання чи групи до яких входять клієнти.

3.7 Діаграма послідовності

Діаграма послідовності інтелектуальної системи наведена в додатку К, з якої видно, що процес роботи починається з відправлення події ініціалізації в шину, внаслідок чого всі модулі виконують свої початкові функції. На діаграмі представлено послідовність дій в момент отримання запиту про інтеграцію.

Головний модуль отримуючи запит про інтеграцію формує подію в шину, одночасно формуючи відповідь. Модуль взаємодії отримує подію, після чого виконує запит до БД системи впровадження. Після отримання відповіді, записує в БД і формує відповідну подію. Після цього модуль журналювання реагуючи на подію, відправляє запит. Під час роботи цього циклу відправляються події про журналювання.

Одночасно з цим, головний модуль породжує подію навчання, модуль взаємодії в системою навчання реагуючи на неї, формує необхідні тестові та тренувальні дані і виконує запит до системи навчання, а в момент отримання відповіді, формує відповідну подію. Під час роботи цього циклу відправляються події про журналювання.

Головний модуль ініціює подію про початок визначення прихованих особливостей отриманих клієнтів. Після того, як було визначено приховані особливості, модуль аналізу клієнтських уподобань формує подію. Під час роботи цього циклу відправляються події про журналювання.

Модуль компаративної оцінки, реагуючи на подію закінчення визначення прихованих особливостей, починає виконувати розбиття. По завершенню формує подію в шину про закінчення розбиття. В кінці головний модуль отримує подію про завершення розбиття та відправляє в брокер повідомлення про завершення інтеграції. Під час роботи цього циклу відправляються події про журналювання.

Модуль журналювання реагуючи на свою подію, відправляє запит до системи журналювання.

Таким чином, було описано послідовність дій при запиті про інтеграцію в систему аналітики компанії-замовника.

3.8 Розгортання системи

Діаграма послідовності інтелектуальної системи наведена в додатку Л.

Згідно вимогам, наведених у підрозділі 1.3, система повинна мати простий процес розгортання системи. Вибрана мова програмування для системи для супутні рішення дозволяють задовольнити описану вимогу. В результаті компіляції основної програми отримуються один файл для запуску та один файл конфігурації до системи, з чого випливає, що система може бути розгорнута на будь-якому сервері. Так інструменти, як брокер повідомлень RabbitMQ, база даних MongoDB та система навчання НМ можуть бути розгорнуті, як на цьому ж сервері, так і окремо, для надання більшого програмного простору інтелектуальній системі.

Першим вузлом на діаграмі є пристрій, на якому користувач відтворює веб-інтерфейс, що може бути персональним комп'ютером. Він з'єднаний з брокером повідомлень через TCP/IP, поверх якого реалізується протокол AMQP, а до самого додатку під'єднаний по HTTP.

Система для навчання нейронних мереж з'єднана по TCP/IP, оскільки дані між ним та додатком передається по ньому. Може бути запущена на будь-якому не зайнятому порті.

Додаток розгортається як на окремому сервері, так і на тому, де інстальовані сукупні технології. Оскільки серед складових системи є сервер для прийому запитів, виникає необхідність в запуску на вибраному порті, що вказується в файлі конфігурації системи. Завдяки такому підходу можна налаштовувати всі мережеві параметри, а саме порт сервера, режим доступу до бази даних та брокеру повідомлень, а також системи навчання НМ. Також існує можливість змінити доступ до системи журналювання, що знаходиться в сторонній системі.

Всі сервіси в системах, де впроваджується дана система виконують запити по HTTP та отримують інформацію в реальному часі через TCP/IP під'єднавшись до RabbitMQ.

3.9 Тестування системи

Для підтвердження проведеної роботи розроблену систему було протестовано. У таблиці 3.21 наведено тестовий сценарій менеджменту користувачів, авторизації та менеджменту ролей, а в 3.22 наведено тестовий сценарій менеджменту клієнтів.

Результатами тестового сценарію можуть бути: «Пройшов», «Не пройшов», «Заблокований»

Таблиця 3.21 — Тестовий сценарій 1.

Назва:	Тестовий сценарій 1	
Функція:	Менеджмент користувачів, авторизація та менеджмент ролей	
Дія	Очікуваний результат	Результат тесту:
Передумова:	Передумов немає	
Кроки тестування:		
Додати роль	Роль додано до бази даних	Пройшов
Зареєструвати користувача	Користувача додано до БД	Пройшов
Авторизувати користувача	Користувачу надано доступ до функцій системи	Пройшов
Вийти	Користувач втрачає доступ	Пройшов
Видалити користувача	Користувача видалено з БД	Пройшов
Видалити роль	Роль видалено з БД	Пройшов
Постумова:	Постумов немає	

Було успішно виконано тест таких функцій, як менеджмент користувачів, авторизації, та менеджменту ролей. В результаті тестування було встановлено, що система відповідає результатам тестування.

Постумов в даному тесті не передбачається.

Таблиця 3.22 — Тестовий сценарій 2

Назва:	Тестовий сценарій 2	
Функція:	Менеджмент клієнтів	
Дія	Очікуваний результат	Результат:
Передумова:	Передумов немає	
Користувач авторизується	Користувачу надано доступ	Пройшов
Кроки тестування:		
Додати клієнта	Клієнта додано до БД. Визначено приховані особливості. Додано до групи.	Пройшов
Отримати інформацію про клієнтів	Отримано список клієнтів та їх дані	Пройшов
Змінити інформацію про клієнта	Клієнта оновлено в БД. Перераховано приховані особливості. Додано до нової групи.	Пройшов
Отримати можливі уподобання клієнта	Визначено уподобання та отримано результат	Пройшов
Видалити клієнта	Видалено клієнта з БД. Перераховане розбиття.	Пройшов
Порівняти клієнтів	Отримано компаративну оцінку клієнтів	Пройшов
Постумова:	Постумов немає	

Було успішно виконано тест таких функцій, як менеджмент клієнтів, отримання уподобання та ознаки клієнтів, а також утворені групи. Перевірено роботу функції порівняння клієнтів. В результаті тестування було встановлено, що система відповідає результатам тестування.

У таблиці 3.23 наведено тестовий сценарій прогнозування, що покриває функції прогнозування.

Таблиця 3.23 — Тестовий сценарій 3

Назва:	Тестовий сценарій 3	
Функція:	Прогнозування особливостей за ознаками та уподобаннями, Прогнозування ознак за уподобаннями, Прогнозування уподобань за ознаками, Прогнозування ознак за уподобаннями	
Дія	Очікуваний результат	Результат:
Передумова:	Передумов немає	
Користувач авторизується	Користувачу надано доступ	Пройшов
Кроки тестування:		
Прогнозувати приховані особливості	Прогнозовано приховані особливості	Пройшов
Прогнозувати ознаки та уподобання клієнта	Прогнозовано ознаки та уподобання	Пройшов
Прогнозувати ознаки за уподобаннями	Визначено найближчого клієнта за уподобаннями та отримано його ознаки	Пройшов
Прогнозувати уподобання за ознаками	Визначено найближчого клієнта за ознаками та отримано його уподобання	Пройшов
Постумова:	Постумов немає	

Було успішно виконано тест таких функцій, як прогнозування особливостей за ознаками та уподобаннями, прогнозування ознак за уподобаннями, прогнозування уподобань за ознаками, прогнозування ознак за уподобаннями. В результаті тестування було встановлено, що система відповідає результатам тестування.

Постумов в даному тесті не передбачається.

У таблиці 3.24 наведено тестовий сценарій прогнозування, що покриває роботу з НМ.

3.10 Висновки

В даному розділі було описано структуру системи та її складових, а також функції кожного з них. Було проведено аналіз та обґрунтовано вибір засобів та інструментів, необхідних для реалізації. Описано сценарії використання сценаріїв та наведено таблицю покриття їм вимог до системи. Наведено основні алгоритми роботи системи та послідовність дій в залежності від функцій системи. Описано процес розгортання системи. Проведено тестування програмної частини системи, в результаті чого наведено таблицю покриття тестами варіантів використання. Описано процес взаємодії з системами журналювання, навчання НМ та інтеграції з сервісами системи компанії-замовника.

Система спроектована за принципами монолітної архітектури, а основний компонент на основі подіє-орієнтованої архітектури. Взаємодія між сервісами відбувається через REST та брокером повідомлень, в результаті чого, під'єднані користувачі отримують інформацію в реальному часі.

4 ОЦІНКА ЯКОСТІ СИСТЕМИ

Для визначення вдалості реалізації продукту, будь-яка система повинна бути протестована та оцінена, що було виконано в даному випадку.

4.1 Технічна оцінка

Було перевірено якість та швидкість отримання відповіді після запиту до системи за допомогою стороннього інструменту Apache Benchmark, звіт якого наведено в таблиці 4.1.

Таблиця 4.1 – Звіт Apache Benchmark

Тест	Кількість запитів	
	200	2000
Всього запитів	200	200
Провалено запитів	0	0
Успішно виконаних	200	2000
Рівень паралельності	100	200
Затрачений час на тестування, с	0.093	0.878
Всього передано, байт	33400	334000
Запитів на секунду	2140.23	2278.03
Час на один запит, мс	46.724	87.795
Час на один запит паралельний, мс	0.467	0.439
Швидкість передачі, Кбайт/с	349.04	371.51

Як видно з наведеної таблиці, система задовольняє вимогам, поставленим до неї. Система в незалежності від кількості одночасних запитів відповідає на всі, причому немає провалених, а час необхідний на виконання одного запиту збільшився лише в два рази, в той час, як кількість запитів в десять.

Для якісної оцінки роботи інтелектуальної системи шляхом аналізу даних, що були передані, та опитування клієнтів компанії, в систему якої було впроваджено дану інтелектуальну систему. Для цього було розбито оцінку на два етапи – оцінка роботи інтелектуальної частини та цілісну оцінку.

В той же час, інтелектуальна частина складається з двох компонентів, один з яких обраховує приховані особливості, а інший виконує компаративну оцінку клієнтів за їх уподобаннями, прихованими особливостями та ознаками.

Тестування компоненту для компаративної оцінки проводилося на вибірці з 8000 клієнтів з 20 категоріальними ознаками та 10 уподобаннями. В таблиці 4.2 наведено час, що був необхідним для розбиття вибірки клієнтів на 9 груп.

Таблиця 4.2 – Час розбиття 9 груп

Кількість клієнтів	Час, с
1000	31,3
2000	56.4
4000	95.4
6000	130.7
8000	231.1

Як видно з таблиці 4.2, на розбиття 8000 клієнтів на 9 груп було витрачено менше п'яти хвилин.

В таблиці 4.3 наведено залежність кількості груп для розбиття від витраченого часу 4000 клієнтів.

Таблиця 4.3 – Залежність часу від кількості груп

Кількість груп	Час, с
15	173.4
30	476.4
60	865.9

За даних, що наведено в таблиці 4.3 видно, що час зростає лінійно. Це пов'язано з тим, що для більшої кількості груп необхідно виконати більше операцій для знаходження найближчого клієнта, що значно сповільнює процес. Для кращої масштабованості вирішенням даної проблеми є використання модифікованих видів алгоритмів таких, як *scalable k-means* та *scalable k-prototype*.

4.2 Оцінка опитуванням

Для якісної оцінки прогнозування клієнтських уподобань було використано оцінку шляхом опитування клієнтів компанії, яким було прогнозовано уподобання та надано в контексті. Опитування було проведено три рази, в результаті якого було встановлено, що з 471 клієнта, що відповіли на нього, система прогнозування уподобань задовольнила потреби 79% опитуюючих. Така поведінка пояснюється тим, що в системі використовуються категоріальні ознаки, а кількісні або приводяться до категоріальних чи пропускаються. Вирішенням даної проблеми є можливість використання обмежених машин Больцмана з реальними значеннями, тобто знаходити приховані ознаки серед, як і категоріальних, так і якісних одночасно. Проте ця область ще не досліджена. Іншим фактором для такої точності може бути висока ймовірність нереальних ознак клієнта, що є людським фактором і не може бути об'єктивним.

4.3 Висновки

В цьому розділі було наведено результати оцінювання якості системи. Її було розбито на два етапи – оцінку технічного забезпечення та якісну, шляхом опитування реальних клієнтів. В результаті опитування було встановлено, що система на задовольняє прогнозування 20% клієнтів, що пояснюється неточними ознаками та недостовірними даними. З технічної оцінки випливає, що система час необхідний для розбиття на групи зростає лінійно, що може бути вдосконалено модифікаціями алгоритмів кластеризації.

5 РОЗРОБКА СТАРТАП ПРОЕКТУ

Стартап – це форма підприємства, яка шукає бізнес-модель, що задовольняє потреби, як споживачів, так і самої компанії. Бізнес-моделлю є процес створення цінності компанії шляхом продажу продуктів чи послуг. В стартапі розрізняють масштабовану відновлювану бізнес-модель, оскільки стартап – це невизначене підприємство і тому бізнес-модель має відповідати вказаним критеріям.

5.1 Опис ідеї проекту

Система впроваджується в системи аналітики та системи управління ресурсами інших компаній. Така система має можливість зменшити витрати, що виділені на пошук нових груп користувачів та клієнтів. Опис ідеї стартапу наведено в таблиці 5.1.

Таблиця 5.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Система впроваджується в інші системи, для аналізу клієнтів та формує групи, опираючись на приховані особливості, що дає можливість збільшити продажі та зменшити витрати.	Будь-яка система аналітики чи управління ресурсами	Формування нових груп користувачів або клієнтів.
		.

На ринку є декілька великих конкурентів-гігантів. Основним недоліком їх впровадження є наявність занадто великої кількості побочних можливостей та функцій.

Аналіз потенційних техніко-економічних переваг ідеї наведено в таблиці 5.2.

Таблиця 5.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів			W (слабка	N (нейтральна	S (сильна сторона)
		IC КОРУ	Adobe	СА			
1	Вартість	Середня вартість	Висока вартість	Висока вартість			+
2	Аналіз уподобань	Так	Так	Так		+	
3	Визначення прихованих особливостей	Так	Ні	Ні			+
4	Впровадження в системи	Так	Так	Ні			+
5	Формування груп	Так	Ні	Так			+
6	Наявність зайвих функцій	Ні	Так	Так			+

5.2 Технологічний аудит ідеї проекту

Для реалізації системи необхідно провести технічний аудит, що наведено в таблиці 5.3.

Таблиця 5.3 – Технологічна здійсненність ідеї проекту

№	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Головний модуль	Go, RabbitMQ, HTTP	Наявні, дороблювати не потрібно	Доступні, відкритий доступ.
2	Модуль для компаративної оцінки	Go	Наявні, дороблювати не потрібно	Доступні, відкритий доступ
3	Модуль для аналізу введень	Go	Наявні, дороблювати не потрібно	Доступні, відкритий доступ
4	Модуль веб-інтерфейсу	Go	Наявні, дороблювати не потрібно	Доступні, відкритий доступ.
5	Користувачський веб-інтерфейс	VueJS	Наявні, дороблювати не потрібно	Доступні, відкритий доступ.
6	Модуль для взаємодії з базами даних системи впровадження	Go	Наявні, дороблювати не потрібно	Доступні, відкритий доступ
7	База даних	MongoDB	Наявні, дороблювати не потрібно	Доступна, відкритий доступ

Продовження таблиці 5.3

№	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
8	Модуль журналювання	Go	Наявні, дороблювати непотрібно	Доступні, відкритий доступ, частковий доступ через авторизацію
9	Модуль взаємодії з системою навчання НМ для аналізу	Go, grpc	Наявні, дороблювати непотрібно	Доступні, відкритий доступ.
10	Система навчання НМ для аналізу клієнтських уподобань	Python3, grpc, Keras, Tensorflow	Наявні, дороблювати непотрібно	Доступні, відкритий доступ
11	Система журналювання	JS, RabbitMQ	Наявні, дороблювати непотрібно	Доступні, відкритий доступ
Обрані технології реалізації ідеї проекту: Go, RabbitMQ, MongoDB, Python3, Keras				

Всі технології для реалізації ідей проекту наявні та доступні, що свідчить про технологічну здійсненість системи. Слід зазначити, що система журналювання була реалізована окремо і була лише використана, додаючи додатковий функціонал.

5.3 Аналіз ринкових можливостей запуску

Аналіз попиту наведений в таблиці (5.4).

Таблиця 5.4 – Попередня характеристика потенційного ринку стартап-проекту

№	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	На ринку видимі конкуренти відсутні, є лише послуги для часткових рішень, що не задовольняють потреби компаній. На світовому ринку їх до 10
2	Загальний обсяг продаж	Ринок України у сфері САпоки є невизначеним, проте ринок систем управління ресурсами оцінюється більше, ніж у 35 мільйонів доларів. На світовому ринку до 40 мільярдів доларів
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Технічна обмеженість компанії-замовників
5	Специфічні вимоги до стандартизації та сертифікації	Відсутні
	Середня норма рентабельності в галузі (або по ринку), %	15-40

Визначення потенційних клієнтів наведено в таблиці 5.5.

Таблиця 5.5 – Характеристика потенційних клієнтів стартап-проекту

Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1. оптимізація закупок; 2. збільшення продаж; 4. зменшення витрат; 3. пошук нових груп клієнтів	Компанії, що мають системи аналітики та управління ресурсами 1. офлайн роздрібна торгівля; 2. онлайн роздрібна торгівля	1. ціна; 2. функціональність; 3. наявність схожих даних; 4. вартість впровадження	1. якість; 2. захищеність; 3. простота впровадження; 4. масштабованість; 5. інтеграція; 6. швидкість

Після визначення потенційних груп клієнтів проводиться аналіз ринкового середовища: складаються таблиця 5.7 та факторів, що сприяють ринковому впровадженню проекту, та таблиця 5.6 факторів, що йому перешкоджають

Таблиця 5.6 – Фактори загроз

№	Фактор	Зміст загрози	Можлива реакція компанії
1	Поява нових конкурентів	Поява конкурентів з подібними рішеннями.	зміна ціни; вирішення інших функцій;
2	Стан економіки	Економіка України є нестабільною і постійно падає, що знижує платоспроможність компаній	Оптимізація витрат та перевезення бізнесу за кордон

До позитивних факторів можна віднести незайнятий ринок аналізу даних клієнтів.

Таблиця 5.7 – Фактори можливостей

№	Фактор	Зміст можливості	Можлива реакція компанії
1.	Вільний ринок України	На ринку немає готових рішень	Простота виходу на ринок
2.	Неусвідомленість компаній	Багато компаній досі пір не усвідомлює значущість аналізу клієнтських уподобань	Реклама, прямі продажі

Надалі проводиться аналіз пропозиції: визначаються загальні риси конкуренції на ринку (таблиця 5.8).

Таблиця 5.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
Тип конкуренції: ринок України – монополістична, світовий – чиста	На ринку України відсутні рішення для аналізу клієнтських уподобань	Максимально розширювати межі та захоплювати сегменти компаній, що потребують даний продукт
Рівень боротьби: національний	Ринок аналізу клієнтів є по всій території України	Використання продукту на території України
За галузевою ознакою: внутрішньо-галузева	Конкуренція тільки в системах аналізу клієнтських даних	Моніторинг ринку
За характером переваг: цінова	ІС КОКУ має менше вартість у порівнянні з конкурентами	Зменшення ціни шляхом пошуку зниження собівартості послуги

Після аналізу конкуренції проводиться більш детальний аналіз умов конкуренції в галузі (за моделлю 5 сил М. Портера), що наведено у таблиці 5.9, з якої випливає що є велика можливість виходу на ринок, пропонуючи унікальні продукти та послуги, а оскільки продуктом є система з інноваційним рішенням, конкурувати буде значно легше, а також беручи до уваги те, що система є простою для впровадження, підтримує масштабованість, вона уде задовольняти потреби більшості компаній.

Таблиця 5.9 – Аналіз конкуренції в галузі за М. Портером

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
Складові аналізу	Прямі конкуренти відсутні на ринку України відсутні, на світовому – SA, Adobe. Непрямі конкуренти: CRM системи з аналітикою клієнтів.	Вихід світової лідерів на ринок України	Постачальники відсутні	Обмеження платоспроможністю	Товарозамінники відсутні
Висновки	Конкуренція несформована	Adobe, SA	Постачальники відсутні	Можлива відмова від рішення	Товарозамінники відсутні.

Далі необхідно проаналізувати та визначити фактори конкурентоспроможності компанії, що наведені в таблиці 5.10.

Таблиця 5.10 – Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Універсальність	Система може бути впроваджена в будь-яку систему аналітики та управління ресурсами компанії
2	Ціна	По відношенню до конкурентів, дана система має меншу собівартість, а це означає, що зменшується вартість використання системи
3	Інновації та технології	Система має інновації в таких місцях, як аналіз уподобань, визначення прихованих особливостей та формування ключових груп клієнтів.
4	Масштабованість	Для забезпечення конкурентоспроможності система має бути масштабована і готова до змін та розміру бізнесу клієнта.
5	Швидкість впровадження	Для бізнесу завжди важлива швидкість, щоб бути попереду конкурентів

За визначеними факторами конкурентоспроможності було проведено аналіз сильних та слабких сторін стартапу.

Таблиця 5.11 – Порівняльний аналіз сильних та слабких сторін «Інтелектуальна система підбору клієнтського контенту»

№	Фактор конкурентоспроможності	Рейтинг товарів-конкурентів у порівнянні						
		-3	-2	-1	0	1	2	3
1	Універсальність		✓					
2	Ціна			✓				
3	Інновації та технології				✓			
4	Масштабованість				✓			
5	Швидкість			✓				

Було розроблено альтернативи ринкової поведінки для виведення стартапу на ринок. На основі SWOT-аналізу розробляються альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок. SWOT аналіз складається з аналізу сильних, слабких сторін, загроз та можливостей. SWOT наведений у таблиці 5.12.

Таблиця 5.12 – SWOT-аналіз стартап-проекту

Сильні сторони: 1. ціна; 2. технологія, інновація	Слабкі сторони: 1. ІТ фахівців. 2. досвід
Можливості: 1. несформована конкуренція; 2. попит на системи аналізу клієнтів	Загрози: 1. поява нових конкурентів; 2. стан економіки середовища

Альтернативи ринкового впровадження наведені в таблиці 5.13.

З означених альтернатив обирається та, для якої:

- а) отримання ресурсів є більш простим та ймовірним;
- б) строки реалізації – більш стислими.

Таблиця 5.13 – Альтернативи ринкового впровадження стартап-проекту

№	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Спільні продажі	Дуже висока. У випадку партнера-гіганта	1 рік
	Індивідуальні продажі	Середня	1,5-2 роки

5.4 Розроблення ринкової стратегії проекту

Для отримання продаж необхідно визначити цільову аудиторію та групу потенційних споживачів. Опис таких груп наведеного в таблиці 5.14. Аналіз груп дає можливість сформулювати стратегію охоплення ринку для виходу на нього.

Таблиця 5.14 – Вибір цільових груп потенційних споживачів

№	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Онлайн та офлайн роздрібна торгівля	Готові	Зростаючий попит	Помірна	Вхід простий
Які цільові групи обрано: Усі.					

Для виходу на ринок необхідно визначити базову стратегію розвитку, яка наведена в таблиці 5.15.

Таблиця 5.15 – Визначення базової стратегії розвитку

№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1.	Спільні продажі	Домовленість між декількома компаніями	Інновація та технічне рішення	Впровадження

За вимогами споживачів у таблиці 5.16 розробляється стратегія позиціонування, що наведено в таблиці 5.17, для ідентифікації торгівельної марки/проекту.

Таблиця 5.16 – Визначення базової стратегії конкурентної поведінки

№	Чи є проект «першо-прохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1.	Так	Шукати нових	Ні	Стратегія спрямованості

Таблиця 5.17 – Визначення стратегії позиціонування

Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1. простота розгортання; 2. простота впровадження; 3. масштабованість; 4. інтеграція; 5. захищеність; 6. надійність; 7. якість 8. 9. вартість	Стратегія спрямованості	1. Цінова перевага. 2. інноваційне рішення; 3. простота розгортання; 4. простота впровадження; 5. масштабованість; 6. інтеграція.	1. підвищення прибутку; 2. оптимізація витрат; 3. збільшення клієнтів.

5.5 Розроблення маркетингової програми

Маркетингова програма – це комплекс завдань, що пов’язані для планомірного здійснення, і адресних заходів соціального, економічного, науково-технічного, виробничого, організаційного характеру з визначенням ресурсів, що використовуються. Першим кроком є формування маркетингової концепції товару, який отримає споживач. Для цього у 5.18 потрібно підсумувати результати попереднього аналізу конкурентоспроможності товару.

Таблиця 5.18 – Визначення ключових переваг концепції потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Збільшення прибутку	Спрямування ресурсів лише на ті групи клієнтів, які більше приносять прибутку	Вартість, технологія, відсутність зайвих функцій
2	Оптимізація витрат	Зменшення витрат на пошук прихованих груп клієнтів	
3	Прогнозування уподобань	Використання системи в якості рекомендаційної	

Наступним кроком є розробка тривірневої маркетингової моделі товару, для чого уточнюється ідея послуг та продуктів, що наведено таблиця 5.19.

Таблиця 5.19 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
I. Товар за задумом	Товар забезпечує клієнтів механізмом прогнозування як і ознак клієнтів, так і уподобань, формування прихованих груп, що приносить більше прибутку та оптимізує витрати компанії

Продовження таблиці 5.19

Рівні товару	Сутність та складові
II. Товар у реальному використанні	Товар представляє собою програмне забезпечення, що складається з одного артефакту
	Програмний артефакт надається у вигляді файлу, що одразу запускається
III. Товар із підкріпленням	До продажу: відбувається первинний запуск, налаштування конфігурування системи, та тренування НМ для клієнта
	Після продажу: підтримка ПЗ та додавання функції за потреби клієнта
Артефакт розповсюджується за ліцензією	

Наступним кроком є визначення цінових меж на потенційний товар, яке передбачає аналіз ціни на товари-аналоги, а також аналіз рівня доходів цільової групи споживачів, що наведене в таблиці 5.20.

Таблиця 5.20 – Визначення меж встановлення ціни

№	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1.	Від 10000\$	Від 25000\$	> 1 млн \$/рік	Нижня: 10000\$ Верхня: 30000\$

Наступним кроком є визначення оптимальної системи збуту, в межах якого приймається рішення, таблиця 5.21:

- проводити збут власними силами або залучати сторонніх посередників (власна або залучена система збуту);
- вибір та обґрунтування оптимальної глибини каналу збуту;
- вибір та обґрунтування виду посередників.

Таблиця 5.21 – Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Прагнення нижчої ціни	Прямі продажі, впровадження	Канал нульового рівня	Прямі продажі

Останньою складовою маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів, що наведено в таблиці 5.22.

Таблиця 5.22 – Концепція маркетингових комунікацій

№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1.	Поведінка B2B	Реклама, SEO Прямі продажі Конференції та тематичні заходи	1. підвищення прибутку; 2. оптимізація витрат; 3. збільшення клієнтів.	Розширення аудиторії	Вказати на використання інновацій та сучасних технологій та кейси вирішення проблем

Цільова аудиторія стартапу – B2B. Ефективні в цьому випадку є знайомства з директорами підприємств, заходи на конференціях та обзвони директором майбутніх клієнтів.

5.6 Висновки

В цьому розділі було проведено маркетинговий аналіз з метою визначення можливості та доцільності виходу на ринок з проектом інтелектуальної системи компаративної оцінки клієнтських уподобань.

В результаті дослідження було встановлено, що існує велика ймовірність успішної комерціалізації системи на ринку. Виходячи з того, що на ринку України прямих конкурентів немає, можна стверджувати, про легкий старт. Однак необхідно брати до уваги економічний стан в країні та можливість входу зарубіжних систем на ринок України. В результаті проведення маркетингового дослідження було обрано стратегії для того, щоб стартап був конкурентоспроможний, а також проаналізовано сильні та слабкі сторони, визначено цільову аудиторію та потенційних клієнтів.

Основною перевагою описуваної системи є інновації, використані при розробці, а беручи до уваги те, що, в першу чергу, проект націлений на B2B сегмент, найкращим каналом збуту є прямі продажі. Сильними сторонами продукту є простота впровадження та інтеграції, масштабованість. Виходячи з цього можна стверджувати про доцільність подальшої реалізації продукту.

ВИСНОВКИ

В даній роботі було описано розробку інтелектуальної системи компаративної оцінки клієнтських уподобань. Для досягнення поставленої задачі було проаналізовано предметну область, існуючі рішення та підходи у сфері аналізу клієнтів, розглянуто їх переваги та недоліки. Було встановлено, що у даній сфері існуючих рішень для використання в системах аналітики, що формують групи клієнтів за їх прихованими особливостями, немає, оскільки розділ цей аналітики тільки починає набирати свою значущість, а також через те, що більшість компаній впроваджують свої рішення. Під час розгляду більш великої сфери аналізу споживачів було визначено основні рішення та наведено їх особливості, в результаті чого було сформовано вимоги до системи.

В результаті роботи було:

- досліджено алгоритми формування груп клієнтів за ознаками, уподобаннями та прихованими особливостями;
- обґрунтовано вибір нейронних мереж та розроблено математичну модель клієнта з категоріальними ознаками;
- обґрунтовано вибір алгоритмів для формування груп клієнтів;
- розроблено та реалізовано систему компаративної оцінки клієнтських уподобань, базуючись на проведених дослідженнях.

Розроблена система може бути впроваджено в інші системи аналізу клієнтів, а завдяки обраній архітектурі, система є масштабованою та легко впроваджується. Також було проведено оцінку якості роботи системи шляхом аналізу наданих даних компанією-замовником та опитуванням її клієнтів, з чого було встановлено, що система прогнозує з великою точністю, а недостача пояснюється неточними даними про клієнтів.

Результати розробок та досліджень роботи були використані при розробці системи, що впроваджена в експлуатацію, що підтверджує практичну цінність одержаних результатів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. What Is Customer Analysis? [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://cmgpartners.com/blog/what-is-customer-analysis/>
2. Analytical CRM: Definition & Applications [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://study.com/academy/lesson/analytical-crm-definition-applications-quiz.html/>
3. Алгоритмы кластеризации [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://basegroup.ru/community/articles/datamining>
4. How to Conduct Customer Analysis and Customer Segmentation [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://www.brandwatch.com/blog/how-to-write-customer-analysis/>
5. The K-means Clustering Algorithm [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: http://kom.aau.dk/group/04gr742/pdf/kmeans_worksheet.pdf
6. Z HUANG. clustering large data sets with mixed numeric and categorical values. [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://pdfs.semanticscholar.org/d42b/b5ad2d03be6d8fef63d25d02c0711d19728.pdf>
7. Customer analytics [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://searchbusinessanalytics.techtarget.com/definition/customer-analytics/>
8. Adobe Analytics [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://www.adobe.com/analytics/adobe-analytics.html/>
9. IBM redefines customer analysis [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://www-935.ibm.com/services/uk/igs/pdf/case-ibm-redefines-customer-analysis.pdf>
10. SAP Marketing Cloud [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://cx.sap.com/en/products/marketing/>
11. CustomerAnalytics [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://customeranalytics.com/>

12. Geoffrey Hinton. A Practical Guide to Training Restricted Boltzmann Machines [Электронный ресурс]. – 2010. – Режим доступа до ресурсу: <https://www.cs.toronto.edu/~hinton/absps/guideTR.pdf>
13. Theory behind Restricted Boltzmann Machines—A powerful Tool for Recommender Systems [Электронный ресурс]. – 2018. – Режим доступа до ресурсу: <https://towardsdatascience.com/deep-learning-meets-physics-restricted-boltzmann-machines-part-i-6df5c4918c15>
14. Hugo Larochelle. Learning Algorithms for the Classification Restricted Boltzmann Machine [Электронный ресурс]. – 2012. – Режим доступа до ресурсу: <http://www.jmlr.org/papers/volume13/larochelle12a/larochelle12a.pdf>
15. Shusen Zhou, Qingcai Chen, Xiaolong Wang. Active Semi-Supervised Learning Method with Hybrid Deep Belief Networks [Электронный ресурс]. – 2014. – Режим доступа до ресурсу: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4160211/pdf/pone.0107122.pdf>
16. Go (programming language) [Электронный ресурс]. – 2018. – Режим доступа до ресурсу: [https://en.wikipedia.org/wiki/Go_\(programming_language\)](https://en.wikipedia.org/wiki/Go_(programming_language))
17. Go versus Python 3 fastest programs [Электронный ресурс]. – 2018. – Режим доступа до ресурсу: <https://benchmarksgame-team.pages.debian.net/benchmarksgame/faster/go-python3.html/>
18. Mark Richards. Software architecture patterns [Электронный ресурс]. – 2015. – Режим доступа до ресурсу: <http://guillemette.org/uqam/mgl7361-h2018/assets/articles/software-architecture-patterns.pdf>

ДОДАТОК А

Приклади реальних даних клієнтів

userID	smoker	drink_level	ambience	transport	marital_st	interest	personality	religion	activity	budget
U1001	FALSE	abstemious	family	on foot	single	variety	thrifty-protector	none	student	medium
U1002	FALSE	abstemious	family	public	single	technology	hunter-ostentatic	Catholic	student	low
U1003	FALSE	social drinker	family	public	single	none	hard-worker	Catholic	student	low
U1004	FALSE	abstemious	family	public	single	variety	hard-worker	none	professor	medium
U1005	FALSE	abstemious	family	public	single	none	thrifty-protector	Catholic	student	medium
U1006	TRUE	social drinker	friends	car owner	single	variety	hard-worker	none	student	medium
U1007	FALSE	casual drinker	solitary	public	single	variety	thrifty-protector	Catholic	student	low
U1008	FALSE	social drinker	solitary	public	single	technology	hard-worker	Catholic	student	low
U1009	FALSE	abstemious	family	on foot	single	variety	thrifty-protector	Catholic	student	medium
U1010	FALSE	social drinker	friends	car owner	married	technology	hard-worker	none	student	medium
U1011	FALSE	abstemious	family	public	single	variety	hard-worker	Catholic	student	medium
U1012	FALSE	casual drinker	family	public	single	technology	hard-worker	Catholic	student	medium
U1013	FALSE	abstemious	friends	public	widow	retro	thrifty-protector	none	professor	medium
U1014	FALSE	abstemious	friends	public	single	none	hard-worker	Catholic	student	medium
U1015	TRUE	social drinker	family	public	single	technology	thrifty-protector	Catholic	student	medium
U1016	FALSE	casual drinker	friends	on foot	single	eco-friendly	thrifty-protector	Catholic	student	medium
U1017	FALSE	casual drinker	family	public	single	retro	hunter-ostentatic	none	professor	medium
U1018	TRUE	casual drinker	friends	public	single	technology	thrifty-protector	Christian	student	low

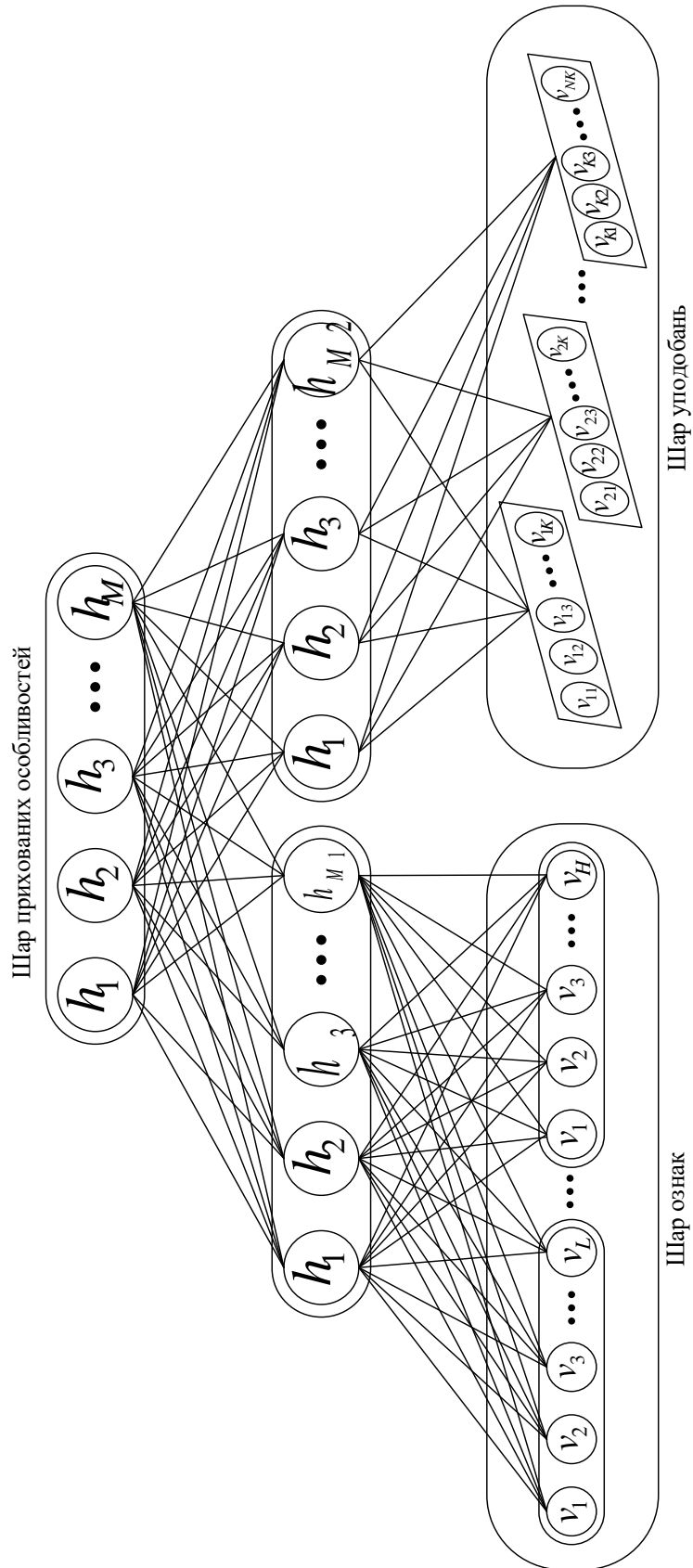
Рисунок А.1 – Приклад ознак клієнтів

userID	1	2	3	4	5	6	7	8	9	10
U1001	10	7	5	4	4	5	7	0	0	6
U1002	1	1	0	0	0	0	1	1	2	0
U1003	3	3	2	6	5	4	1	4	4	2
U1004	5	6	4	3	0	8	7	5	4	3
U1005	1	7	0	0	0	7	5	6	10	1
U1006	7	5	5	3	9	4	0	9	5	2
U1007	6	4	3	7	7	5	4	0	0	0
U1008	10	5	4	4	9	7	3	3	1	0
U1009	9	5	7	5	4	6	6	6	8	3
U1010	7	5	5	6	8	3	7	7	8	0
U1011	8	7	7	7	9	10	6	5	1	2
U1012	3	4	5	3	1	8	0	6	5	6
U1013	6	5	4	8	6	7	4	7	9	7
U1014	1	6	8	0	5	4	7	8	9	0
U1015	7	6	5	1	9	0	1	2	8	7
U1016	6	7	9	0	1	3	6	8	10	10
U1017	7	5	4	5	8	5	4	0	0	0
U1018	10	7	5	2	9	6	0	6	8	5

Рисунок А.2 – Приклад уподобань клієнтів

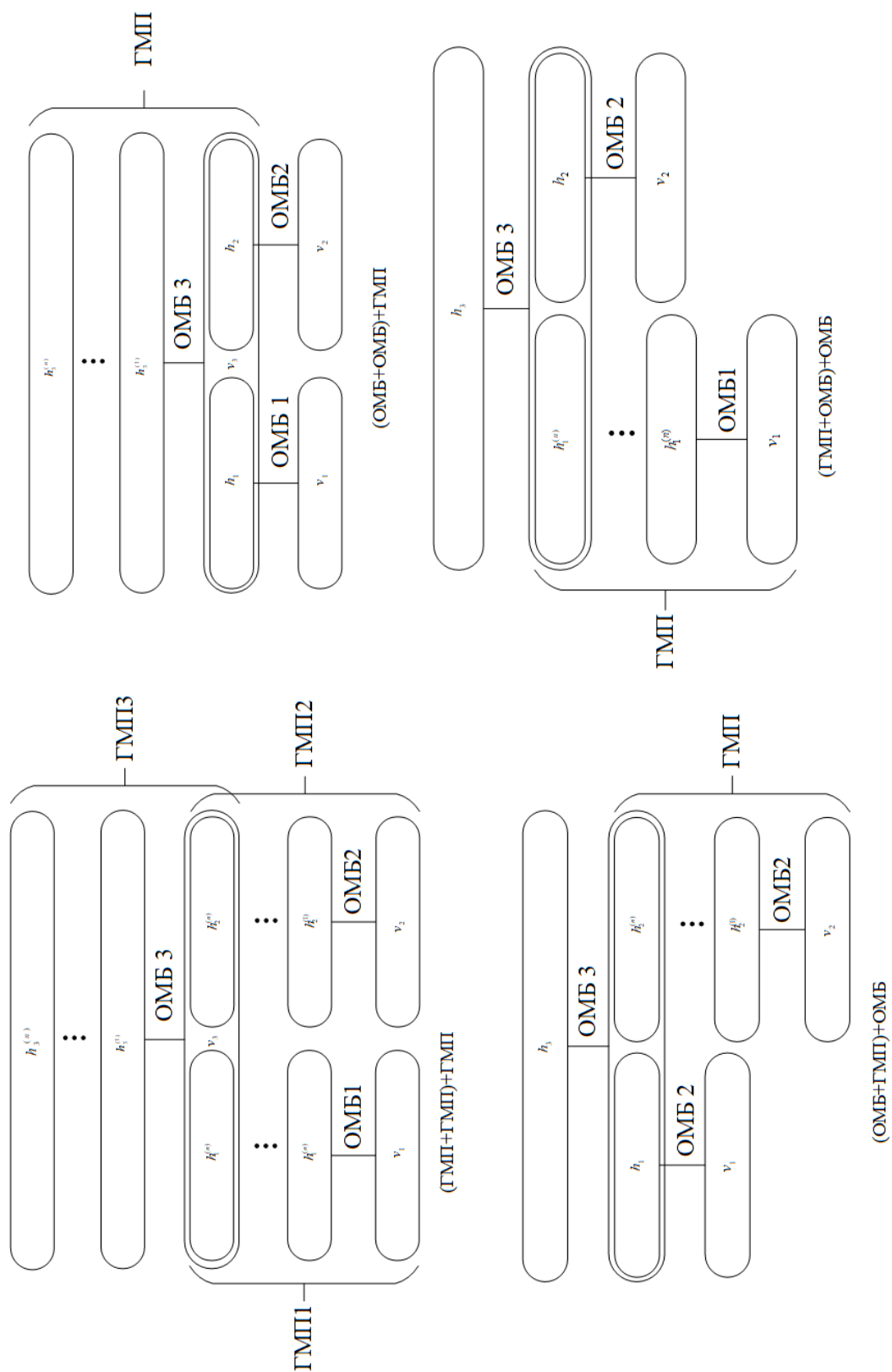
ДОДАТОК Б

Структурна схема моделі модуля для аналізу клієнтських уподобань



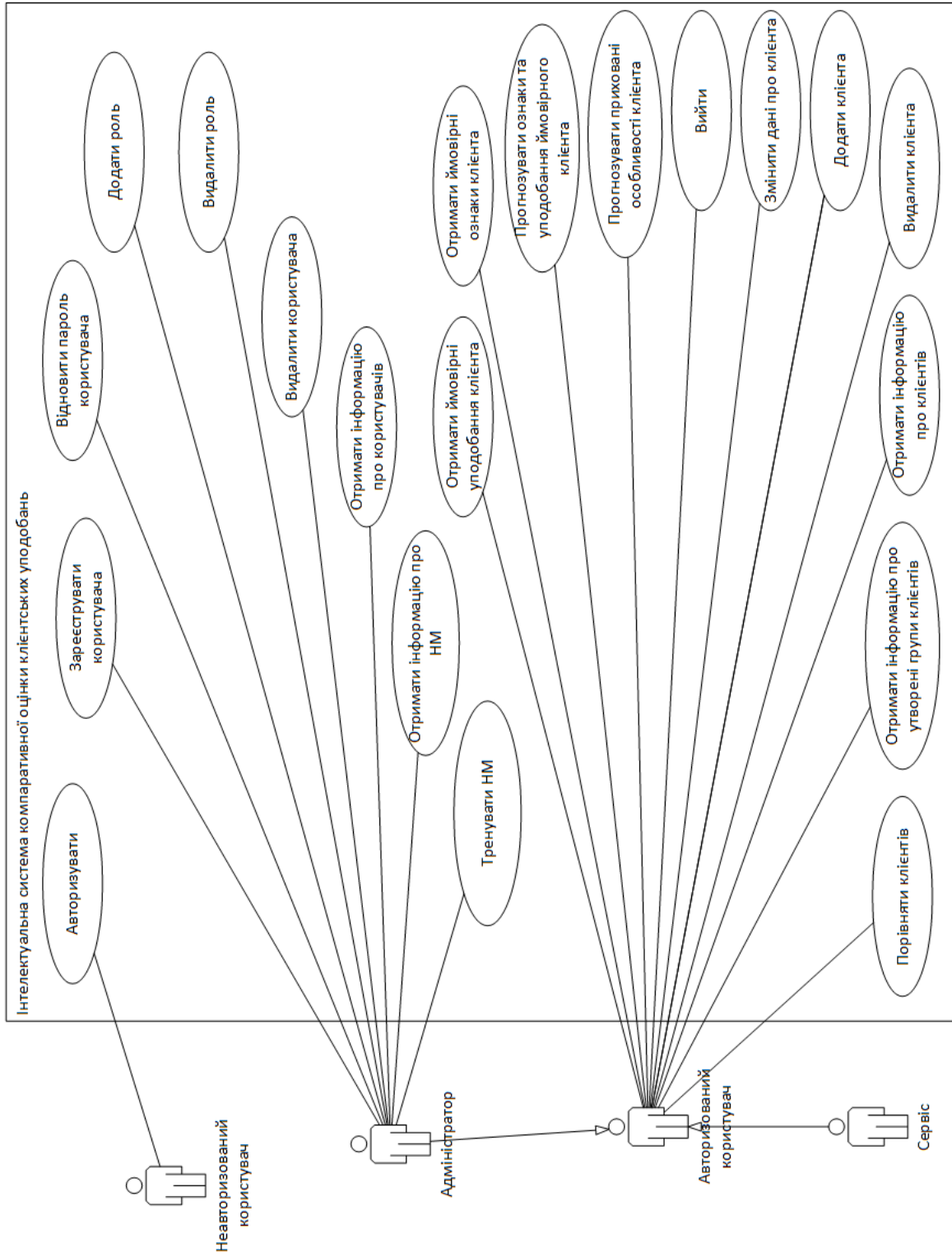
ДОДАТОК В

Варіації моделі модуля аналізу клієнтських уподобань



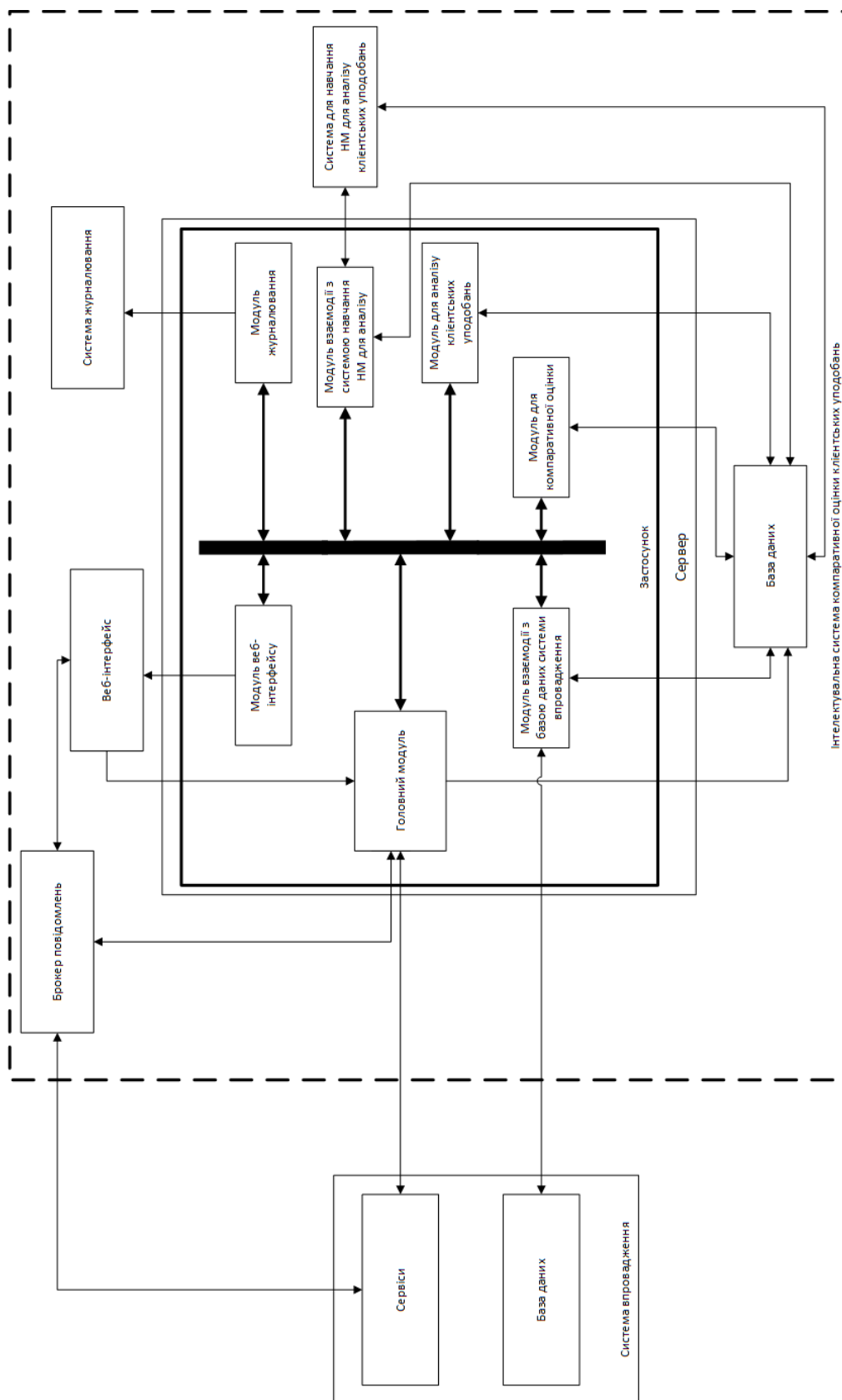
ДОДАТОК Г

Діаграма варіантів використання інтелектуальної системи



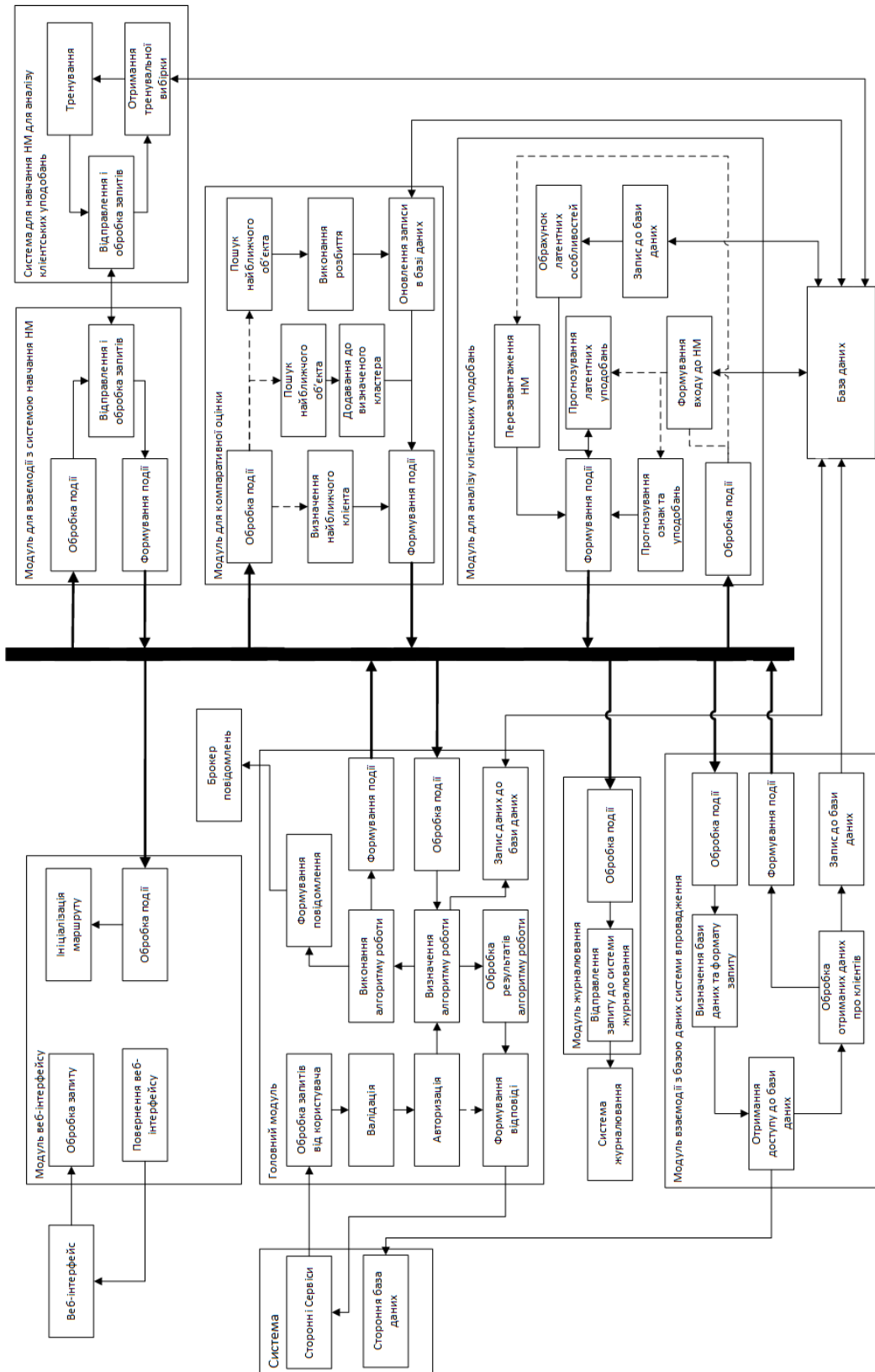
ДОДАТОК Д

Структурна схема інтелектуальної системи



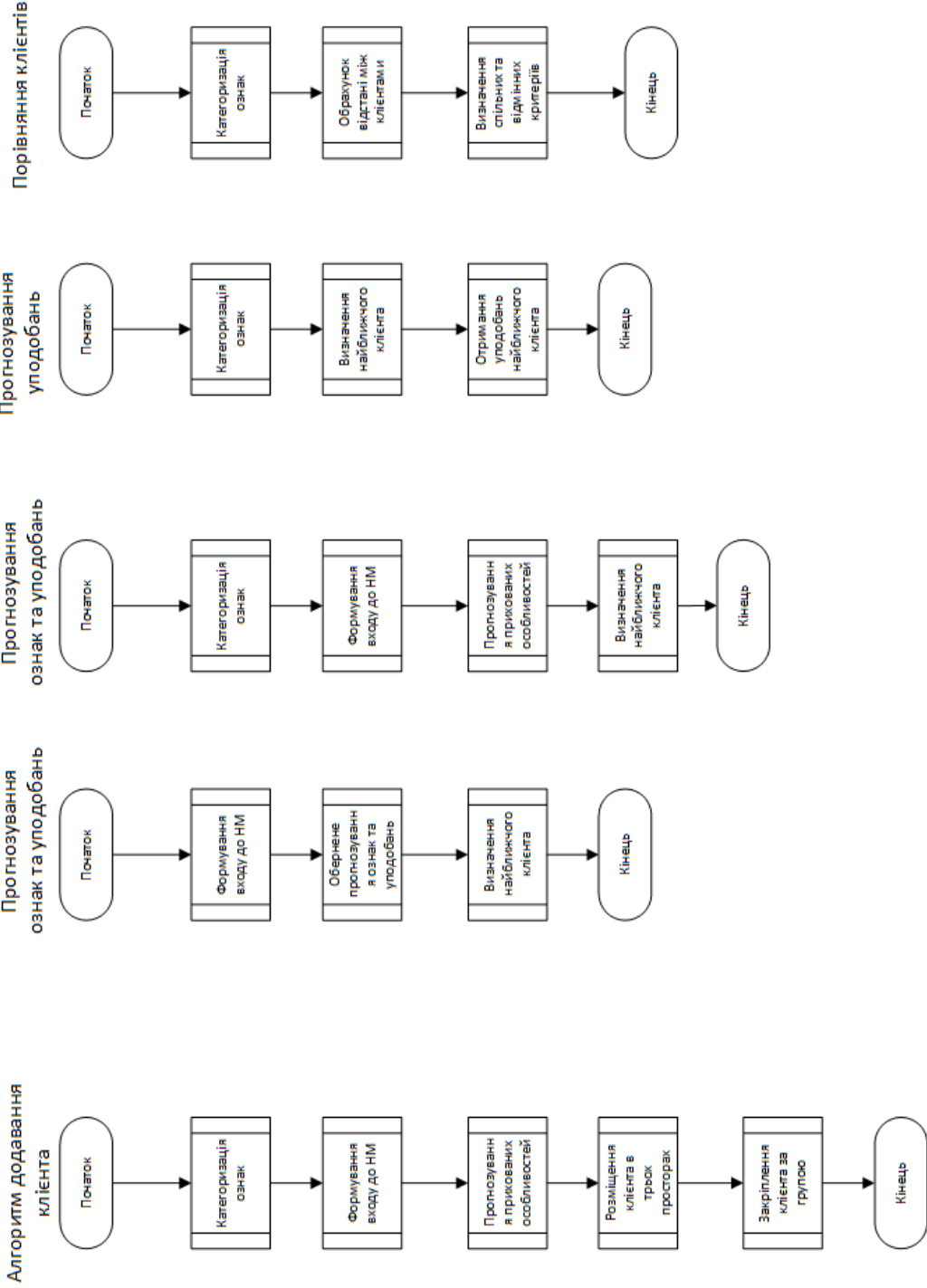
ДОДАТОК Е

Функціональна схема інтелектуальної системи



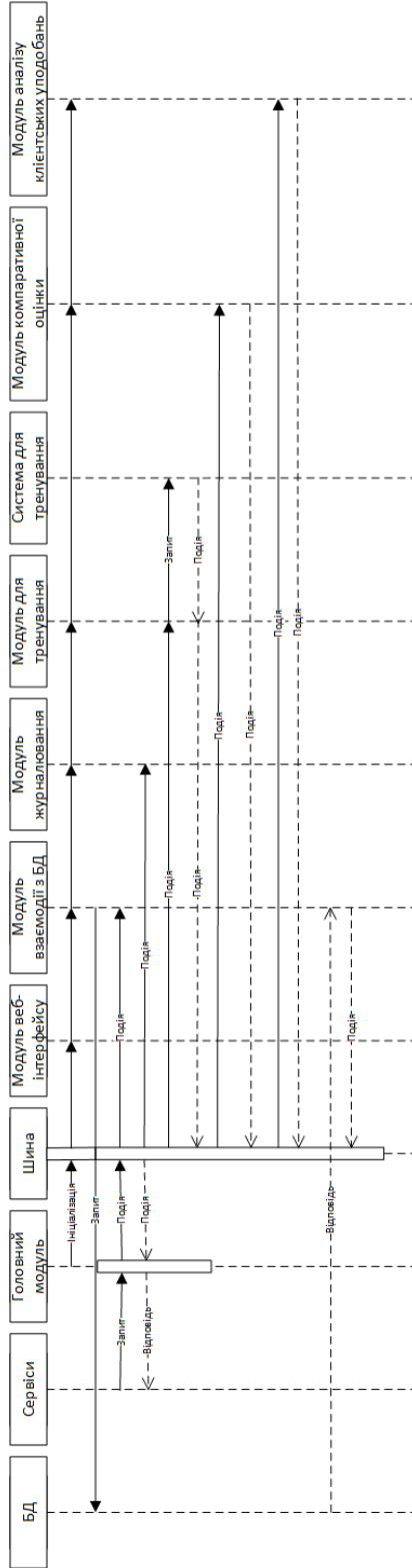
ДОДАТОК Ж

Схема алгоритмів роботи



ДОДАТОК К

Діаграма послідовності



ДОДАТОК Л

Діаграма розгортання

