

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____
(повна назва)

Кафедра _____ Інформаційних управляючих систем _____
(повна назва)

АТЕСТАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти _____ другий (магістерський) _____

_____ Дослідження методів формування пояснень у проектах _____
рекомендаційних систем _____
(тема)

Виконав:

студент 2 курсу, групи УПГІТМ-19-1

Різниченко О.А.

(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки

(код і повна назва спеціальності)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Управління проектами у
галузі інформаційних технологій

(повна назва освітньої програми)

Керівник проф. Чалий С. Ф.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

_____ (підпис)

Петров К.Е.

(прізвище, ініціали)

2020р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)Кафедра Інформаційних управляючих систем
(повна назва)Рівень вищої освіти другий (магістерський)Спеціальність 122 Комп'ютерні науки
(код і повна назва)Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)Освітня програма Управління проектами у галузі інформаційних технологій
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« _____ » _____ 20 ____ р.

ЗАВДАННЯ
НА АТЕСТАЦІЙНУ РОБОТУстудентові Різниченко Олексію Андрійовичу
(прізвище, ім'я, по батькові)1. Тема роботи Дослідження методів формування пояснень у проектах рекомендаційних системзатверджена наказом по університету від 27.10 2020 р. № 1454 Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 20__ р.

3. Вихідні дані до роботи Науково-технічні публікації та інтернет джерела з тематики атестаційної роботи4. Перелік питань, що потрібно опрацювати в роботі вступ, аналіз проектів рекомендаційних систем, аналіз структури рекомендаційних систем, аналіз властивостей пояснень в рекомендаційних системах, дослідження методів побудови пояснень в інтелектуальних системах, постановка задачі атестаційної роботи, дослідження методів побудови пояснень в рекомендаційних системах на правилах, опис динаміки вподобань користувачів з використанням темпоральних правил, удосконалений метод побудови пояснень з урахуванням змін в уподобаннях користувачів, опис проекту розробки методу, статут проекту, планування проекту, розробка програмного модулю формування пояснень в рекомендаційній системі, експериментальна перевірка удосконаленого методу формування пояснень та опис отриманих результатів, висновки.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз літератури та Інтернет-джерел	02.11.2020	
2	Постановка задачі	03.11.2020-05.11.2020	
3	Обробка матеріалу	06.11.2020-08.11.2020	
4	Дослідження сучасної структури роботи рекомендаційних систем	08.11.2020-10.11.2020	
5	Дослідження методів формування пояснень у проектах рекомендаційних систем	10.11.2020-19.11.2020	
6	Дослідження особливостей методів реалізації формування пояснень у проектах рекомендаційних систем	19.11.2020-21.11.2020	
7	Апробація результатів дослідження на прикладі	21.11.2020-23.11.2020	
8	Написання пояснювальної записки	29.11.2020-07.12.2020	
9	Підготовка презентації	08.12.2020-09.12.2020	
10	Перевірка на плагіат	10.12.2020-11.12.2020	
11	Нормоконтроль	11.12.2020-15.12.2020	
12	Захист	18.12.2020	

Дата видачі завдання 02.11 2020 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис) проф. Чалий С. Ф
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка до магістерської атестаційної роботи містить: 92 с., 4 розділи, 22 рис., 2 табл., 41 джерело.

ЕЛЕКТРОННА КОМЕРЦІЯ, ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ, ІНТЕРНЕТ-МАГАЗИН, МЕТОД УРАХУВАННЯ ТЕМПОРАЛЬНОЇ ДИНАМІКИ, РЕКОМЕНДАЦІЙНІ СИСТЕМИ, ФОРМУВАННЯ РЕКОМЕНДАЦІЙ, ШТУЧНИЙ ІНТЕЛЕКТ

У роботі виконано огляд методів формування пояснень у проектах рекомендаційних систем. Проаналізовано існуючі методи формування пояснень. На підставі проведеного аналізу запропоновано покращений метод формування пояснень у проектах рекомендаційних систем з урахуванням темпоральної динаміки зміни звичок користувача.

В ході дослідження отримані такі результати: визначені існуючі методи надання рекомендацій; визначені існуючі властивості пояснень у рекомендаційних системах; визначені існуючі методи формування пояснень у інтелектуальних системах; визначені існуючі методи формування пояснень у рекомендаційних системах; виконана класифікація методів формування пояснень та опис удосконаленого методу; проведено експериментальну перевірку удосконаленого методу.

ABSTRACT

Explanatory Note to master certification work contains 92 pages, 4 sections, 22 pictures, 2 tables, 41 sources.

ARTIFICIAL INTELLIGENCE, ELECTRONIC COMMERCE, INTELLECTUAL SYSTEMS, FORMATION OF RECOMMENDATION, METHOD OF RECORDING TEMPORAL DYNAMICS, RECOMMENDATION SYSTEMS, WEB STORE

The paper reviews the methods of forming explanations in the projects of recommendation systems. The existing methods of formation of explanations are analyzed. On the basis of the conducted analysis the improved method of formation of explanations in projects of recommendation systems taking into account temporal dynamics of change of habits of the user is offered.

The research work obtained the following results: the existing properties of explanations in recommendation systems are determined; the existing methods of formation of explanations in intelligent systems are defined; the existing methods of formation of explanations in recommendation systems are defined; the classification of methods of formation of explanations and the description of the improved method is executed; an experimental test of the improved method was performed.

**ПЕРЕЛІК СКОРОЧЕНЬ, УСЛОВНИХ ПОЗНАЧЕНЬ,
СИМВОЛІВ, ОДИНИЦЬ ТА ТЕРМІНІВ**

КФ – Коллаборативна фільтрація;

РС – Рекомендаційна система;

ШІ – Штучний інтелект;

СВР – Case Based Reasoning;

СМS – Content Management System;

СРМ – Critical Path Method;

СРМ – Customer Relationship Management;

JRE – Java Runtime Environment.

JVM – Java Virtual Machine.

LID – Lazy Induction Directions;

КBS – Knowledge Based System;

UI – User Interface;

UX – User Experience;

ЗМІСТ

Вступ.....	8
1 Аналіз проектів рекомендаційних систем.....	10
1.1 Аналіз структури рекомендаційних систем.....	10
1.2 Аналіз властивостей пояснень в рекомендаційних системах.....	13
1.3 Дослідження методів побудови пояснень в інтелектуальних системах.....	23
.	
1.4 Постановка задачі дослідження	34
2 Види пояснень та представлень у рекомендаційних системах.....	35
2.1 Методи побудови пояснень на правилах в рекомендаційних системах.	35
2.2 Опис динаміки вподобань користувачів з використанням темпоральних правил.....	38
2.3 Метод побудови пояснень з урахуванням змін в уподобаннях користувачів.....	42
3 Проект розробки методу надання представлень у проектах рекомендаційних систем.....	48
3.1 Опис проекту розробки методу.....	48
3.2 Статут проекту.....	49
3.3 Планування проекту.....	50
4 Практичне використання отриманих результатів.....	57
4.1 Розробка програмного модулю формування пояснень в рекомендаційній системі.....	57
4.2 Експериментальна перевірка удосконаленого методу формування пояснень.....	63
Висновки.....	67
Перелік джерел посилання.....	68
Додаток А	73

ВСТУП

Розвиток інформаційних технологій забезпечує нові можливості з рекомендації товарів та послуг для кінцевих споживачів.

Персоналізація в системах електронної комерції базується на регіоні проживання, перегляді товарів і історії покупок. Ці дані допомагають власникам компаній формувати пропозиції, які здатні максимально зацікавити кожного конкретного клієнта.

Рекомендаційні підсистеми використовуються для організації процесів персоналізації продажів. Вони пропонують товари споживачам на основі їх профілів користувачів, таким чином забезпечуючи індивідуальні рекомендації. Рекомендаційні системи виявилися корисними в таких сферах, як електронна комерція, веб-пошукові системи, формування рекомендацій для програм цифрового телебачення та ін.

Важливу роль у рекомендаційних системах відіграють пояснення. Пояснення можуть широко впливати на вибір користувача та його рівень задоволення. Провідні компанії та корпорації по продажу товарів онлайн останнім часом збільшують відсоток інвестицій у дослідження та розроблення нових методів та підходів надання пояснень користувачам.

Одним з базових і найпоширеніших методів є рекомендації на основі кейсів. Система буде рекомендувати елементи, які отримали високі рейтинги іншими користувачами з подібними смаками та інтересами.

У системах, що використовують метод кейсів, профіль користувача визначається на основі сукупності оцінок, присвоєних різним елементам. Ці рейтинги можуть бути зафіксовані явним шляхом, тобто, за результатами відповідей користувача або неявно, внаслідок аналізу його взаємодії з рекомендаційною системою

Рейтинг може бути представлений унарним значенням (відображенням лише відповідних елементів), бінарним (що дозволяє розрізнити хороші та

погані елементи) або, більш загально, як чисельне значення на відповідній кінцевій шкалі.

Сучасні методи надання рекомендацій не вирішують ряд взаємопов'язаних проблем, які впливають на якість рекомендацій. До них, зокрема, відносяться проблеми холодного старту, спам-атак неузгодженості інтересів споживачів, та масштабованості.

Проблема холодного старту стосується складності у наданні рекомендацій користувачам, які нещодавно були введені в систему. В таких випадках користувачі ще не оцінили достатньо товарів, тому рекомендаційна система не може прогнозувати їх інтереси.

Для вирішення цієї проблеми використовуються початкові рейтинги, тобто система змушує користувача спочатку оцінити певний набір товарів. Однак ці первинні рейтинги можуть ввести упередженість в систему. Проблема холодного запуску також впливає на нові товари, оскільки вони не будуть в списку рекомендацій, поки достатньо користувачів їх не оцінили.

Проблема спам-атаки полягає у введенні рекомендаційної системи в оману недобросовісними користувачами.

Для вирішення комплексу наведених проблем використовують пояснення, що підвищують довіру користувача до роботи системи.

Однак вказані методи не враховують зміну вподобань користувача з часом.

Для вирішення вказаної проблеми доцільно використовувати темпоральні залежності, що відображають зміну кількості проданих товарів або послуг з часом.

Отриманим результатом роботи є удосконалений метод побудови пояснень в рекомендаційних системах на основі врахування темпоральних залежностей, що описують зміну інтересів користувача для процесів покупки товарів на обраному інтервалі часу.

Отримані результати виконання магістерської атестаційної роботи оформлені у вигляді пояснювальної записки.

1 АНАЛІЗ ПРОЕКТІВ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

1.1 Аналіз структури рекомендаційних систем

В останні роки спостерігається підвищений інтерес до більш орієнтованих на користувача метрик оцінки для рекомендаційних систем. Також було визнано, що багато рекомендаційних систем, функціонували як чорні ящики, не забезпечуючи прозорості роботи процесу рекомендацій, а також не пропонували жодної додаткової інформації, яка супроводжує рекомендації, окрім самих рекомендацій.

Пояснення можуть забезпечити прозорість, викриваючи міркування та дані, що стоять за рекомендацією. Це стосується, наприклад, пояснень розміщених на Amazon, таких як: "Клієнти, які придбали цей товар, також купили ...". Пояснення можуть також слугувати іншим цілям, наприклад, допомогти викликати довіру та лояльність користувачів, підвищити рівень задоволеності, облегшити та прискорити пошук користувачів, виявити їх вподобання, і переконати їх спробувати або придбати рекомендований товар. Таким чином, ми розрізняємо різні пояснення, такі як наприклад пояснення, як працює механізм рекомендацій (прозорість), і пояснити, чому користувач може спробувати продукт (ефективність). Ефективне пояснення може бути сформульоване у формі "Можливо, вам (не) сподобається пункт А, тому що ...". На відміну від прикладу Amazon вище, це пояснення не обов'язково описує, як було обрано рекомендацію - у цьому випадку воно не є прозорим.

Можна сказати, що експертні системи є попередниками рекомендаційних систем. Тому в наступному розділі ми коротко розкриємо дослідження методів оцінки пояснень в експертних системах, інтелектуальних системах та оцінки пояснень у рекомендаційних системах. Ми також оцінюємо розвиток систем рекомендацій, які могли б викликати поживлення інтересу до наукових досліджень з часів існування експертних систем.

До цього часу було мало консенсусу щодо того, як оцінювати пояснення, або чому взагалі пояснювання потрібні. У роботі ми перераховуємо сім пояснювальних критеріїв та описуємо, як вони вимірювались у попередніх системах. Ці критерії також можна розуміти як переваги, які пояснення можуть запропонувати рекомендаційним системам, відповідаючи на питання, чому пояснення потрібні. У наведених вище прикладах ефективних та прозорих пояснень ми побачили, що два критерії оцінки можуть бути взаємовиключними.

Приклад процесу роботи пояснень зображено на рисунку 1.1. Ми побачимо, що пояснення можуть бути подані у різних формах, використовуючи як текст, так і графіку.

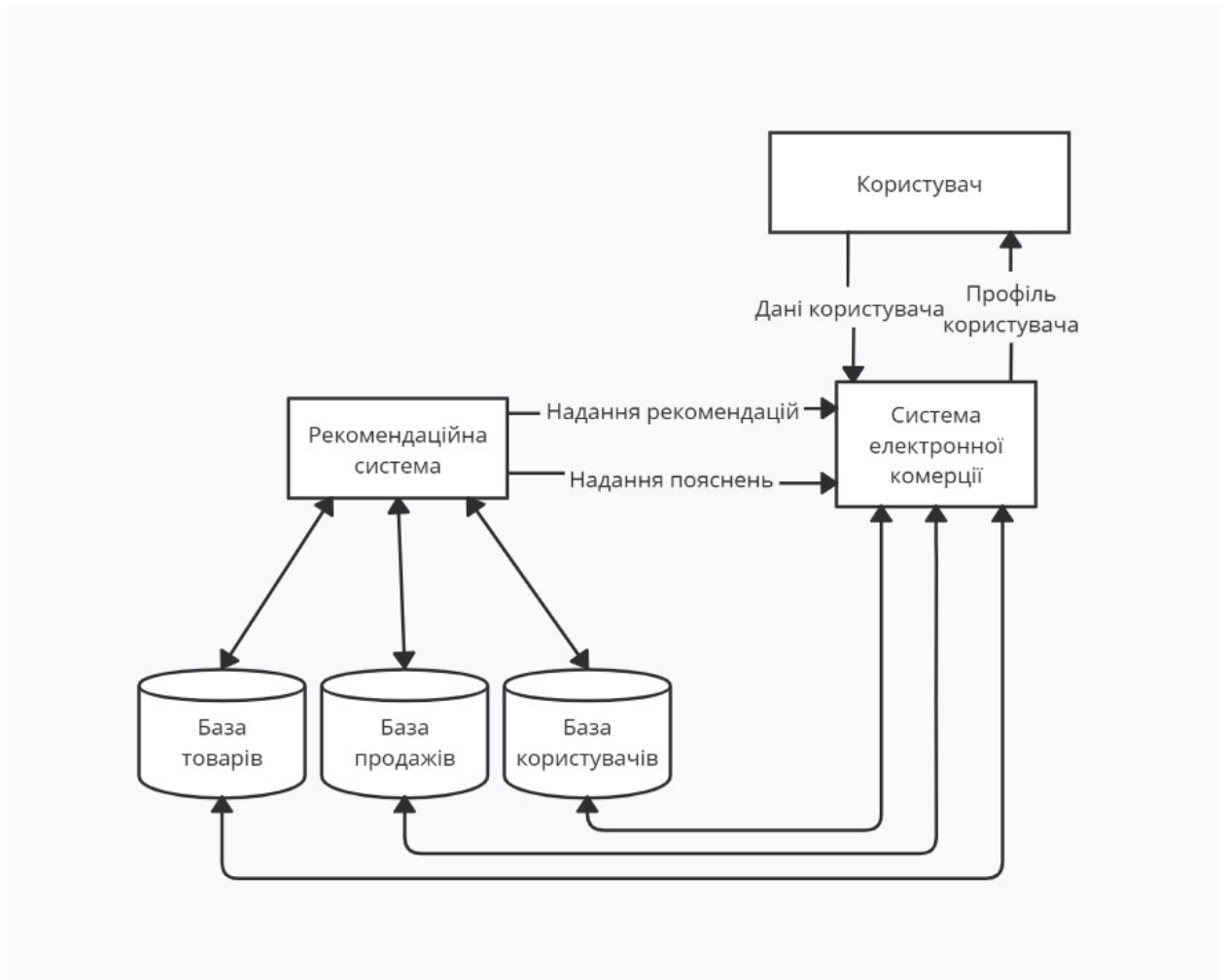


Рисунок 1.1 – Вигляд рекомендаційної системи тер

Я вважаю, що основа рекомендаційної системи впливає на оцінку пояснень, і обговорюю це з точки зору показників оцінки, які зазвичай використовуються для рекомендаційних систем (наприклад, точності та охоплення). У процесі аналізу ми розбираємо та ілюструємо приклади пояснень, та розглядаємо основну концепцію роботи пояснень у рекомендаційних системах.

Крім того, пояснення не відокремлюються від самих рекомендацій або способу взаємодії користувачів із системою рекомендацій: обидва фактори впливають один на одного та можливість їх сформувати, впливає на ступінь досягнення цілей пояснення.

Більше того, базовий алгоритм механізму рекомендацій може впливати на типи пояснень, які можуть бути сформовані, і одночасно є можливість, що пояснення, обрані розробником системи, не завжди відображають базовий алгоритм. Це особливо стосується обчислювально складних алгоритмів, для яких пояснення може бути складніше сформувати із-за параметрів, наприклад, спільна фільтрація. У цьому випадку розробник повинен враховувати компроміси між різними цілями пояснення, такими як задоволеність та прозорість.

Рекомендаційні системи призначені для побудови персоналізованого списку предметів, наприклад товарів або послуг, які відповідають інтересам поточного користувача. Такі системи формують рекомендації на основі подібності предметів або схожих критеріїв користувачів. В якості вхідних даних для побудови рекомендацій, використовуються інформація про продажі предметів, а також рейтинги цих предметів, які були внесені попередніми користувачами. Рекомендаційні системи полегшують вибір для користувача, що дозволяє збільшити обсяги продажів. Тому такі системи широко застосовуються при продажу товарів, перегляді фільмів, продажу туристичних турів, рекомендації конференцій та інших.

1.2 Аналіз властивостей пояснень в рекомендаційних системах

Не так давно відбулася активізація досліджень в області використання штучного інтелекту для пояснень, оскільки дослідники та практики прагнуть забезпечити більшу прозорість своїх алгоритмів для рекомендацій. Значна частина цього дослідження зосереджена на явному поясненні рішень та дій користувача, і твердження, що розгляд процесу, як люди надають пояснення один одному, може слугувати корисною відправною точкою для пояснень в штучному інтелекті. Однак справедливо сказати, що більшість робіт над поясненнями у штучному інтелекті, використовують лише інтуїцію дослідників щодо того, що становить "якісне" пояснення. Існує достатньо велика кількість цінних досліджень у галузі філософії, психології та когнітивної науки про те, як люди визначають, генерують, відбирають, оцінюють та представляють пояснення, та вони стверджують що люди застосовують певні когнітивні упередження та соціальні очікування в процесі пояснення.

В рамках цього розділу я відібрав чотири основні пункти з проаналізованої літератури, які, на мою думку, є важливими для пояснень к ШІ, та одночасно, на мою думку, більшістю дослідників та фахівців з питань штучного інтелекту в даний час не досліджені до кінця:

Пояснення мають такі характеристики:

- контрастивність;
- упередженість;
- ймовірнісний характер;
- соціальний характер;

Контрастивність пояснень полягає у тому, що їх шукають у відповідь на конкретні випадки, що мають набір факторів, які в цьому випадку називається контрастами. Тобто, користувачі не запитують, чому сталася подія P, а будують питання навпаки, чому сталася подія P замість іншої події

Q. Цей алгоритм побудови питання має важливі соціальні та обчислювальні наслідки для пояснень у ШІ.

Пояснення підбираються (упереджено) - користувачі рідко, якщо взагалі коли-небудь, очікують пояснення, яке складається з фактичної та повної причини події. Користувачі зазвичай обирають одну чи дві причини із певної вибірки причин, які будуть поясненням для них. Однак на цей вибір впливають певні когнітивні упередження.

Ймовірнісний характер пояснень має велике значення, посилення на ймовірності чи статистичні зв'язки в поясненні не демонструють приріст ефективності, як посилення на причини. Іноді пояснення не завжди є найкращими обґрунтуванням вибору для користувача. Головна ціль використання статистичних узагальнень для обґрунтування того, чому сталася подія, це супроводження основним поясненням причини всього узагальнення.

Соціальний характер займає важливу роль у поясненнях та представляє собою передачу знань, представлених частиною бесіди або взаємодії з системою, і таким чином впливає на переконання користувача.

Усі ці чотири параметри сходяться навколо однієї точки: пояснення - це не просто подання асоціацій та причин (причинно-наслідкових процесів), вони є контекстуальними. Хоча подія може мати певний набір причин, часто пояснення націлене лише на невелику підмножину причин, після цього система вибирає коефіцієнт цієї підмножини на основі кількох різних критеріїв, і далі працює взаємодія системи та користувача щодо правильності надання пояснення.

Користувачі використовують пояснення по наступним причинам:

- потреба в інформації;
- розуміння;
- причинно-наслідковий фільтр;
- потреба контрастності;

Потреба в інформації є одним із основних критеріїв, які використовують користувачі, але до інших прагматичних причин належить і огляд інформації - наприклад, вчитель просить своїх учнів про пояснення пройденого матеріалу на іспиті, з метою перевірки знань учнів з певної теми; та наукове пояснення - питання, чому ми спостерігаємо певне явище навколишнього середовища.

Зрозуміло, що головною функцією пояснення є сприяння навчанню. Навчаючись, ми отримуємо кращі моделі того, як проходять певні події та їх властивості, і ми можемо використовувати ці моделі з наукової точки зору. Хайдер стверджує, що люди шукають пояснень, щоб поліпшити своє розуміння процесів, щоб вони могли отримати стабільну модель, яку можна використовувати для прогнозування та контролю. Ця гіпотеза підкріплена дослідженнями, що свідчать про те, що користувачі, як правило, задають питання про події чи спостереження, які вони вважають ненормальними або несподіваними з власної точки зору.

Ломброзо стверджує, що пояснення відіграють роль у виведенні висновків саме тому, що вони є поясненнями саме завдяки причинно-наслідковій інформації, яку вони розкривають. По-перше, пояснення дають деякий "фільтр" про причинно-наслідкові процеси події. По-друге, попередні знання можуть змінюватись, надаючи пояснення; тобто, попросивши суб'єкт надати пояснення щодо того, чи є певна властивість істинною чи хибною, пояснювач змінює їх сприйняту ймовірність вимоги. По-третє, пояснення, які пропонують менший список причин, і пояснення, що описують численні спостереження, вважаються більш правдоподібними та ціннішими; але це не стосується причинно-наслідкових тверджень. Ломброзо йде далі і показує, що включення до пояснення, але без досягнення правильного результату пояснення може покращити розуміння людини. Він описує це як "пояснення для найкращого висновку", на відміну від типової моделі пояснення, яка стверджує що кращим буде "висновок для найкращого пояснення".

Малле, який надає найбільш повну оцінку повсякденних пояснень у контексті обґрунтування соціальних дій / взаємодій, стверджує, що користувачі потребують пояснень з двох причин:

- знайти значення: порівняти суперечності або невідповідності між елементами структур знань;

- управління соціальною взаємодією: створити спільний зміст і змінити переконання та враження інших, їхні емоції або вплинути на їхні дії;

Створення спільного значення важливо для пояснень в ШІ. У багатьох випадках пояснення, надане інтелектуальним агентом, полягатиме саме в тому, щоб створити спільне розуміння рішення, яке було прийнято між ним та користувачем, принаймні до певного часткового рівня.

Ломброзо зазначає, що пояснення мають кілька функцій, крім передачі знань, такі як переконання, навчання або виявлення неправди; і що в деяких випадках соціального пояснення цілі пояснювача та користувача можуть бути різними. Що стосується пояснень в ШІ, переконання, безперечно, займає перше місце: якщо метою пояснення інтелектуального агента є викликати довіру у користувача, то переконання, що рішення є правильним, у певному випадку може вважатися важливішим, ніж фактичне передання справжньої причини. Наприклад, існують ситуації де можливість дати менш вірогідне пояснення, яке є більш переконливим для користувача, якщо ми хочемо, щоб користувач діяв у позитивному для нас, шляху. У цьому випадку цілі пояснювача (створити довіру) відрізняються від цілей користувача (зрозуміти рішення).

Пояснення на основі контрастів є одним з найважливіших висновків у філософсько-когнітивній науковій літературі з точки зору пояснень у ШІ. Дослідження показують, що користувачі не обґрунтовують причини подій як вони є, а ґрунтують причину події відносно іншої події, яка не відбулася; тобто пояснення завжди має форму "Чому P, а не Q?", де P - цільова подія, а Q – контрастний випадок, який не відбувся, навіть якщо Q є неявним у питанні. Це називається контрастним поясненням.

Деякі автори називають Q випадком протидії. Однак важливо зазначити, що це не той самий фактор, на який посилаються при протидії. Що стосується причинно-наслідкового зв'язку, то контрасти є гіпотетичними "непричинами", при яких не треба пояснити подію, що підлягає поясненню, - це є протиправним фактором, що спричиняє C -, тоді як у протилежних поясненнях контрасти є гіпотетичними результатами - тобто протилежністю події E .

Ломброзо посилається на два випадки, P і Q , як факт і контраст відповідно; факт - це подія, яка відбулася, а контраст - подія, яка не відбулася.

Для пояснення стилю на основі змісту передбачуваними вхідними даними для механізму рекомендацій є рейтинги користувача u (для підмножини) елементів в I . Ці оцінки потім використовуються для створення класифікатора, який відповідає поведінці рейтингу u , та використання його на i . Прогноз для рекомендованого елемента ґрунтується на тому, наскільки він відповідає цьому класифікатору. Наприклад якщо він подібний до інших високо оцінених предметів.

Комерційна система Pandora має пояснення для своєї рекомендації щодо пісень відповідно до музичних властивостей, таких як темп та тональність. Ці функції впливають із рейтингу пісень користувачів. Приклад пояснення у рекомендаційній системі наведено на рисунку 1.2. Тут користувачу пропонується по одній пісні (верхній пункт) і він висловлює свою думку натискаючи на елемент "великий палець вгору" або "великий палець вниз", цей спосіб також може розглядатися як числовий рейтинг.

Інтерфейс в цьому прикладі відіграє важливу роль у формуванні впливу на користувача та може надати якісну оцінку його діям. Також частину від впливу відіграє дизайн та кольорова складова у поясненні на прикладі рекомендаційної системи Pandora.

Рисунок 1.2 – Пояснення у рекомендаційній системі сайту Pandora

Якщо спростимо це твердження, можна сказати, що алгоритми на основі вмісту враховують схожість між елементами на основі оцінок користувачів, але враховуючи властивості елементів. Аналогічно, пояснення стилю на основі вмісту базуються на властивостях предметів. Наприклад, рекомендаційна система MovieXplain обґрунтовує рекомендації щодо фільму відповідно до того, хто, на їх думку, є улюбленим актором користувача. Приклад наведений у таблиці 1.1. Хоча основний підхід насправді є гібридом спільних підходів та підходів, які базуються на вмісті, стиль пояснення передбачає, що вони обчислюють подібність між фільмами відповідно до наявності функцій у фільмах з високою оцінкою. MovieXplain вирішили представити користувачам кілька рекомендацій та пояснень, які можуть підходити більше в тому випадку, якщо користувач хотів би зробити вибір між фільмами залежно від інформації, наведеної в поясненнях (наприклад, відчувати себе комфортніше при перегляді фільму з Гаррісоном Фордом у головній ролі ніж Брюсом Віллісом). Модель взаємодії базується на рейтингах предметів.

Звідси можна побачити приклад використання пояснення та його основний посил, який будується на взаємодії користувача з системою для отримання найкращого варіанту рекомендації фільму, як наведено в прикладі.

Таблиця 1.1 – Приклад пояснення у MovieXplain

Назва фільму	Причина по якій його показують	Де з'являється
1	2	3
Indiana Jones and the Last Crusade (1989)	Ford, Harrison	5 фільмів які ви оцінили
Die Hard 2 (1990)	Willis, Bruce	2 фільма які ви оцінили
На основі	Tags ordered by relevance or	MovieXplain.

контенту	preference	com
----------	------------	-----

Також пропонується більш незалежний від домену підхід, який надає аналогічний показник на основі ключових слів або тегів, визначених користувачем. Пояснення, використані у цьому підході, використовують зв'язок між ключовими словами та елементами (релевантність тегу) та взаємозв'язок між тегами та користувачами (перевага тегів) для надання рекомендацій. Сортування тегів показано на рисунку 1.3. Вподобання тегів або те, наскільки тег доречний для даного користувача, можна розглядати як форму пояснення на основі вмісту, оскільки це середньозважена оцінка даного користувача фільмами з цим тегом. Релевантність тегу або наскільки релевантне ключове слово для рекомендації товару, з іншого боку, це співвідношення між сукупними перевагами користувачів щодо тегу та їхнім уподобанням щодо фільму, з яким тег пов'язаний. У цьому прикладі показ рекомендацій як одного з верхніх елементів дозволяє користувачеві переглядати велику кількість тегів, пов'язаних із цим елементом. Модель взаємодії знову базується на числових рейтингах.

Рисунок 1.3 – Сортування тегів за релевантністю

Пояснення також можуть не включати згадування значних властивостей продукту або послуги і зосереджуватись насамперед на подібних елементах, що використовуються для надання рекомендацій. Таким чином, використовувані предмети розглядаються як випадки для порівняння, що призводить до пояснень стилів на основі кейсів. Ми зазначаємо, що системи CBR сильно відрізняються щодо алгоритму рекомендацій. Наприклад, рекомендація FINDME базується на критиці, а рейтинг предметів у рекомендації базується на їх присутності в планах подорожей користувачів, які висловили подібні інтереси. Хоча ці системи CBR використовували різні

методи для представлення своїх пояснень, нагадаємо, що цей розділ та розділи, що описують інші стилі пояснення, зосереджені на стилі пояснення, а не на фактичному базовому алгоритмі. Таким чином, кожна з цих систем могла теоретично мати стильове пояснення на основі конкретного випадку.

Насправді, у цій главі ми вже бачили тип стильового пояснення на основі реєстру, „стильове пояснення на основі впливу”. В цьому випадку вплив елемента на рекомендацію обчислюється, базуючись на різниці в оцінці рекомендації з цим елементом та без нього. У цьому випадку рекомендації були представлені як головний пункт, припускаючи взаємодію на основі рейтингу. В іншому дослідженні було обчислено схожість між рекомендованими елементами і використано подібні предмети як обґрунтування рекомендації щодо найважливіших предметів у поясненнях „вчитися на прикладі”

Для пояснень, що базуються на знаннях та стилях, передбачуваним входом у механізм рекомендацій є опис потреб або інтересів користувача *u*. Потім механізм рекомендації визначає відповідність між потребами елемента *i* та *u*. Система рекомендацій, заснована на знаннях, враховує, такі властивості камери, як пам'ять, роздільна здатність та ціна. Загалом ці властивості відображають наявні опції, а також уподобання користувача. Ця система може пояснити рекомендації щодо вибору камери таким чином: "У цієї камер менше пам'яті та нижча роздільна здатність але вона доступніша за ціною". Тут рекомендації представлені як форма структурованого огляду, що описує конкуруючі варіанти, а модель взаємодії передбачає, що користувачі можуть вносити зміни в елементах рекомендації. Інформація на основі аргументів представлена на рисунку 1.4.

We recommend: "The Cars"

Your Predicted Rating: ★★★★★

Your ratings for similar artists

- + 'Fun For All'
- + 'Atari Teenage Riot'
- + 'Bacers'
- + 'Death Cab'
- + 'Rise Against'
- + 'Funny Boys'
- 'The Big Band'

+ Liked this artist - Didn't like this artist

Рисунок 1.4 – Інформація на основі конкретних аргументів.

Подібним чином, у системі на основі аргументів, користувачі поступово вказують (і модифікують) свої уподобання, поки не буде досягнуто найвищої рекомендації. Ця система може генерувати такі пояснення, які наведені нижче, щодо рекомендації відпочинку під назвою "Курорт 574": "Кращий вибір: Курорт 574 відрізняється від вашого запиту лише ціною і є найкращим вибором, незалежно від того, якому транспорту, тривалості відпочинку чи типу житла ви віддаєте перевагу".

Для пояснень, пов'язаних із демографічними стилями, очевидним входом в механізм рекомендацій є демографічна інформація про користувача U . З цієї інформації алгоритм рекомендацій визначає користувачів, які демографічно схожі на U . Прогноз для рекомендованого елемента I екстраполюється з того, як попередні користувачі оцінили цей елемент, і наскільки вони схожі на U .

Огляд ряду систем, які використовують демографічний фільтр, показав що на практиці їх кількість мінімальна. Ми могли знайти лише одну, яка пропонує пояснення: "Для дітей дана локація буде цікавою, тому що вона не вимагає глибоких знань, але потребує уважності, а візит буде зручним та коротким. Вас ця локація зацікавить із-за історичної цінності та зручності. Для людей з обмеженими можливостями ця локація буде дуже цікавою та має високу історичну цінність". У цій системі рекомендації пропонувались як

структурований огляд, класифікуючи місця для відвідування відповідно до їх придатності до різних типів користувачів ті їх стилів поведінки (наприклад, дітей, людей з обмеженими можливостями). Потім користувачі можуть додавати ці елементи до свого маршруту, але в цьому форматі не існує моделі взаємодії, яка модифікує подальші рекомендації.

Наскільки нам відомо, не існує інших експертних систем, які б використовували пояснення демографічного стилю. Можливо, це пов'язано з чутливістю демографічної інформації; ми можемо уявити, що більшість користувачів не хотіли б, щоб їм рекомендували товар або послугу на основі їх статі, віку чи етнічної приналежності.

1.3 Дослідження методів побудови пояснень в інтелектуальних системах

Як ми знаємо, існує значна кількість KBS, встановлених у компаніях, але не їх не використовують. Існує вагома кількість аргументів, які намагаються пояснити потреби використання KBS. Одною простою причиною відсутності потреби використання KBS може бути недовіра кінцевого користувача щодо його особистої безпеки при роботі з системою. Якщо система буде мати успіх імплементування, компанії не потрібно тримати у штаті велику кількість працівників. Однак більшість KBS призначені як помічники професіоналів, та зовсім не як їх заміщення.

Впровадження нового KBS в організацію, як правило, призводить до модифікації в робочій практиці, що, як правило, викликає дискомфорт у працівників. Також часто виникають проблеми з взаємодією KBS із класичним програмним забезпеченням в самій організації, а також з тим, що у KBS часто працюють з людьми, які володіють технологіями на низькому рівні. Щоб зменшити негативний ефект від низки проблем, важливо залучити кінцевого користувача до всього процесу проектування KBS. Приклад цільової аудиторії KBS показано на рисунку 1.5.

Залучення кінцевого споживача у процес розробки KBS не тільки призводить до створення системи, яка працює так, як вподобає кінцевий користувач, але й дозволить органічно інтегруватися з існуючим програмним забезпеченням і робочими процедурами. Насамперед цей процес повинен створити відчуття володіння кінцевим продуктом, що зменшує недовіру щодо безпеки у процесі роботи.

Рисунок 1.5 – Цільова аудиторія KBS

У підході KBS важливим аспектом є розуміння користувачем моделі, що використовуються системою, і впевненість у вихідних даних системи. Загальноприйнятим підходом до спроби підвищити впевненість у результатах роботи системи є включення пояснення результату до результатів роботи системи. Пояснення існують як важлива особливість KBS, які часто не інтегруються з іншими типами комп'ютерних систем. Вони допомагають обґрунтувати передбачення у складних ситуаціях, де ситуація не до кінця очевидна або використовуються складні евристичні методи. Однак багато початкових KBS, які імплементували пояснення, не надто ефективно виконували функцію переконання користувачів. Пояснення часто були надані лише від системи до користувача, без втручання спеціаліста у побудову пояснення і дуже часто пояснення відхилялись від заданої цілі, оскільки кінцевий користувач не міг зрозуміти пояснення.

Процес формування ефективних та зрозумілих пояснень є складним завданням, ніж може здатися на перший погляд. Основна складність полягає в тому, що різні користувачі розуміють певні типи пояснень краще за інших. Одним із поширених методів спрощення типу пояснень для різних людей є створення одного типу пояснень для експертів, а іншого для новачків. Дослідження показали, що експерти, як правило, віддають перевагу

поясненням низького рівня, тоді як новачки віддають перевагу поєднанню пояснень вищого рівня, довідковій інформації, та поясненням низького рівня.

Також дослідження свідчать, що через відсутність первинних знань про продукт користувачі на початковому рівні мають необхідність в поясненнях для розуміння результатів роботи KBS, ніж експерти, вони зазвичай використовують пояснення для розуміння і віддають перевагу виправданому стилю пояснення. Крім того, експерти в області, як правило, використовують пояснення в аномальних ситуаціях, оскільки вони частіше розпізнають аномалії, ніж користувачі-початківці. Експерти-користувачі, як правило, використовують стиль пояснення (як?) для отримання детального пояснення, ніж користувачі-початківці.

На розробку моделей користувачів для індивідуальних пояснень зазвичай відводять багато часу. Існує підхід, який використовує стереотипи. Використовуючи стереотипи, рівень деталізації підлаштовується під кожен стереотип. Правила мають різні рівні, пов'язані з ними, і запускаються, якщо користувач належить до цього рівня. Хоча цю систему легше впровадити, ніж користувальницькі моделі, існує проблема, якщо користувач не відповідає стереотипу, пояснення буде незадовільним.

Дослідження показали, що використання пояснень загалом покращує прийняття користувачем KBS. Однак подальший аналіз підходу стверджує, що помилково розроблені для різних користувачів пояснення є, мабуть, гіршими, ніж надання будь-яких пояснень. Але що дає гарне пояснення? У наступному підрозділі розглядаються деякі аспекти гарного пояснення.

Методи побудови пояснень представлені на рисунку 1.6.

Рисунок 1.6 – Методи формування пояснень

З деякими KBS, такими як моделювання та нейронні мережі, існують складності до надання пояснення для підтримки прогнозів. Це пов'язано з тим, що більшість з цих систем трактується як "чорні скриньки", при цьому

вихідна інформація надається без жодного пояснення того, як був отриманий результат. Науковці вивчають використання імітаційних моделей у сільськогосподарському секторі для допомоги прийняття рішень для фермерів. У сільськогосподарській громаді існує загальна недовіра до KBS. Для того, щоб подолати початкову недовіру, система повинна давати правдоподібні пояснення будь-яких рішень, які вона приймає, і взагалі використовувати термінологію та логіку, зрозумілі та перевірені кінцевим користувачем.

Архітектура системи (яка називається EXPLAIN) складається з інтерфейсу користувача, імітаційної моделі, моделі пояснення та моделі користувача, як показано на рисунку 1.7. Інтерфейс користувача має три задачі: запитувати у користувача відповідні входи до моделювання, отримувати знання про поточного користувача, та виведення результатів та пояснення моделювання для кінцевого користувача. Модуль симуляції проводить моделювання на введенні користувачами данні та виводить результати до модуля пояснення. Використовуючи вхідні дані від імітаційного модуля разом із вхідними даними модуля користувача, що містить інформацію про користувача, пояснювальний модуль створює індивідуальне пояснення. Потім це пояснення передається користувачеві через інтерфейс.

Модуль пояснення створює персональне пояснення, спочатку створюючи внутрішнє представлення моделювання. Потім, використовуючи модуль користувача, він персоналізує представлення, яке потім передається через генератор природних мов перед поверненням результату користувачеві.

Використання генератора натуральної мови та модуля користувача для створення пояснення симуляції створює пояснення, яке є зрозумілим для читання мовою та стандартами, зрозумілим для користувача. Описаний процес повинен призвести до загального підвищення рівня довіри в передбачення для користувача.

Окрім усього іншого, точні пояснення можуть допомогти підвищити довіру та лояльність користувачів, підвищити рівень задоволеності, полегшити пошук нових користувачів, їх вподобання, і переконати їх спробувати або придбати рекомендований товар.

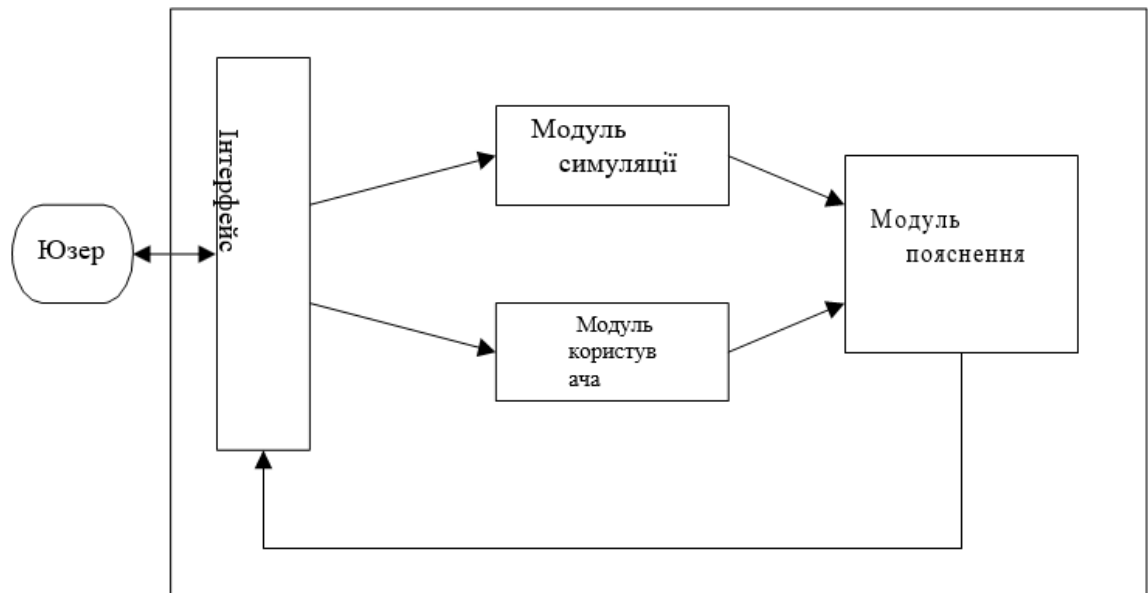


Рисунок. 1.7 – Архітектура системи EXPLAIN

Системи CBR пропонують зовсім інший підхід до побудови пояснень, ніж у інших типах експертних систем. Вхідні дані до системи, що базується на конкретних випадках, не перетворюються на правила, як у системі, що базується на правилах, або на складну взаємопов'язану мережу перцептронів, як у нейронних мережах, але вони зберігаються у вихідному вигляді, як і були введені. Перевага підтримання випадків у первісному вигляді, полягає в тому, що "рішення нової проблеми можна пояснити, просто представивши випадок, який був пристосований для створення рішення". Якщо процес адаптації не ускладнений, відмінності між рішенням та випадком проблеми повинні бути мінімальними, щоб користувач міг легко судити про справедливість рішення.

Дослідницьку роботу з роз'яснення, що ґрунтується на конкретних випадках (СВЕ), як правило, можна класифікувати за підходами, що

ґрунтуються на знаннях та потребують знань. Наукомісткий підхід до СВЕ включатиме такі механізми, як висновок, заснований на правилах або на основі моделей, який можна використовувати для створення пояснень. Серед ранніх прикладів цього підходу є робота SWALE; нещодавно з'явилася Система адаптації кейсів на основі моделі (MoCAS) , абстракція та вдосконалення плану в інтегрованій системі (PARIS-система), і нарешті система DIRAS для оцінки довгострокові ризику діабету.

З іншого боку, підхід "Knowledge Light", як правило, використовується в системах пошуку лише або в інтерактивній адаптації. Хорошим прикладом підходу "Knowledge Light" до CBR є система CARES для прогнозування рецидивів колоректального раку. CBR успішно використовується при розробці діагностичних систем, заснованих на представленні значення-характеристики. Вид представлень у CBR показано на рисунку 1.8.

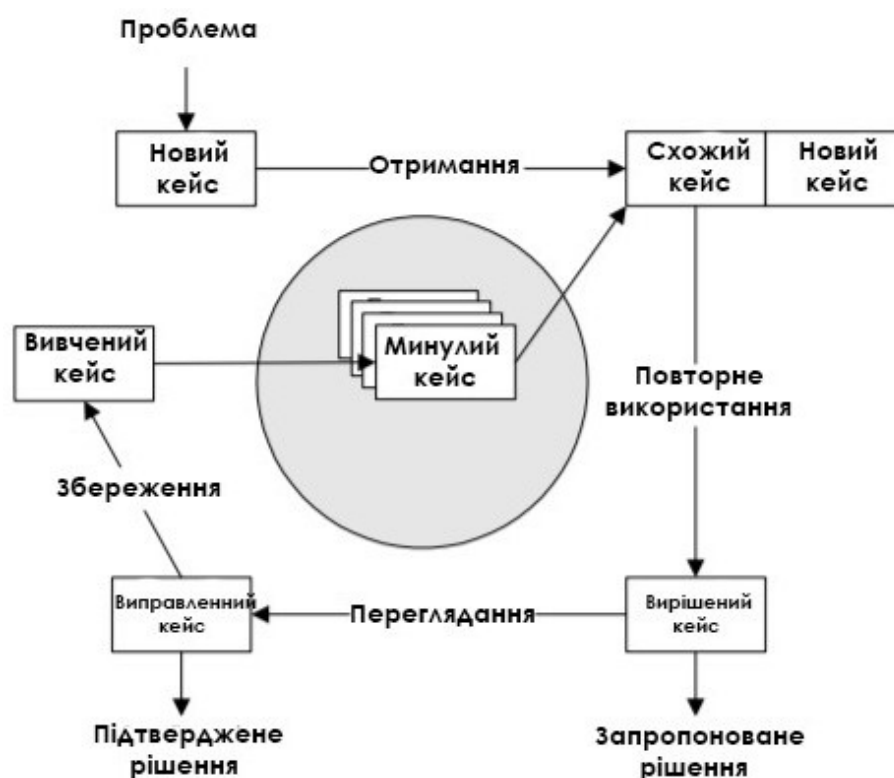


Рисунок 1.8 – Вид представлень у CBR

У системах CBR, які використовують просте представлення, що базується на характеристиці, отримані випадки можуть бути використані для пояснення наступним чином:

"Система передбачає, що результатом буде X, оскільки таким був результат у випадку C1, який відрізнявся від поточного випадку лише значенням ознаки F, яка становила f2 замість f1.

На додаток результат у C2 також був X ... "

Пояснення в цих термінах (тобто виражене в словниковому запасі особливостей справи) не завжди буде адекватним, але в деяких ситуаціях, наприклад, при обґрунтуванні медичних рішень, воно може бути корисним.

MoCAS використовує подібність на основі пояснень для пошуку та адаптації випадків. У подібності, що ґрунтується на поясненнях, подібність двох випадків базується на подібності їхніх пояснень. Для цього знання про домен моделюються на декількох рівнях абстракції. Це моделювання повинно мати можливість переключення між різними рівнями шляхом перетворення репрезентативних термінів з одного рівня на інший. Пояснення щодо нижчого рівня абстракції є більш детальним і містить більше правил, ніж пояснення щодо більш високого рівня абстракції. Тому пояснення двох випадків можуть дуже сильно відрізнятися на низькому рівні абстракції, але можуть бути однаковими на вищому рівні абстракції. Чим нижчий рівень абстракції, на якому два пояснення однакові, тим вища оцінка їх подібності. На рисунку 1.9 зображено приклад підходу.



Рисунок 1.9 – Приклад підходу у CBR

Цукровий діабет - одна з найпоширеніших хвороб у світі. В даний час ця хвороба уражає приблизно 3% європейського населення. Незважаючи на те, що з діабетом можна впоратись із певними змінами способу життя, погане відстеження хвороби призведе до мікро ускладнень (сліпота, ниркова недостатність або полінейропатія) та макро ускладнень (гангрена та ампутація, загострення ішемічної хвороби серця або інсульту). Метою DIRAS є прогнозування індивідуальної оцінки ризику для пацієнта з діабетом розвитку одного з цих ускладнень.

DIRAS використовує "ліниву індукцію описів" (LID) для розробки висновку про ризик, що стосується кожного конкретного виду ускладнень. LID працює, обираючи найбільш дискримінаційну функцію (використовуючи відстань Лопеса де Мантараса (RLM)). Це значення функції додано до опису. З цього моменту використовуються лише випадки з однаковим значенням ознаки. Система перевіряє, чи всі наступні випадки мають однаковий рівень ризику для ускладнення, яке досліджується. Якщо ні – наступною обирається найбільш дискримінаційна риса решти випадків. Цей процес триває до тих пір, поки сукупність решти випадків не матиме однаковий рівень ризику. Опис, що використовується для пояснення класифікації, містить усі дискримінаційні ознаки, що використовуються під час процесу, та їх значення.

DIRAS також формує звіт, який може бути корисним для непрофесійних лікарів, щоб мати змогу відстежувати динаміку діабетичних пацієнтів. Цей звіт складається з 4 розділів:

- особисті дані (наприклад, вік, тип діабету, рік діагностики діабету);
- оцінки заходів (наприклад, рівень холестерину);
- інформація про макроскладнення;
- інформація про мікроскладнення;

MOCAS та DIRAS є прикладами наукомістких систем. Однак деякі люди стверджують, що "наукомісткий ЦБР відсутній у сенсі ЦБР, тобто потенціал, який CBR повинен вдосконалити у розробці знань, маніпулюючи

випадками, що складаються з частин знань". Національний університет Сінгапуру спільно з Сінгапурською лікарнею розробив систему CARES, щоб мати можливість прогнозувати ризик рецидивів колоректального раку та запропонувати відповідний режим спостереження за пацієнтом. Без цього прогнозу пацієнт може зіткнутися з непотрібними, болісними та дорогими оглядами. Точна система прогнозування значно зменшить їх кількість.

CARES використовує процес відповідності, відомий як найближчий сусід. Після вступу до нового випадку, система розглядає всі існуючі випадки в системі та отримує найбільш схожі випадки. Враховуючи новий випадок X та отриманий випадок Q , для кожної ознаки f система обчислює ступінь збігу між значеннями ознаки, використовуючи функцію подібності.

Під час процесу пошуку система веде список десяти найбільш подібних випадків у базі даних. Потім ці випадки відображаються лікарю для перегляду. З цих найбільш подібних випадків система також виробляє свої прогнози та схему подальших дій та пояснює свої рішення, використовуючи отриманий випадок як основу.

Слід також зазначити, що лікарі могли вибрати та відрегулювати, яку функцію включати при перевірці подібності, а також яку вагу надавати характеристиці. Це дозволяє системі базувати свої рішення на подібних критеріях, подібно до лікаря, виводячи найбільш інформативні висновки для лікаря. У системі CARES усі 10 отриманих випадків не повинні мати такий самий результат, як висновок, представлений системою. Існує велика вірогідність того, що вони уникають таких ускладнень, однак, маючи змогу переглядати отримані випадки, лікар може врахувати будь-які аномалії, які є у 10 випадках.

Також важливим параметром є корисність. Постає питання а наскільки корисні ці пояснення? Чи користувачі так само будуть відчувати впевненість переглянувши лише один прогноз? Чи їм насправді потрібне пояснення? І якщо вони їх розглядають, які пояснення їм найбільше подобаються? Чи вони віддають перевагу поясненням на основі конкретних випадків (СВЕ), через їх

узгодженість з тим, що спостерігали психологи при вирішенні природних проблем, як це роблять користувачі, або пояснення на основі правил (RBE) є більш прийнятним?

В подальшому ми досліджуємо корисність пояснень як в CBR, так і в методах, заснованих на правилах (RBR). Науковці використовують домен рівня алкоголю в крові (BAL), де була створена система, яка могла передбачити особу BAL за допомогою таких характеристик, як кількість випитого, тривалість пиття, кількість їжі, стать та вага. Кожен із проблемних випадків був представлений тричі, один раз із передбаченнями та поясненнями на основі конкретних випадків, один раз із передбаченнями та поясненнями, заснованими на правилах, і один раз із передбаченнями та без пояснення. Пояснення, що ґрунтуються на правилах та випадках, були представлені разом, але порядок змінювався, щоб уникнути будь-яких упереджень через знайомство з системою. Випробовуваних просили оцінити, наскільки вони переконані, поясненнями за 5-бальною шкалою (Ні, Можливо Ні, Можливо, Можливо Так, Так). При оцінці результатів ці бали інтерпретувались як числові значення 1-5.

Неправильний прогноз у поєднанні з поганим поясненням був включений до кожної категорії, щоб допомогти оцінити увагу, яку суб'єкти приділяють оцінці. Середня оцінка для цих невірних прогнозів становила 1,5, тоді як середня оцінка для інших прогнозів становила 3,9 (за шкалою 1-5). Рейтинги цих невірних прогнозів не враховувались далі в оцінці. На рисунку 1.10 показано середні показники решти рейтингів.

Дві речі, на які слід звернути увагу, - це сильна ефективність СВЕ та той факт, що прогнози без пояснень все ще були досить переконливими. Статистичні тести були проведені на даних, і парний t-тест показав, що СВЕ був кращим, ніж RBE (значення $P = 0,0005$) і краще, ніж "Без пояснення" (значення $P = 0,005$).

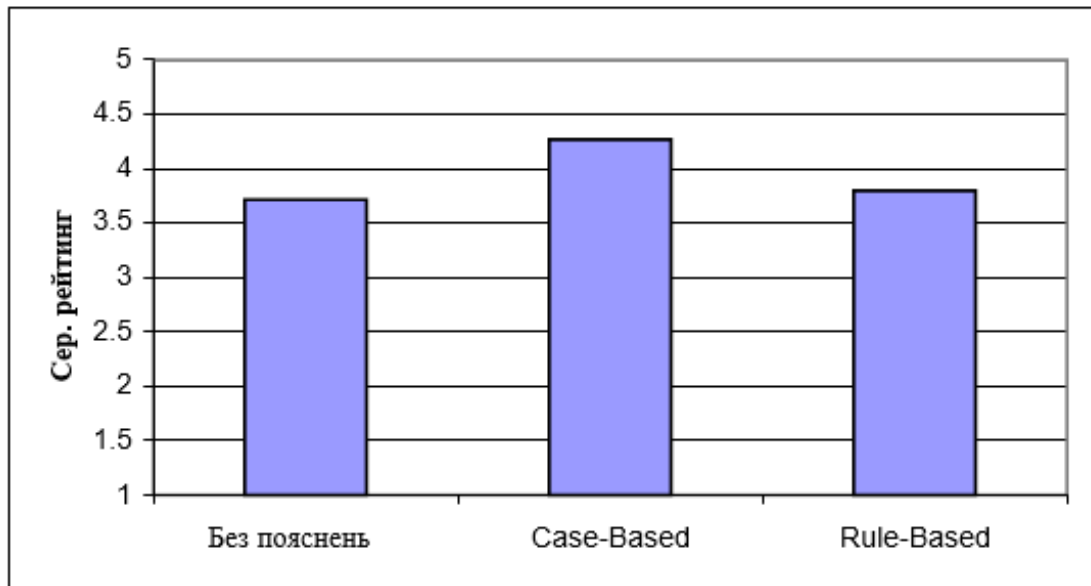


Рисунок 1.10 – Приклад середніх оцінок трьох альтернативних систем прогнозування та пояснення

Незважаючи на те, що оцінка надавала перевагу підходу, що базується на конкретних випадках, це не означає, що цей підхід завжди буде найкращим. Слід пам'ятати, що хоча пояснення, яке базується на конкретних випадках, було кращим, воно в результаті було лише трохи кращим, ніж пояснення, засноване на правилах, яке незначно відрізнялося за відсутності пояснення.

1.4 Постановка задачі дослідження

Об'єктом дослідження в рамках магістерської атестаційної роботи є процес надання пояснень до рекомендацій для користувача до товарів або послуг.

Предметом дослідження є методи надання пояснень у проектах рекомендаційних систем.

Метою даної роботи є дослідження методів надання пояснень до рекомендацій з урахуванням темпоральної динаміки зміни вподобань користувача.

Для досягнення мети, необхідно вирішити такі задачі:

- дослідження особливостей рекомендаційної системи в комерційних продуктах;
- дослідження методів надання пояснень до рекомендацій;
- аналіз методів надання пояснень до рекомендацій в умовах зміни звичок користувача;
- удосконалення методу надання пояснень до рекомендацій з урахуванням темпоральної динаміки;
- експериментальна перевірка удосконаленого методу надання пояснень до рекомендацій у проектах рекомендаційних систем.

2. ВИДИ ПОЯСНЕНЬ ТА ПРЕДСТАВЛЕНЬ У РЕКОМЕНДАЦІЙНИХ СИСТЕМАХ

2.1 Методи побудови пояснень в рекомендаційних системах на правилах

У цьому розділі ми описуємо пояснення, натхненні певним базовим алгоритмом та різними "стилями пояснення". Треба зауважити, що пояснення можуть відповідати "стилю" певного алгоритму, незалежно від того, як саме такі рекомендації були отримані чи обчислені. Іншими словами, стиль пояснення може відображати або не відображати основний алгоритм, за яким обчислюються рекомендації. Часто існує розбіжність між способом отримання рекомендацій та стилем поданих пояснень. Отже, різні методи пояснень не будуть узгоджуватися з метою прозорості, але можуть підтримувати інші цілі пояснення. Приклади пояснень у комерційних та академічних системах представлено у таблиці 2.1.

Таблиця 2.1 – Приклади пояснень у комерційних та академічних системах, упорядкованих за стилем пояснення

Стиль пояснення	Приклад пояснення	Система
1	2	3
На основі випадку	“Because you have selected or highly rated: Movie A”; “Recommended By User X for Book A”; A list of similar movies the user has rated highly in the past	LoveFilm.com
Колаборативна	“Customers Who Bought This Item Here Also Bought . . . ”	Amazon.com
На основі контенту	Tags ordered by relevance or preference	MovieExplain.com

Продовження таблиці 2.1

1	2	3
Діалогова	Dialog e.g. “Where would you like to eat?” “Oh, maybe a cheap Indian place.”; Dialog e.g. “What kind of movie do you feel like?” “I feel like	Advisor.com

	watching a thriller.”	
Демографічна	“For children it is much eye- catching, it requires low back- ground knowledge, it requires a few seriousness and the visit is quite short. For yourself it is much eye- catching and it has high historical value. For impaired it is much eye- catching and it has high historical value.”.	Intrigue.com
На основі знань	“Less Memory and Lower Resolution and Cheaper”; “Case 574 differs from your query only in price and is the best case no matter what transport you choose”; “. . . cameras capable of taking pictures from very far away will be more expensive . . . ”	Owikshop.com CameraShop.com

Прозорість - не єдина мета пояснень, яку слід враховувати, вибираючи стиль пояснення. Наприклад, для певної системи можна виявити, що користувачі більше задоволені поясненнями стилю на основі вмісту, навіть незважаючи на те, що пояснення на основі критики є більш ефективними. На сьогодні порівняння стилів пояснення, їхньої ефективності та цілей пояснення є не до кінця підсумованим. Тільки Хінгстон порівняв зрозумілість і можливість вивчення різних стилів пояснення, хоча в інших випадках на пояснення також безпосередньо впливали різні базові алгоритми. Однак інші дослідження розглядали вплив варіацій інтерфейсів пояснень на різноманітність пояснювальних цілей.

Незважаючи на це, основний алгоритм механізму рекомендації певною мірою впливатиме на типи пояснень, які можна створити. Науковці узагальнюють найчастіше використовувані стилі пояснень (на основі випадків, змісту, співпраці, демографічно, знань та корисності) з прикладами кожного з них. Для комерційних систем, де конфіденціальна інформація грає ключову роль, можна запропонувати метод на здогадах. Хоча діалогові системи описані як підхід, ми розглядаємо діалогові системи більше як стиль взаємодії, ніж конкретний алгоритм.

Для опису інтерфейсу між рекомендаційною системою та компонентом пояснення ми використовуємо позначення: U - це набір користувачів, чий

уподобання відомі, а u - користувач, для якого потрібно сформувавши рекомендації. I - це набір елементів, які можна рекомендувати, та i - це елемент, для якого ми хотіли б передбачити переваги u .

Для стилю пояснень, заснованих на спільній роботі, передбачуваними вхідними даними для механізму рекомендацій є рейтинги користувачів у елементів у I . Ці оцінки використовуються для визначення користувачів, подібних за рейтингами до u . Таких подібних користувачів часто називають "сусідами", оскільки підходи найближчих сусідів зазвичай використовуються для обчислення подібності. Потім прогноз для рекомендованого товару екстраполюється з оцінок сусідів i .

У комерційному відношенні найбільш відомими поясненнями у стилі спільної роботи є ті, які використовує Amazon.com: "Клієнти, які придбали цей товар, також купили ...". Це пояснення передбачає, що користувач переглядає товар, який його вже цікавить. Це означає, що система знаходить подібних користувачів (які купили цей товар), а також отримує та рекомендує товари, які купували подібні користувачі. Рекомендації представлені у форматі, подібному до верхнього пункту. Крім того, це пояснення передбачає модель взаємодії, згідно з якою рейтинги неявно виводяться через поведінку покупців.

Науковцями було запропоновано 21 інтерфейс пояснення з використанням тексту, а також графіки. Ці інтерфейси різнилися між собою відповідністю змісту та стилю, але ряд цих пояснень безпосередньо посилався на поняття сусідів. Наприклад, було описано, як сусіди оцінили даний (рекомендований) фільм, гістограму з рейтингами "добре", "середнє" та "погано", згрупованими в різні стовпці. Знову ж таки, ми бачимо, що це пояснення дається як для конкретного способу рекомендування предметів, так і для конкретної моделі взаємодії: це одна рекомендація (або найважливіший елемент, або один елемент із списку топ-N), і передбачає, що користувачі отримують інформацію про рейтинг товарів.

2.2 Опис динаміки вподобань користувачів з використанням темпоральних правил

Покращений метод видачі пояснень базується на методі побудови пояснень для рекомендаційних систем на основі темпоральної динаміки вибору користувача та рейтингу

Рекомендаційні системи, призначені для створення персоналізованого переліку предметів, таких як товари чи послуги, які відповідають інтересам поточного користувача. Такі системи формують рекомендації на основі подібності тем або схожих уподобань користувачів. Інформація про продаж товарів, а також рейтинги цих товарів, встановлені користувачами, використовуються як вхідні дані для побудови рекомендацій. Рекомендаційні системи полегшують роботу користувача, що дозволяє збільшити продажі. Тому такі системи широко використовуються при продажі товарів, перегляді фільмів, продажу туристичних турів, рекомендуванні конференцій тощо.

У процесі функціонування рекомендаційних систем виникають проблеми, пов'язані з неповнотою або навмисним спотворенням вхідної інформації.

Проблема неповної інформації або проблема холодного старту виникає при появі нових, "холодних", а також нерегулярних користувачів системи рекомендацій. Оскільки нові користувачі ще не зробили вибір тем, система рекомендацій не має достатньої інформації про їх уподобання. Тому система не може скласти точних рекомендацій, а замість предметів, що цікавлять користувача, може запропонувати просто популярні продукти.

Проблема спотворення вхідної інформації або проблема шилінг-атак полягає в тому, що зловмисник штучно знижує або підвищує рейтинг деяких предметів. Підвищення рейтингу вибраних товарів дозволяє збільшити їх продаж, а зниження - зменшити продажі конкурентів. Зміна рейтингу в результаті нападу може призвести до видачі нерелевантних рекомендацій та

до повного збою системи рекомендацій. Користувач перестає довіряти такій системі.

Для вирішення проблеми холодного старту використовуються гібридні методи, які поєднують використання спільної фільтрації з аналізом контекстних даних. Ця комбінація доповнюється можливостями активного навчання. Гібридні методи також враховують зміни вимог користувача з часом і. е. тимчасова динаміка їх уподобань. У цьому випадку використовуються часові обмеження, часовий графік, багат шаровий графік, модель нейронної мережі.

Методи виявлення шилінг-атак орієнтовані на виявлення шаблону атакуючого користувача на основі статистичних методів та методів машинного навчання, а також часової динаміки його уподобань.

Загальною особливістю розглянутих підходів є те, що вони вибирають товари для "холодного" споживача, розпізнають спотворення рейтингу без активної участі користувача. Це може зменшити його довіру до системи рекомендацій та відмовитись від використання рекомендації.

Для підвищення довіри користувачів використовується концепція екранів пояснення рекомендацій. Ряд критеріїв запропоновано в системі для оцінки пояснень. Ці критерії характеризують довіру користувача до рекомендацій, вплив пояснень на ефективність продажів, ступінь простоти використання рекомендацій для користувачів. Оцінка пояснень на основі опитувань користувачів, а також повторний вибір користувачів після отримання пояснень свідчить про зменшення відмов у виборі пунктів після отримання пояснень.

Загалом, існуючі підходи до побудови пояснень не враховують зміни в уподобаннях користувачів з часом. Однак, враховуючи відсутність інформації про нових користувачів, а також спотворення рейтингу в результаті шилінг-атак, важливо сформулювати найбільш відповідні рекомендації. Для цього необхідно враховувати тимчасову динаміку у вимогах користувачів. Для вирішення цієї проблеми доцільно

використовувати часові залежності, що відображають зміну кількості товарів або послуг, що продаються з часом.

Метою цього розділу є розробка методу побудови пояснень у системах рекомендацій на основі врахування тимчасових залежностей, що описують зміни в інтересах користувача щодо процесів придбання товарів протягом обраного інтервалу часу та персональних доповнень.

Адаптація правил для вирішення проблеми побудови рекомендацій полягає в обчисленні їх ваги таким чином, щоб враховувати не лише популярність певного товару, а й зміни кількості одиниць, проданих в електронній системі комерції.

Цей метод використовує часові правила типу F (Future), оскільки часовий оператор типу F дозволяє асоціювати пару довільних моментів в одному часовому потоці. Поєднання цих правил дозволяє описати зміну продажів за всі моменти, що належать до певного інтервалу часу.

Використовуючи часові правила, метод описує динаміку споживчого попиту протягом заданого інтервалу часу ΔT . Кінцевою точкою цього інтервалу є поточний час. Тривалість ΔT залежить від групи товарів та предметної області та повинна відображати цикли збільшення / зменшення попиту споживача. Для товарів, які продаються циклічно, 2–3 рази на рік, цей інтервал становить кілька місяців. Прикладами таких продуктів є туристичні екскурсії, одяг. Для товарів у галузі високих технологій цей інтервал залежить від частоти випуску і становить рік і більше.

Період ΔT ділиться на послідовність інтервалів $\Delta \tau_m$. Інтервали $\Delta \tau_m$ відображають рівень деталізації часу (години, доби тощо). Для кожного інтервалу $\Delta \tau_m$ розраховується кількість покупок кожного товару. Тоді формується набір F - часових правил r_{jm} , які пов'язують кожен інтервал $\Delta \tau_m$ з кінцевим інтервалом $\Delta \tau_M$. Вага правила залежить від різниці покупок між інтервалами: чим більше змінилася кількість проданих товарів, тим більша вага правила.

Приклад розглянутих часових правил для одного пункту i_j представлений на рисунку 2.1

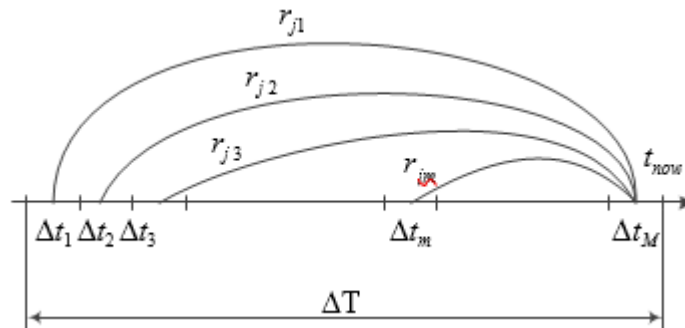


Рисунок 2.1 – Приклад часових правил

Кожне правило r_{jm} описує збільшення / зменшення обсягів продажу товару i_j протягом пари часових інтервалів $(\Delta\tau_m, \Delta\tau_M)$. Весь набір правил $R_j = \{r_{jm}\}$ характеризує зміну обсягу продажу товарів за певний проміжок часу ΔT .

2.3 Метод побудови пояснень з урахуванням змін в уподобаннях користувачів

Розроблений метод призначений доповнити рекомендації поясненнями з метою підвищення ефективності системи рекомендацій.

Ефективність таких пояснень може бути оцінена за допомогою таких показників, як ефективність та довіра. Перший показник визначається через взаємозв'язок між кількістю продажів при використанні рекомендацій з поясненнями, а також на основі продажів щодо рекомендацій без пояснень. Другий показник відображає вплив пояснень на вибір споживачем товарів. Наприклад, якщо після розгляду пояснень користувач обрав новий, більш престижний і дорогий товар, то це означає, що пояснення переконали його зробити дорожчу покупку. Таким чином, підвищення ефективності системи

рекомендацій, що ґрунтується на поясненнях, здійснюється як за рахунок збільшення продажів загалом, так і в результаті збільшення продажів нових, більш технологічних, модних товарів, які мають вищу ціну. Для досягнення зазначеного підвищення ефективності за допомогою пояснень необхідно, щоб користувач змінив свій вибір після прочитання пояснень. Іншими словами, користувач повинен довіряти цим поясненням. Показник впевненості показує, наскільки запропоноване пояснення виправдовує для користувача переваги певного елемента. Таке обґрунтування, як і рекомендації, можна отримати або на основі характеристик цього елемента, або на основі відомого вибору інших користувачів. У цій роботі другий підхід реалізовано.

Розроблений метод генерує пояснення, що відображають часову динаміку попиту клієнта і дозволяють користувачеві зробити усвідомлений вибір, коли бракує або спотворюється інформація про переваги користувача, наприклад, у ситуації холодного старту або у випадку шилінг-атак на систему рекомендацій.

Розроблений метод використовує адаптовану версію зважених часових правил, запропоновану в. Традиційно кожне таке правило встановлює порядок часу для пари подій або дій. Наприклад, послідовність вибору пари товарів у системі електронної комерції. Вага правила залежить від ймовірності його використання. Тому для рекомендаційних систем, вага правила повинна бути пов'язана з популярністю продукту, описаного цим правилом. Набір правил дозволяє описати впорядковану за часом послідовність вибору товарів та послуг користувачем системи рекомендацій. Така послідовність відображає неявні причинно-наслідкові зв'язки між діями користувача. Часові залежності, відповідні цим зв'язкам, можна отримати на основі аналізу журналів. У системах рекомендацій журнали представлені журналами продажів.

Метод як вхід використовує: журнал продажів L ; період часу для обчислення пояснень ΔT ; рівень грануляції часу $\Delta \tau$; підмножина пунктів I рекомендації, щодо яких формуються пояснення.

Кожен рядок l журналу продажів L представлений п'ятіркою (користувач u_k , елемент i_j , кількість придбаних предметів n_j , рейтинг товарів m_j , позначка часу τ_n). Період часу ΔT залежить від категорії позицій, щодо яких формуються рекомендації.

Спосіб включає наступні етапи.

Етап 1. Вибір підмножини $L_{\Delta T}$ записів журналу продажів L для обчислення часової динаміки з підмножини товарів I за певний період часу ΔT .

Етап 2. Нормування кількості покупок по інтервалам $\Delta \tau$. Результатом другого етапу є множина записів Q з нормованою кількістю покупок n_{jm} по виділеним інтервалам часу $\Delta \tau$.

Крок 2. 1. Нормалізація кількості покупок n_j для кожного рядка журналу продажів $L_{\Delta T}$:

(2.1)

Крок 2. 2. Підсумовування покупок для кожного товару $i_j \in I$ для даного рівня грануляції $\Delta \tau$:

(2.2)

Змінна n_{jm} може містити покупки за годину, за день, за тиждень тощо.

Результатом цього кроку є набір , кожен елемент якого містить кількість покупок товар i_j для інтервалу $\Delta\tau_m$:

(2.3)

Етап 3. Розрахунок рейтингових оцінок - для кожного елемента товару i_j :

(2.4)

Етап 4. Формування набору незважених темпоральних правил для кожного предмету, представленого в множині $L_{\Delta T}$:

(2.5)

Етап 5. Розрахунок ваги цих правил W_j як різниця між нормованим кількістю покупок для пар інтервалів:

(2.6)

або для рейтингів:

(2.7)

Етап 6. Розрахунок змін вподобань користувачів з часом по кожному предмету. Ця динаміка представляється сумою ваг темпоральних правил по предмету для всього інтервалу часу:

Етап 7. Формування пояснень у вигляді переліку товарів, для яких динаміка позитивна, тобто .

На відміну від інших методів, мій модифікований метод надання пояснень в проектах рекомендаційних систем надає більш повну картину пояснення рекомендації. Удосконалення полягає в додаванні рейтингів товарів за певний проміжок часу від реальних користувачів, тим самим підвищуючи довіру користувача при виборі товару або послуги.

Результатом роботи є метод побудови пояснень, що доповнює рекомендації числовою оцінкою темпоральної динаміки переваг користувачів для обраного періоду часу. Динаміка інтересів користувачів фіксується за допомогою правил, кожне з яких пов'язує покупки на парі інтервалів часу. Сукупність таких правил показує зростання або падіння продажів в рамках обраного періоду в порівнянні з поточним інтервалом часу.

Метод відбирає підмножину вихідних записів, в яких зафіксовані покупки в рамках заданого часового періоду. На другому етапі ці записи перетворюються у відповідності з обраним рівнем деталізації часу. Кількість покупок нормується, після чого підмножина розбивається на часові інтервали. Результатом другого етапу є множина записів з нормованою кількістю покупок по виділеним інтервалам часу. На третьому етапі формується набір невзважених темпоральних правил для кожного предмета, представленого в множині. Потім розраховуються ваги цих правил як різниця між нормованою кількістю покупок для пар інтервалів. На п'ятому етапі розраховується темпоральна динаміка по кожному предмету. Остання представляється сумою вагів темпоральних правил по предмету для всього інтервалу часу. Результатом методу є пояснення у вигляді переліку предметів із зазначенням їх темпоральної динаміки. Пояснення містить тільки позитивні значення, щоб стимулювати користувача до вибору відповідного предмета.

Даний метод дозволяє користувачам зробити обґрунтований вибір предметів з рекомендованого списку, особливо в ситуаціях спотворення рейтингів і неповноти даних про вимоги поточного користувача.

Відмінність запропонованого методу полягає в тому, що отримані пояснення враховують не тільки рейтинг кожного предмета з рекомендації, а й інтегральну оцінку зміни попиту на предмет за заданий період часу.

Перевага методу полягає в можливості побудувати пояснення в режимі онлайн, з урахуванням найостанніших змін переваг користувачів рекомендаційною системи і без необхідності розглядати конкретні властивості рекомендованих предметів

Недолік методу полягає в тому, що розрахована оцінка зміни попиту залежить від апріорно заданих часових параметрів. На оцінку впливає величина обраного періоду часу, на якому розглядається зміна уподобань користувача. Оцінка також залежить від рівня грануляції часу у вихідних даних.

Метод має обмеження, пов'язані з режимом роботи рекомендаційною системи. Метод не призначений для оффлайн-режиму роботи, оскільки в такому режимі розрахунок переваг користувача потрібно робити заздалегідь, до побудови рекомендацій. Даний же метод використовує найостанніші дані про попит для побудови пояснень.

Метод формує пояснення, яке доповнює показник найбільш популярних товарів за рейтингом. Використана форма пояснень орієнтована на нові товари або вдосконалені, адаптовані їхньою версією. З позицій задоволення призначених для користувача вимог облік темпоральної динаміки дає можливість вибирати товари з циклічним (наприклад, сезонним) попитом, а також перспективні товари, з великим передбачуваним терміном використання.

Метод може бути також застосований при вирішенні завдань, що характеризуються зміною попиту споживачів, наприклад для побудови

наочного пояснення причин зростання трафіку інформаційної системи [19], а також для інформаційної підтримки клієнтів віртуального підприємства [20].

3. ПРОЕКТ РОЗРОБКИ МЕТОДУ НАДАННЯ ПРЕДСТАВЛЕНЬ У ПРОЕКТАХ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

3.1 Опис проекту розробки методу

Для реалізації проекту по розробці методу у якості програмного додатку була зібрана команда, що складається із наступних спеціалістів:

- менеджер проектів;
- програміст;
- тестувальник;

Тривалість проекту по примірній оцінці складатиме 74 робочих днів. Використовується каскадна модель (waterfall) життєвого циклу проекту. Результатом декомпозиції проекту на задачі є наступний перелік обов'язкових задач:

- проектування. Включає в себе процес підготовки вхідних даних для розробника. Складається із визначення вимог які повинні бути у додатку, формування задачі, написання технічного завдання та створення макету графічного інтерфейсу програмного додатку;

- реалізація алгоритму надання пояснень у проектах рекомендаційних систем;

- юніт-тестування алгоритму надання пояснень у проектах рекомендаційних систем;

- розробка графічної частини додатку;
- інтеграція додатку з CMS Tilda;
- юніт-тестування інтеграції додатку;
- первичне тестування додатку;
- розробка дизайну;
- тести інтерфейсу споживача.

3.2 Статут проекту

1. Причини ініціалізації проекту (визначення проблеми, яка буде розв'язана за допомогою розробленого додатку та підстави для ініціалізації проекту в організації)

На сьогоднішній день існує вагома причина, що не дає підприємцям повною мірою надавати якісні пояснення до рекомендацій. І ця причина – не врахування зміни звичок та поведінки користувача упродовж часу. Основною причиною створення цього проекту є мета надати підприємцям удосконалений додаток надання пояснень до рекомендаційних систем.

2. Сутність запропонованої інноваційної ідеї та спосіб її використання для розв'язання конкретної проблеми

Після детального аналізу існуючих алгоритмів та методів надання пояснень у проектах рекомендаційних систем у усіх них був виявлений один суттєвий недолік: усі вони не враховують факт зміни звичок користувача з часом. Але подібні алгоритми не здатні надати доцільні пояснення до рекомендацій на певному часовому проміжку з умовами зміни вподобань користувача. Задача ще більше ускладнюється коли немає вхідних даних про користувача системи. Саме тому, було вирішено розробити метод що самостійно аналізує вподобання користувача з часом та за допомогою оцінки рейтингів надає точне пояснення до рекомендації.

Саме тому було вирішено розробити покращений метод надання пояснень у проектах рекомендаційних систем, та інтегрувати його у популярну CMS - Tilda.

3. Мета проекту: створити зручний та зрозумілий додаток до вже широкоживаної CMS що дозволить швидко та ефективно надавати пояснення до рекомендацій.

4. Очікувані вигоди від проекту

Створення програмного додатку сприяє досягненню цілей стратегії розвитку компанії “Dev warrior”, а саме: розширення компанії за рахунок

виконання власних замовлень; ріст клієнтської бази за рахунок створення доброї репутації компанії, що здатна швидко та якісно випускати актуальні продукти на ринок; фінансова мета: заробіток коштів за рахунок користувачів додатку.

5. Обмеження проекту

Початок проекту – не пізніше 1 вересня 2020 року. Завершення не пізніше 16 Грудня 2020 року.

6. Завдання проекту, критерії їх досягнення

– розробка покращеного методу. Критерієм досягнення вважається підвищення конверсії у категорії товарів;

– розробка додатку. Критерієм вважаємо позитивні відгуки від користувачів системи про досягнення їх очікувань і виконання більшості умов;

7. Межі (зміст) проекту

Межі проекту визначаються списком робіт від початку планування до передачі кінцевого тестування додатку. До проекту не стосуються роботи з розповсюдження додатку та робота з клієнтами.

3.3 Планування проекту

Безперервний процес визначення найкращого способу дій для досягнення поставлених цілей проекту з урахуванням ситуації, що складається. Планування є найбільш важливим процесом управління проектом, визначним у часі всю діяльність по виконанню проекту.

Процеси планування виконуються протягом усього життєво циклу проекту, починаючи з попереднього повного плану в складі концепції проекту і закінчуються детальним планом робіт фази завершення проекту. При цьому відбувається уточнення і деталізація планів по мірі усього прогресу проекту. На стадії планування визначаються методи і засоби управління здійсненням проекту, як цілісної системи, так і в розрізі окремих

її етапів і елементів. Планування логічно пов'язане з іншими важливими стадіями процесу управління, такими як ініціація, організація і контроль виконання, аналіз і регулювання, завершення проекту.

Мета планування полягає в побудові моделі реалізації проекту. Основним результатом стадії планування є статут проекту, який об'єднує результати планування з усіх функцій управління проектом. Цей документ є головним і визначним при здійсненні проекту, він виконує роль моделі (плану) дій і прогнозу етапів здійснення проекту і його оточення. У процесі виконання проекту можуть відбуватися зміни як всередині проекту, так і поза ним. Тому основне призначення планування полягає в безперервному підтримці курсу здійснення проекту на шляху до його успішного завершення.

Об'єктами планування в проекті є: Предметна область, час, вартість, якість, організація, комунікації, ризики, поставки та контракти, зміни, інші компоненти проекту, інтеграційний план, планування предметної області проекту.

Предметна область проекту (Project Scope) - сукупність продуктів і послуг, виробництво яких має бути забезпечене в результаті завершення здійснюваного проекту. Предметну область проекту визначають цілі, результати та роботи проекту. В процесі життя проекту всі складові предметної області проекту можуть зазнавати зміни. Цілі результати, роботи і їх характеристики можуть змінюватися або уточнюватися як у процесі розробки проекту, так і в міру досягнення проміжних результатів.

Злагоджена робота всіх учасників проекту організовується на основі календарних планів або розкладів робіт проекту, основними параметрами яких є: терміни виконання, ключові дати, тривалості робіт і ін.

Календарними планами називають проектно-технологічні документи, що встановлюють повний перелік робіт проекту, їх взаємозв'язок, послідовність і терміни виконання, тривалості, а також виконавців і ресурси, необхідні для виконання робіт проекту. Планування проекту по часових параметрів полягає в складанні різних календарних планів (розкладів робіт),

які відповідають всім вимогам і обмеженням проекту і його частин. Календарні плани складаються на весь життєвий цикл проекту і його етапи, для різних рівнів управління і учасників проекту.

Графік проекту

Етап проекту	Початок	Тривалість	Затримка	Кінець
Постановка задачі	01.09.2020	1	0	01.09.2020
Написання ТЗ	02.09.2020	11	0	12.09.2020
Розробка макету	16.09.2020	9	3	24.09.2020
Розробка дизайну графічної частини додатку	25.09.2020	15	0	09.10.2020
Розробка додаткових модулів	05.10.2020	25	-5	29.10.2020
Розробка кольорових рішень та графіки	03.11.2020	12	4	14.11.2020
Розробка додатку	12.11.2020	12	-3	23.11.2020
Розробка методу обробки даних	24.11.2020	5	0	28.11.2020
Розробка програмної частини	29.11.2020	4	0	02.12.2020
Імплементация в CRM систему	01.12.2020	10	-2	10.12.2020
Тестування	11.12.2020	5	0	15.12.2020
Розробка звіту	16.12.2020	1	0	16.12.2020

Рисунок 3.1 – Декомпозиція робіт проекту (WBS)

Ошибка: источник перекрестной ссылки не найден відображає перелік робіт проекту задачі побудови методу пояснень у рекомендаційних системах. Із декомпозиції робіт проекту можна дізнатися перелік усіх робіт, а також їх тривалість. Майже усі роботи декомпоновані на менші роботи. Також з діаграми можна побачити порядок виконання робіт, а також оцінки на тривалість їх виконання. Друга колонка містить оптимістично оцінку, третя – вірогідну оцінку, четверта – песимістичну. Також на діаграмі зображені вірогідні дати завершення роботи. Якщо строки попередньої роботи змінюються, графік наступних робіт здвигається згідно до змін у попередній. Також, якщо деякі роботи можуть бути виконаними паралельно, декомпозиція робіт може це відображати. У випадку із даним проектом, роботи не можуть бути виконаними паралельно, усі вони формують критичний шлях. Діаграма Ганта із послідовністю виконання робіт та зображена на рисунку 3.2, що знаходиться нижче.

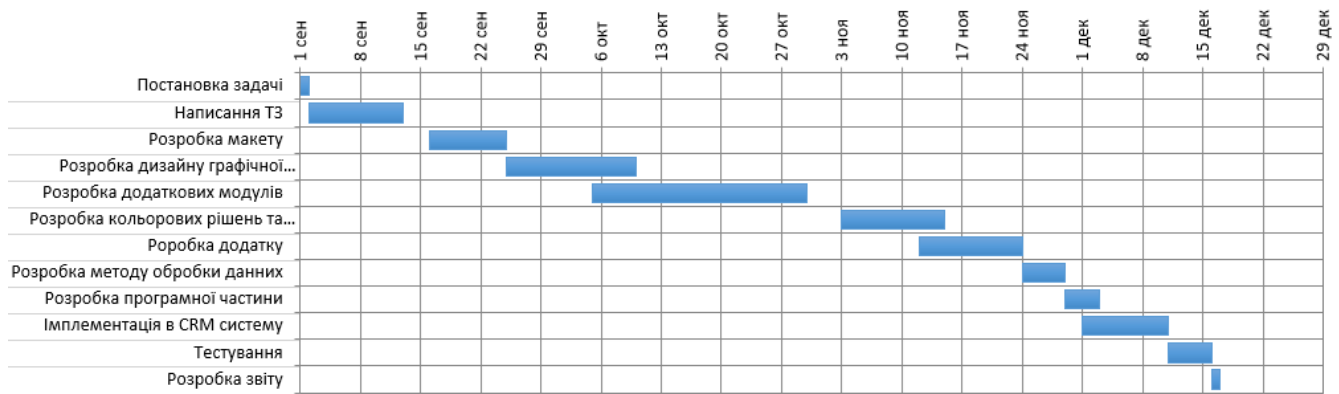


Рисунок 3.2 – Діаграма Ганта із критичним шляхом

містить діаграму Ганта. Представлене у вигляді діаграми Ганта розклад календарного робочого плану, дає інструментальну можливість якісніше оцінити ресурсні склади і взаємозв'язку робіт. В кінцевому рахунку, це зменшує число наступних змін в ході реалізації проектних заходів. А форма подання робить прив'язку завдань за часом більш наочною. Для цього і потрібна діаграма Ганта. У трактуванні РМІ поняття календарного плану збігається з поняттям розкладу проекту. Такий розклад створюється, щоб розрахувати і показати повну тривалість проекту як вираження термінів виконання проектних операцій. Послідовність ланок в ланцюжку процесу планування, виражена в розкладі, співвідноситься з ієрархічною структурою робіт, залученістю матеріальних, фінансових і людських ресурсів. Глибина декомпозиції завдань багато в чому залежить від можливості швидко знайти співробітників, матеріали і бюджетні кошти для виконання такого завдання. Контроль за вмістом, термінами і послідовністю робіт реалізуються за допомогою координуючого календарного плану. Діаграма Ганта показує розклад запланованих операцій по проектній задачі, але значущості, ресурсоемності та області дії робіт графік Ганта не демонструє.

Так як у проекті немає робіт, що можуть виконуватися паралельно, послідовність усіх робіт проекту формує критичний шлях. Найдовша послідовний ланцюг операцій в проекті називається критичним шляхом. Послідовність завдань, розташованих на такому шляху, в разі порушення тимчасових регламентів відсувають і дату завершення проекту в цілому. Крім того, критичний шлях проекту - це ще й один з ключових методів управління проектами. У західній управлінській практиці він носить назву Critical Path Method (CPM), а в радянській практиці управління метод критичного шляху (МКП) разом з аналогом системи PERT застосовувався в загальній системі методів мережевого планування і управління.

Рисунок 3.3 – Організаційна структура проекту (OBS)

містить зображення організаційної структури проекту. Необхідність розробки організаційної структури пояснюється тим, що для виконання проекту створюється команда проекту - новий тимчасовий робочий колектив, що складається з фахівців різних структурних підрозділів компаній з боку Виконавця і з боку Замовника. Як і для будь-якого нового колективу, для членів команди проекту необхідно визначити проектні ролі (тимчасові посади), функції, обов'язки, відповідальність, повноваження і правила взаємодії, а також організаційну схему, яка відображатиме відносини підпорядкованості. При цьому несуттєво, на який період часу буде створюватися команда проекту - на кілька місяців або на кілька років. Структура проекту визначається складністю, масштабністю розробки та впровадження ІС, кількістю і спеціалізацією членів команди проекту. В команду проекту можуть включатися спеціалісти, як на повну, так і на часткову зайнятість.

i	Назва	Затрати	Деталі	01 Вер 20				18 Лис 20				17 Жов 20				16 Гру 20			
				с	п	с	ч	с	ч	в	в	п	с	п	с	с	ч		
				Трудозатр.				Трудозатр.				Трудозатр.				Трудозатр.			
1	Програміст	160 ч	Трудозатр.			32ч	24ч	16ч	32ч	16ч				8ч	24ч	8ч			
			Превыш.																
	Написання ТЗ	24 ч	Трудозатр.			24ч													
			Превыш.																
	Робота з макетом	40 ч	Трудозатр.			8ч	24ч	8ч											
			Превыш.																
	Розробка графіки	16 ч	Трудозатр.					8ч	8ч										
			Превыш.																
	Розробка методик	40 ч	Трудозатр.							24ч	16ч								
			Превыш.																
Інтеграція	16 ч	Трудозатр.											8ч	8ч					
		Превыш.																	
Тестування	24 ч	Трудозатр.													16ч	8ч			
		Превыш.																	
2	Тестувальник	96 ч	Трудозатр.							16ч	24ч	24ч	32ч						
			Превыш.																
3	Менеджер проекту	72 ч	Трудозатр.	8ч	32ч			8ч							16ч	8ч			
			Превыш.																
	Постановка завдань	16 ч	Трудозатр.	8ч	8ч														
			Превыш.																
	Написання ТЗ	24 ч	Трудозатр.		24ч														
			Превыш.																
	Розробка дизайну	8 ч	Трудозатр.					8ч											
			Превыш.																
	Тестування	24 ч	Трудозатр.												16ч	8ч			
			Превыш.																

Рисунок 3.4 – Таблиця використання ресурсів (RBS)

містить опис усіх ресурсів, що використано на проекті. На сторінці Resource Graph (Графік ресурсів) відображається інформація про різні аспекти участі ресурсів у проекті: виконуваної ними роботи, завантаженні, можливості виконувати завдання крім тих, на які вони вже виділені. Зручність цієї таблиці полягає в тому, що з її допомогою можна порівнювати участь в проекті декількох ресурсів або ресурсу і групи ресурсів, і це робить її потужним аналітичним інструментом.

Група ресурсів відбирається на основі фільтру, і за замовчуванням в якості групи виступають всі ресурси проекту. Змінивши умову фільтрації, можна змінити і склад групи. Дані можна переглядати як для групи в цілому, так і для будь-якого окремого ресурсу всередині групи. При цьому дані про групу і окремого ресурсу можна відображати одночасно.

На таблиці використання ресурсів відображена завантаження у годинах зайнятості усіх задіяних у проект спеціалістів: а саме тестувальника, програміста та менеджера проекту.

Таблиця дає змогу краще зрозуміти та структурувати усі етапи розробки, затрати в часовому еквіваленті та повний аспект робіт розподілений серед учасників та спеціалістів проекту. Також до таблиці в подальшому можна додати вузьких спеціалістів в області дизайну та побудови UI/UX для покращення результатів конверсії від частини з якою буде працювати потенціальний клієнт. Обов'язки вузьких спеціалістів, наприклад в області дизайну UI/UX повинні бути сформовані Менеджером проекту та також додані до загальної таблиці використання ресурсів. Час затрачений на розробку правильного та гармонійного UI/UX також повинен бути відображений в загальній структурі таблиці використання ресурсів, тому що, планування роботи спеціалістів в вузькій предметній області може значно виділятися серед часу відведеного для інших робіт із-за складності роботи над певними елементами. Тож в задачі Менеджера проекту входить найбільш важлива частина планування та слідування чіткого плану роботи

Таблиця розділена на дві частини. У лівій частині розташована коротка інформація, в якій виводиться назва ресурсу і перелік робіт що виконуються цим ресурсом. Данні про дату виконання а також необхідний час відображається в правій частині.

4. ПРАКТИЧНЕ ВИКОРИСТАННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

4.1 Розробка програмного модулю формування пояснень в рекомендаційній системі

Розроблений програмний засіб являє собою веб-додаток на мові Java до Tilda CMS який базується на сервері інтернет-магазину і аналізуючи данні від рекомендаційної системи та користувача видає пояснення.

Обрання Java як мови розробки програмного додатку було не випадкове, тому що на сьогоднішній день ця мова є одною з найпрогресивніших платформ яка повністю задовольняє наші вимоги. Розглянемо переваги мови програмування Java при реалізації даного додатку:

- Мова програмування Java є мультиплатформенною, тому програмне забезпечення написане на ньому виконується на всіх основних апаратних платформах і може працювати в середовищі всіх основних операційних систем.

- Програмне забезпечення може виконуватися на будь-якому сервері. Єдиною вимогою є установка на даному сервері виконуючої системи Java, так званої Java Runtime Environment (JRE).

- Програмне забезпечення реалізоване на мові Java виконується на відокремленої віртуальній машині JVM і не взаємодіє безпосередньо з апаратними ресурсами сервера і програмними ресурсами операційної системи.

- Для роботи програмного забезпечення написаного на мові Java потрібен тільки установка Java Runtime Environment.

- Однією з сильних сторін віртуальної машини Java є її здатність роботи з декількома потоками. JVM оптимізована для великих багатоядерних машин, і вона без проблем може керувати великою кількістю потоків.

- Всі основні web-браузери мають підтримку для роботи з додатками написаними на мові Java.

При реалізації програмного додатку є певні завдання, для вирішення яких використання мови програмування Java є найбільш оптимальним:

- Побудова універсального діалогового інтерфейсу, який функціонує під управлінням різних операційних систем.
- Реалізація WEB-інтерфейсу додатків.
- Реалізація універсального інтерфейсу програмування додатків (API).
- Реалізація мультиплатформових методів збору і аналізу статистичної інформації.

Розглянемо, як використовується мова програмування Java при реалізації таких додатків на прикладах:

- при реалізації діалогового інтерфейсу мову програмування Java використовується у всіх перерахованих вище системах управління;
- для роботи з своїм API компанія Hitachi використовує мову JavaScript, а компанія HP в якості базової мови для свого API вибрала мову програмування Java;

Таким чином ми бачимо, що мова програмування Java знайшла дуже широке застосування при реалізації систем управління центрами обробки даних і повноцінного конкурента в цьому сегменті при створенні мультиплатформових додатків у нього поки немає.

Вхідні данні:

- записи про кількість проданого товару за проміжок часу;
- записи про рейтинг товарів;
- категорія товару;
- записи про користувачів ресурсу;
- записи про вибір користувачів ресурсу;

Частина яка відповідає за збір інформації про кількість проданого товару зображена на рисунку 4.1.

```

import java.util.Map;

public class search extends HttpServlet {

    protected void doPost(HttpServletRequest request,
        HttpServletResponse response) throws ServletException,
        IOException {
        String next_page = "/user/homepage.jsp";

        if (request.getParameter("action") != null) {
            String action = request.getParameter("action");
            if (action.equals("searchKeyword")) {
                // PostRedirectGet pattern
                String url = constructURL(request);
                response.sendRedirect(url + "&reallyActive
                    =true&page=0");
                return;
            }
        }
        /* if user has provided another action, then reroute him
        */
        RequestDispatcher view = request.getRequestDispatcher
            (next_page);
        view.forward(request, response);
    }

    protected void doGet(HttpServletRequest request,
        HttpServletResponse response) throws ServletException,
        IOException {
        String next_page = "/user/homepage.jsp";
        HttpSession session = request.getSession();

        if (request.getParameter("action") == null) {
            RequestDispatcher view = request.getRequestDispatcher
                (next_page);
            view.forward(request, response);
        }
    }
}

```

Рисунок 4.1 – Ініціалізація збору інформації про продажі

В даній частині коду ми можемо побачити що на мові Java було розроблено модуль який при отриманні запиту від рекомендаційної системи отримує команду `request.getRequestDispatcher` на збір даних про продажі певного товару по ключових словах у категорії. Тож, після запуску процесу збору додаток очікує відповіді від бази даних та вподальшому проводить аналіз отриманих даних за певний проміжок часу з урахуванням історії

пошуку зареєстрованого користувача в системі. Через запит `request.getSession()` додаток виділяє поточну сесію браузеру клієнта з системою та зберігає інформацію про виконані дії.

Частина яка відповідає за вивід максимального рейтингу в категорії зображена на рисунку 4.2.

```

RequestDispatcher view = request.getRequestDispatcher
(next_page);
view.forward(request, response);
}
SearchService searchService = new SearchService();
int page = 0;
if (request.getParameterMap().containsKey("page")) {
    page = Integer.parseInt(request.getParameter("page"
));
}

/* min and max values are assigned from here in order to
use some old code
* this is used on searchRating with a Restrictions
.between
* */
searchService.setMinPrice(0d);
if (request.getParameterMap().containsKey("minRating")) {
    if (!request.getParameter("minRating").equals("")) {
        searchService.setMinPrice(Double.parseDouble
(request.getParameter("minRating")));
    }
}
searchService.setMaxPrice(Double.MAX_VALUE);
if (request.getParameterMap().containsKey("maxRating")) {
    if (!request.getParameter("maxRating").equals("")) {
        searchService.setMaxPrice(Double.parseDouble
(request.getParameter("maxRating")));
    }
}

switch (request.getParameter("action")) {
    case "advancedSearch":
        /* gather all categories to display on jsp */
        CategoryService categoryService = new
        CategoryService();
        List categoryLst = categoryService
        .getAllCategories();

```

Рисунок 4.2 – Вивід максимального рейтингу категорій

Зображена частина програмного коду на мові розробки Java відповідає за отримання даних про останні рейтинги товарів в категорії за ключовими словами за період часу. Команда `request.getParameter("maxRating")` надає запит до бази даних рейтингу товарів в категорії за ключовими словами та отримавши відповідь сортує та формує оцінку яка буде використана для виводу відсортованого рейтингу товарів у категорії за ключовими словами. Швидкість та зручність роботи додатку обумовлена простотою написаного програмного коду та широкою підтримкою його імплементування у сучасних веб-системах.

На етапі обробки, наша рекомендаційна система отримавши всю необхідну інформацію, може спрогнозувати вподобання користувача та

сформуванати список товарів які можуть бути використані для запропонування користувачеві як альтернатива або доповнення. Після попередньо сформованої рекомендації вступає в роботу наш веб-додаток і основуючись на інформації з бази даних про користувача проводить пояснення для рекомендаційної системи. Структура програмного додатку зображена на рисунку 4.3.

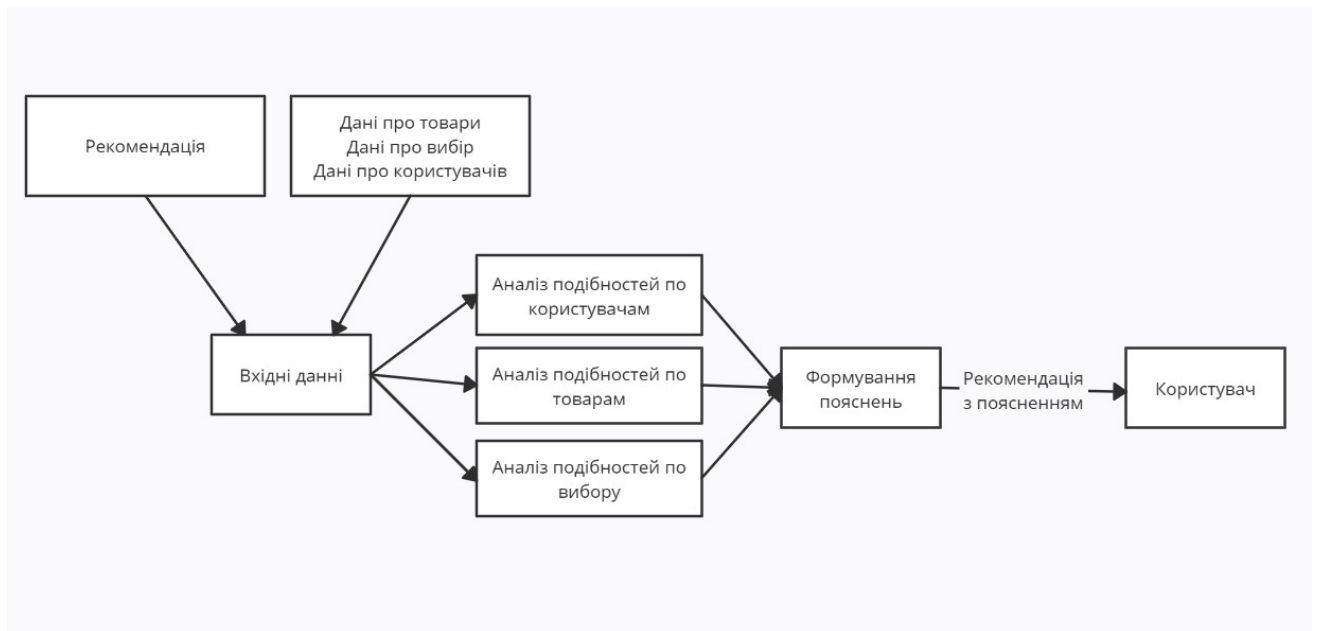


Рисунок 4.3 – Структура програмного додатку

На рисунку 4.4 зображено алгоритм роботи програмного додатку надання пояснень в проектах рекомендаційних систем. Після початку роботи алгоритму є необхідність введення первинних значень запиту користувача, після чого йде етап обробки цих значень та формування рейтингу оцінок товару за проміжок часу. Після аналізу даних стартує процес формування рекомендацій то певного запиту користувача. На наступному кроці вступає в роботу процес надання пояснення до рекомендації згідно з методикою роботи додатку. Після того, як користувач отримав рекомендації та пояснення до них, настає момент очікування дії від користувача.

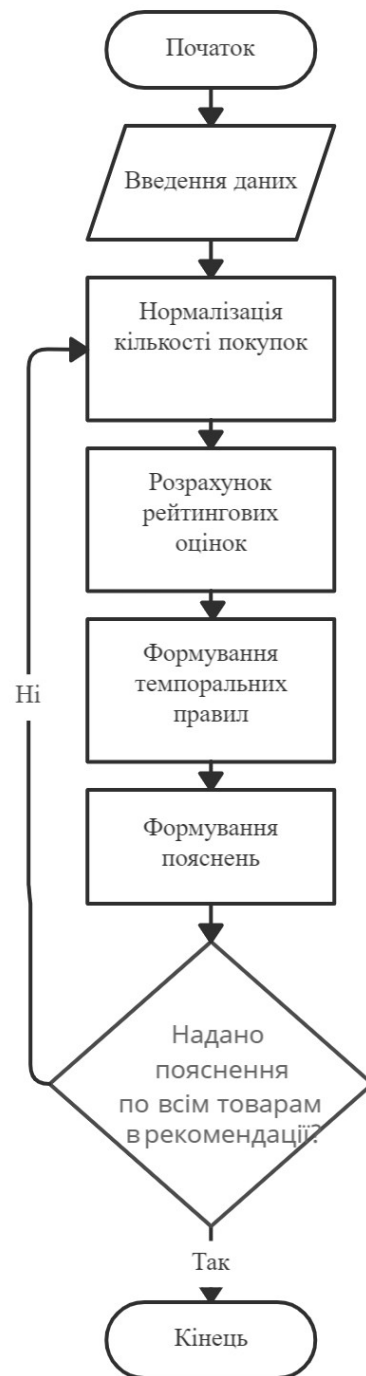


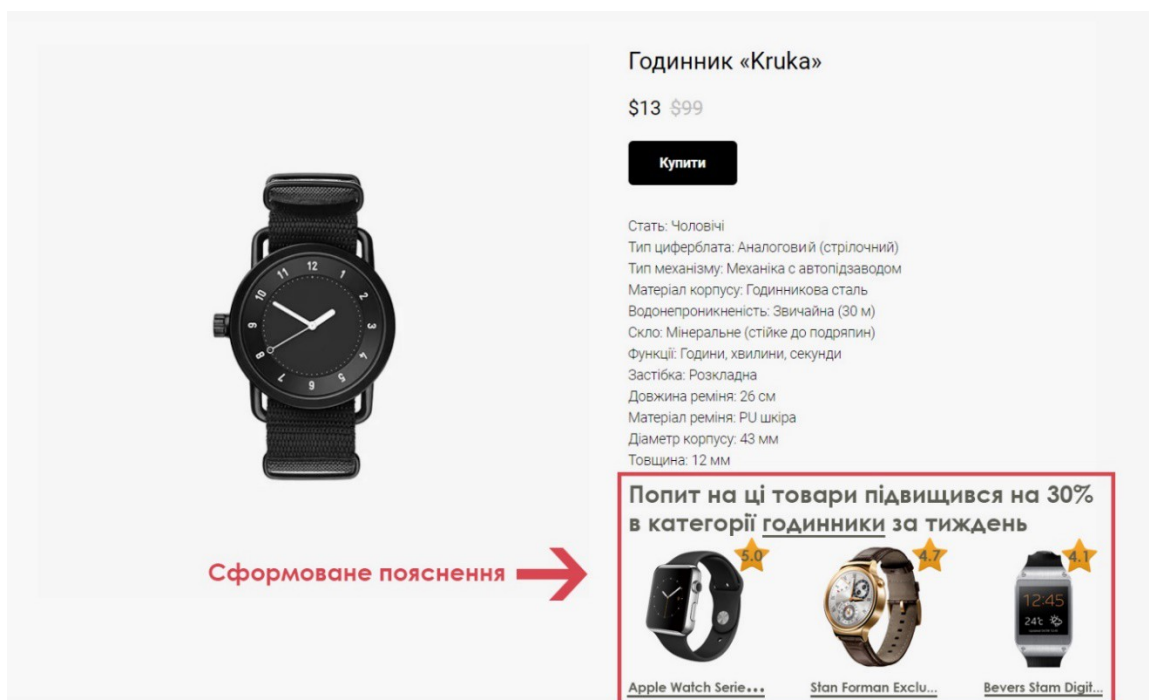
Рисунок 4.4 – Алгоритм роботи додатку

Після взаємодії користувача з модулем рекомендації та пояснень йде крок запам'ятовування вибору користувача у його профілі та алгоритм завершує свою роботу. У разі відмови від взаємодії з користувачем алгоритм повторює процес формування та надання пояснень.

4.2 Експериментальна перевірка удосконаленого методу формування пояснень

В першому випадку ми отримали данні про рейтинг товарів із категорії "Годинники" який був створений завдяки оцінкам попередніх користувачів. Очевидно, що нас цікавлять товари з найбільшим рейтингом за час який цей товар існує. Тобто, додаток сортує `rating_id` рекомендованих товарів, і завдяки дизайну нашого екрану пояснень може виділити 3 лістинга товарів та вивести їх на сторінку. Виведена на екран рекомендація супроводжується поясненням по неї "Попит на ці товари підвищився на 30% в категорії годинники за тиждень". У результаті, потенціальний покупець при виборі годинника може перевести увагу на позиції які вже були придбані та отримали найкращі відгуки від користувачів та зробити свій вибір на користь перевірених товарів.

На рисунку 4.5 зображено приклад екрану пояснень з темпоральною залежністю.



Годинник «Kruka»
\$13 \$99

Купити

Стать: Чоловічі
Тип циферблата: Аналоговий (стрілочний)
Тип механізму: Механіка с автопідзаходом
Матеріал корпусу: Годинникова сталь
Водонепроникність: Звичайна (30 м)
Скло: Мінеральне (стійке до подряпин)
Функції: Години, хвилини, секунди
Застібка: Розкладна
Довжина реміня: 26 см
Матеріал реміня: PU шкіра
Діаметр корпусу: 43 мм
Товщина: 12 мм

Сформоване пояснення →

Попит на ці товари підвищився на 30% в категорії годинники за тиждень




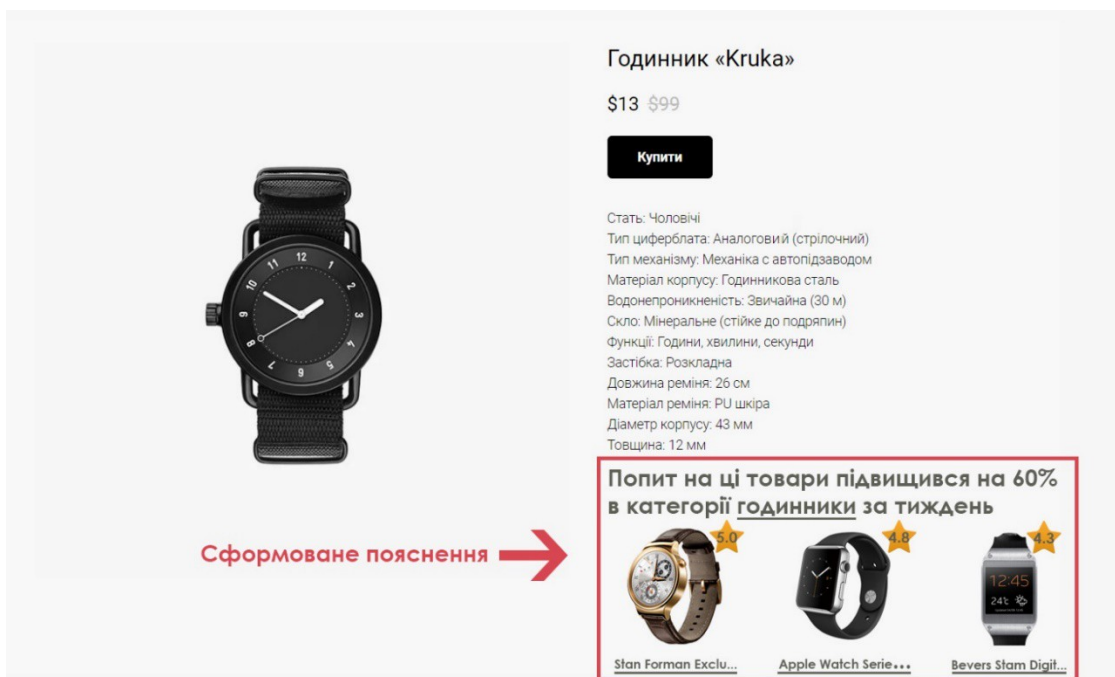
 5.0	 4.7	 4.1
Apple Watch Serie...	Stan Forman Exclu...	Beverly Stam Digit...

Рисунок 4.5 – Приклад екрану пояснення з темпоральною залежністю

В другому випадку ми отримали данні про рейтинг товарів із категорії "Годинники" який був розрахований за наступний період часу. На рисунку 4.6 зображено пояснення до рекомендації товарів у категорії яке інформує що за період часу попит і рейтинги товарів які були сформовані на попередньому тижні змінилися та зросли на 60%. Це може бути обумовлено сезонністю формування пояснень, наприклад напередодні свят таких як Новий Рік або День Програміста. Тим самим користувач проінформований про різку зміну попиту на данні товари та може відслідити рейтинги оцінок товарів від користувачів які вже попередньо купили або замовили цей товар. Як результат, надання пояснення з уточненням та включенням темпоральної динаміки, може позитивно впливати на вибір кінцевого користувача, а точне пояснення до рекомендації стимулює інтерес клієнта та підвищує шанси на те, що клієнт зробить покупку на користь представлених товарів.



Годинник «Kruka»
\$13 \$99

Купити

Стать: Чоловічі
Тип циферблата: Аналоговий (стрілочний)
Тип механізму: Механіка с автопідзаходом
Матеріал корпусу: Годинникова сталь
Водонепроникність: Звичайна (30 м)
Скло: Мінеральне (стійке до подряпин)
Функції: Години, хвилини, секунди
Застібка: Розкладна
Довжина реміня: 26 см
Матеріал реміня: PU шкіра
Діаметр корпусу: 43 мм
Товщина: 12 мм

Сформоване пояснення →

Попит на ці товари підвищився на 60% в категорії годинники за тиждень

Товар	Рейтинг
Stan Forman Exclu...	5.0
Apple Watch Serie...	4.8
Bevers Stam Digif...	4.3

Рисунок 4.6 – Приклад екрану пояснення за умови зміни попиту

На рисунку 4.7 надається зручний вивід усієї інформації щодо надання пояснень додатком у проектах рекомендаційних систем. Ми імплементували даний спосіб в нашому додатку до Tilda CMS та отримали певні результати. На даному графіку показано кількість замовлень в категорії годинники без нашого додатку та порівняно з результатами включення нашого додатку до рекомендаційної системи.

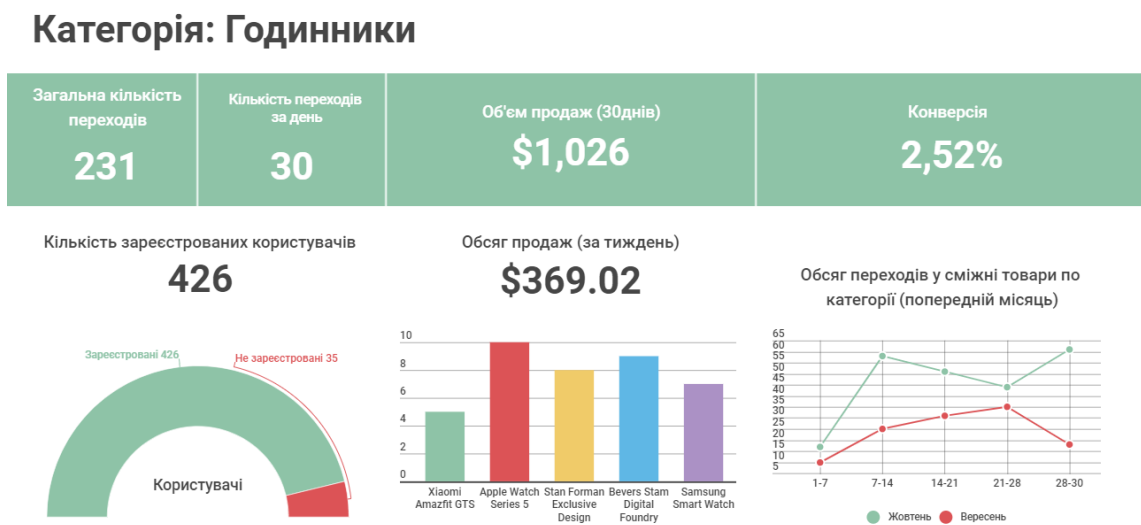


Рисунок 4.7 – Без пояснення рекомендацій

. У такій ситуації доцільно враховувати темпоральних динаміку переваг користувачів. Остання задається набором темпоральність правил, які відображають зміни продажів цільового товару для двох незв'язаних інтервалів часу, наприклад для двох різних днів.

Метод включає в себе етапи формування невиважених темпоральних правил, розрахунку ваг темпоральних правил, формування пояснень у вигляді суми ваг правил по кожному предмету.

Даний метод відрізняється тим, що представляє пояснення у вигляді єдиного показника, який об'єднує зміну продажів по кожному рекомендованому товару за заданий період часу.

Пропонований метод дозволяє підвищити довіру користувача до рекомендацій в силу очевидності використовуваного показника. Це дає можливість підвищити ефективність роботи рекомендаційної системи.

Категорія: Годинники

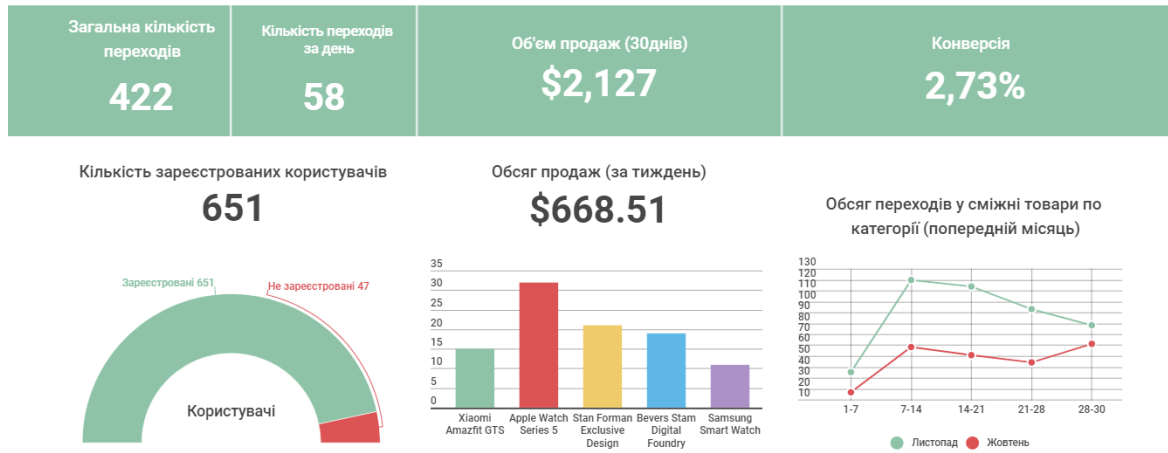


Рисунок 4.8 – З поясненням рекомендації

Як ми можемо бачити, результати показують, що кількість замовлень підвищилась у результаті імплементування пояснень на відміну ввід версії без пояснення. На рисунку 4.8 показано результати імплементування пояснення рекомендації. Користувач краще став розуміти цілі та мету з якою представлена рекомендація до його поточного вибору. Так як кількість замовлень підвищилась, з даного графіку ми бачимо що якість рекомендацій зросла майже вдвічі. За цим слідує кількість переходів які зробили користувачі за допомогою пояснення виведеної рекомендації, тобто кількість зацікавлених та потенціальних користувачів зробили фінальний вибір у сторону найбільш оцінених та перевірених позицій. З цього можна зробити висновок що пояснення до рекомендації спрацювало та привело до кінцевого позитивного результату для користувача.

ВИСНОВКИ

В магістерській атестаційній роботі досліджено проблему формування пояснень з урахуванням змін вподобань користувачів з часом.

Проведений аналіз показав, що використання пояснень дає можливість підвищити довіру користувачів до отриманих рекомендацій та збільшити конверсію в системі електронної комерції.

Дослідження процесу формування пояснень дало можливість зробити висновок про необхідність враховувати зміну інтересів користувачів з часом.

В роботі удосконалено метод надання пояснень у проектах рекомендаційних систем, шляхом додавання урахування рейтингів та змін темпоральної динаміки.

Для реалізації розробленого додатку був спланований проект, що включає у себе розробника, тестувальника та проектного менеджера. Результатом успішного виконання проекту став програмний додаток до CMS Tilda, що виконує функцію надання пояснень до рекомендаційної видачі. Реалізовано модуль формування пояснень який виконує функції аналізу вхідних даних про користувачів, товари та дані про продажі. Додаток на виході формує пояснення у вигляді переліку товарів, упорядкованих за тимчасовою динамікою за умови, що ця динаміка позитивна

Виконано експериментальну перевірку удосконаленого методу, та роботу додатку.

Експериментальна перевірка розробленого додатку показала збільшення конверсії з 2.52% до 2.73%

Результати магістерської роботи представлені у матеріалах 24-го міжнародного молодіжного форуму «Радіоелектроніка та молодь у XXI столітті» 2020р. опубліковані тези доповіді «Дослідження методів формування пояснень у проектах рекомендаційних систем».

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Левикин В.М., Міхнов Д.К., Саєнко В.І., Євланов М.В., Міхнова А.В., Керносов А.М., Методичні вказівки щодо розробки та оформлення магістерської атестаційної роботи за спеціальністю 8.05010101 – *Інформаційні управляючі системи та технології. Освітньо-кваліфікаційний рівень – магістр*. Харків. ХНУРЕ, 2012. – 28 С.
2. Чалый С. Ф. Исследование методов анализа социальных сетей для определения групп пользователей программного продукта. Харьков. :ХНУРЭ, 2013. – № 165. – С. 107–111
3. Чалый С. Ф., Прибыльнова И. Б., Темпорально-объектные модели события и процесса в задачах моделирования рассуждений на основе прецедентов. *Уральский научный вестник*. - 2015. -№19 (150). - С. 71-75.
4. Чалый С. Ф., Левикин И.В., Метод адаптивного процесного управління на основі прецедентного підходу. *Наукоємні технології* № 4 (32), 2016 – С. 410-414.
5. Чалый С.Ф., Прибыльнова И. Б., Ситуационный поход к представлению темпоральных знаний прецедентов. *Вісник Національного технічного університету "ХПИ". Серія : Системний аналіз, управління та інформаційні технології*. Харків - 2016. - № 45. - С. 70-73.
6. Чалый С.Ф., Лещинський В.О., Лещинська І.О. Моделювання контексту в рекомендаційних системах. *Науковий журнал «Проблеми інформаційних технологій»*, 2018, №. 1(023). С. 21-26.
7. Чалый С.Ф., Прибыльнова И.Б., Ситуаційна модель користувачького вибору в рекомендаційні системі. *Системи управління, навігації та зв'язку*. – 2019. – Вип. 2(54). – С.159-163.
8. Чалый С.Ф., Лещинський В.О., Лещинська І.О. Моделювання пояснень щодо рекомендованого переліку об'єктів з урахуванням

темпорального аспекту вибору користувача. *Системи управління, навігації та зв'язку*, 2019. Том 6 № 58. С. 97-101.

9. Чалий С.Ф., Лещинський В.О., Лещинська І.О. Доповнення вхідних даних рекомендаційної системи в ситуації циклічного холодного старту з використанням темпоральних обмежень типу «NEXT». *Системи управління, навігації та зв'язку*, 2019. Вип. 4(56). С. 105-109.

10. Чалий С.Ф., Лещинський В.О., Лещинська І.О. Модель інтерфейсу пояснень з темпоральними параметрами в рекомендаційній системі. *Системи управління, навігації та зв'язку*. 2020. Вип. 2(60). С. 105-109.

11. Чалий С.Ф., Лещинський В.О., Лещинська І.О., Багаторівнева персоналізація пояснень в рекомендаційних системах. *Сучасні інформаційні системи*. 2020. Том 4, № 2. С. 170-175.

12. Чалий С.Ф., Лещинський В.О., Лещинська І.О., Концепція формування пояснень в рекомендаційних системах за принципом білого ящика. *Збірник наукових праць*. – Полтава: ПНТУ, 2019. – Т. 3 (55). – С. 156-160.

13. Gunes I., Kaleli C., Bilge A., Polat H. Shilling attacks against recommender systems: a comprehensive survey. *Artificial Intelligence Review*, 2014. 42, 767–799.

14. Chala O., Novikova L., Chernyshova L., Method for detecting shilling attacks in e-commerce systems using weighted temporal rules. *EUREKA: Physics and Engineering*. 2019. Vol. 5, 29-36. DOI: 10.21303/2461-4262.2019.00983

15. Tintarev N., Masthoff J. A survey of explanations in recommender systems. *23rd International Conference on Data Engineering Workshop*, 2007, 801– 810.

16. Tintarev N., Masthoff J. Designing and evaluating explanations for recommender systems, in Ricci, F. Rokach, L. Shapira, B. and Kantor, P. (Eds.), *Recommender systems handbook*. Springer, Dordrecht, 2010, 479–510.

17. Tintarev N., Masthoff J. Evaluating the effectiveness of explanations for recommender systems. *User Model User-Adap Inter*, Vol 22, 2012, 399– 439.

18. Chalyi S., Levykin I., Petrychenko A. and Bogatov I.. Causality-based model checking in business process management tasks. In Proc.: *IEEE 9th International Conference on Dependable Systems, Services and Technologies*. Ukraine, Kyiv, 2018, 478 – 483.

19. Wu Z., Cao J., Mao B. Wang Y. Semi-SAD: applying semi-supervised learning to shilling attack detection. Proceedings of the fifth ACM conference on Recommender systems, 2011, 289–292.

20. Elahi M., Ricci F., Rubens N. A survey of active learning in collaborative filtering recommender systems. *Computer Science Review*. 2016, 29-50.

21. Xiang L., Yuan Q. Temporal Recommendation on Graphs via Long-and Short-term Prefence Fusion. *KDD'10 of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2010,723-732.

22. Chalyi S., Pribylnova I. The method of constructing recommendations online on the temporal dynamics of user interests using multilayer graph. *EUREKA: Physics and Engineering*. 2019. Vol. 3. P. 13-19.

23. Zhu Y., Lin J., He S., Wang B., Guan Z., Liu H., and Cai D. Addressing the item cold-start problem by attribute-driven active learning,”. *arXiv* 2018, Vol. 3. P. 12-19.

24. Braunhofer M. Hybrid Solution of the Cold-Start Problem in Context-Aware Recommender Systems. *UMAP 2014: User Modeling, Adaptation, and Personalization*. 2014, 484-489.

25. Son Le Hoang. Dealing with the new user cold-start problem in recommender systems: A comparative review. *Information Systems*, 58, 2016, 87-26. Aggarwal C. Recommender Systems. *The Textbook*. New York, 2017, 498 p.

27. Lee, D., Park, S. E., Kahng, M., Lee, S. and Lee, S. Exploiting contextual information from event logs for personalized recommendation. *In Computer and Information Science*. 2010, pp. 121–139.

28. Chalyi S., Leshchynskiy V. Knowledge Representation in the Recommendation System Based on the White Box Principle. *Сучасні інформаційні системи*. 2019. Т. 3, № 3 С. 82-86.

29. Lee, J. S. and Lee, J. C. Music for my mood: A music recommendation system based on context reasoning. *In Smart sensing and context*. 2006, pp. 190–203.

30. Chalyi S., Leshchynskiy V. Temporal representation of causality in the construction of explanations in intelligent systems. *Advanced information systems*. VOL 4, NO 3.2020, P. 113-117.

31. Chalyi S., Leshchynskiy V., Leshchynska I. Detailing explanations in the recommender system based on matching temporal knowledge. *Eastern-European Journal of Enterprise Technologies* . 2020, Vol 4, No 2 (106). P. 6-13

32. Elahi M., Ricci F., Rubens N. A survey of active learning in collaborative filtering recommender systems. *Computer Science Review*. 2016, 29-50.

33. Chalyi S., Levykin I., Biziuk A., Vovk A., Bogatov I. Development of the technology for changing the sequence of access to shared resources of business processes for process management support. *Eastern-European Journal of Enterprise Technologies*. 2020. P. 22-29.

34. Claypool, M., P. Le, M. Wased and D. Brown, Implicit interest indicators. *Proceeding of International Conference on Intelligent User Interfaces*, 2001, pp: 33-40.

35. Kalynychenko, O., Chalyi, S., Bodyanskiy, Y., Golian, V., Golian, N. Implementation of search mechanism for implicit dependences in process mining. *IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS)*. 2013, pp 25-28.

36. Herlocker, J.L., J.A. Konstan, A. Borchers and J. Riedl, An algorithmic framework for performing collaborative filtering. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. Berkeley, California, USA, 1999, pp: 230-237.

37. Guo, G. Improving the performance of recommender systems by alleviating the data sparsity and cold start problems. *In Proceedings of the Twenty-Third international joint conference on Artificial Intelligence, AAAI Press, 2013*, pp. 3217-3218.

38. Lops, P., Degenmis, M. and Semeraro, G. Improving social filtering techniques through WordNet-Based user profiles. *User Modeling 2007. Springer Berlin Heidelberg, 2007*, pp. 268-277.

39. Lika, B., Kolomvatsos, K. and Hadjiefthymiades, S. Facing the cold start problem in recommender systems. *Expert Systems with Applications. Vol. 41, 2014*, pp. 2065–2073.

40. Bobadilla, J., Ortega, F., Hernando, A. and Gutiérrez, Recommender systems survey. *Knowledge-Based Systems, Vol. 46, 2013*, pp. 109–132.

41. 22-й Міжнародний молодіжний форум «Радіоелектроніка та модуль у XXI столітті». Зб. *Матеріалів форуму. Т.6. – Харків: ХНУРЕ. 2018 – 496с. – 131с.*