Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)
Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Розробка інтелектуальної інформаційної системи рекомендацій пива на основі аналізу користувацьких відгуків та методів машинного навчання
(тема)

Виконав:
здобувач четвертого року навчання,
групи ІТШІ-22-4
Денис ГЛАДКОВ
(власне ім'я, прізвище)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)
Тип програми освітньо-професійна
Освітня програма Штучний інтелект
(повна назва освітньої програми)
Керівник ас. Максим ПОЛІЇТ
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ _____ Лариса ЧАЛА
(підпис) (власне ім'я, прізвище)

2026 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Штучного інтелекту

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Штучний інтелект
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Гладкову Денису Максимовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка інтелектуальної інформаційної системи рекомендацій пива на основі аналізу користувацьких відгуків та методів машинного навчання

затверджена наказом університету від 5 червня 20 26 р. № 488Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 9 липня 20 26 р.

3. Вихідні дані до роботи Вихідні дані до роботи: наукові публікації та дослідження у сфері рекомендаційних систем, машинного навчання та обробки природної мови; датасет користувацьких оцінок і відгуків про пиво; методи колаборативної фільтрації та аналізу тональності текстів; засоби розробки Python, FastAPI, PostgreSQL, бібліотеки Pandas, NumPy, Scikit-learn, TensorFlow (або PyTorch, якщо використовувався), а також нормативна та довідкова література за тематикою роботи.

4. Перелік питань, що потрібно опрацювати в роботі

1) Аналіз предметної галузі та постановка задачі дослідження

2) Аналіз методів та підходів до побудови рекомендаційних систем на основі текстових відгуків

3) Розробка інтелектуальної інформаційної системи рекомендацій пива

4) Експериментальне дослідження та аналіз результатів

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	08.06.2026	виконано
2	Аналіз предметної області та постановка задачі	08.06.2026–10.06.2026	виконано
3	Збір та аналіз літературних джерел	11.06.2026–12.06.2026	виконано
4	Дослідження існуючих рекомендаційних систем	12.06.2026–14.06.2026	виконано
	та методів машинного навчання		
5	Аналіз та підготовка набору даних	15.06.2026	виконано
	користувацьких відгуків		
6	Розробка математичної моделі та алгоритмів	15.06.2026–18.06.2026	виконано
	рекомендацій		
7	Проєктування архітектури інформаційної	19.06.2026–20.06.2026	виконано
	системи		
8	Реалізація серверної частини та бази даних	22.06.2026–23.06.2026	виконано
9	Реалізація алгоритмів машинного навчання	24.06.2026–28.06.2026	виконано
10	Тестування та налагодження системи	29.06.2026	виконано
11	Аналіз результатів та оцінка ефективності	30.06.2026	виконано
12	Оформлення пояснювальної записки	01.07.2026–02.07.2026	виконано
13	Захист перед ЕК	09.07.2026	виконано

Дата видачі завдання 8 червня 2026 р.

Здобувач _____
(підпис)

Керівник роботи _____ ас. Максим ПОЛІЇТ
(підпис) (посада, власне ім’я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 91 с., 45 рис., 3 табл., 1 дод., 32 джерела.

ГІБРИДНА МОДЕЛЬ, КОЛАБОРАТИВНА ФІЛЬТРАЦІЯ, КОНТЕНТНО-ОРІЄНТОВАНА ФІЛЬТРАЦІЯ, МАШИННЕ НАВЧАННЯ, ОБРОБКА ПРИРОДНОЇ МОВИ, ПИВО, РЕКОМЕНДАЦІЙНА СИСТЕМА, СЕНТИМЕНТ-АНАЛІЗ, BERT, NLP.

Об’єкт дослідження – процес формування персоналізованих рекомендацій пива на основі користувацьких відгуків та рейтингових даних.

Предмет дослідження – методи машинного навчання, обробки природної мови та рекомендаційні алгоритми для аналізу текстових відгуків і формування рекомендацій пива.

Мета роботи – розроблення інтелектуальної інформаційної системи рекомендацій пива, що забезпечує формування персоналізованих рекомендацій на основі аналізу текстових відгуків користувачів та методів машинного навчання.

Методи дослідження – аналіз предметної галузі, методи колаборативної та контентно-орієнтованої фільтрації, обробка природної мови, сентимент-аналіз, векторизація тексту, глибоке навчання, статистичний аналіз та оцінювання якості рекомендаційних систем.

У роботі розроблено інтелектуальну інформаційну систему рекомендацій пива на основі аналізу користувацьких відгуків та методів машинного навчання. Досліджено сучасні підходи до побудови рекомендаційних систем і методи обробки природної мови. Реалізовано механізми аналізу текстових відгуків та формування персоналізованих рекомендацій. Практичне значення роботи полягає у створенні програмного прототипу для використання в сервісах рекомендації напоїв та електронній комерції.

ABSTRACT

Bachelor’s thesis contains: 91 pp., 45 fig., 2 tabl., 1 ann., 32 references.

BEER, BERT, COLLABORATIVE FILTERING, CONTENT-BASED FILTERING, HYBRID MODEL, MACHINE LEARNING, NATURAL LANGUAGE PROCESSING, NLP, RECOMMENDER SYSTEM, SENTIMENT ANALYSIS.

The object of research is the process of generating personalized beer recommendations based on user reviews and rating data.

The subject of research is machine learning methods, natural language processing techniques, and recommendation algorithms for analyzing textual reviews and generating beer recommendations.

The purpose of this work is to develop an intelligent beer recommendation information system that provides personalized recommendations based on the analysis of user reviews and machine learning methods.

Research methods include domain analysis, collaborative and content-based filtering methods, natural language processing, sentiment analysis, text vectorization, deep learning, statistical analysis, and evaluation of recommender system performance.

This thesis presents the development of an intelligent beer recommendation information system based on user review analysis and machine learning methods. Modern approaches to recommendation system design and natural language processing techniques were studied. Mechanisms for analyzing textual reviews and generating personalized recommendations were implemented. The practical significance of the work lies in the development of a software prototype that can be used in beverage recommendation services and e-commerce platforms to improve product selection quality.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів 8
Вступ 10
1 Аналіз предметної галузі та постановка задачі дослідження 11
1.1 Світовий ринок пива та поведінка споживача 11
1.2 Огляд існуючих підходів до рекомендацій напоїв на основі відгуків 14
1.3 Постановка задачі дослідження 25
2 Аналіз методів та підходів до побудови рекомендаційних систем на основі текстових відгуків 27
2.1 Методи колаборативної фільтрації 27
2.2 Контентно-орієнтовані методи та обробка природної мови для аналізу текстів відгуків 32
2.3 Гібридні рекомендаційні системи та моделі на основі глибокого навчання 36
2.4 Метрики оцінювання якості рекомендаційних систем 40
3 Розробка інтелектуальної інформаційної системи рекомендацій пива 45
3.1 Технологічний стек та інструментарій розроблення 45
3.2 Підготовка набору даних 49
3.3 Методи попередньої обробки відгуків та формування ознак 54
3.4 Побудова гібридної рекомендаційної моделі 58
3.5 Архітектура системи та програмна реалізація 60
4 Експериментальне дослідження та аналіз результатів 66
4.1 Методика експерименту 66
4.2 Результати оцінювання з рівномірними вагами 69
4.3 Оптимізація ваг гібридної формули 71
4.4 Фінальні результати на тестових даних 73
4.5 Аналіз якості рекомендацій та демонстрація роботи системи 74
Висновки 84

Перелік джерел посилання 86
Додаток А Відомість кваліфікаційної роботи 91

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ШІ – штучний інтелект;

ABSA – Aspect-Based Sentiment Analysis – аспектно-орієнтований аналіз тональності;

AI – Artificial Intelligence – штучний інтелект;

ALS – Alternating Least Squares – метод найменших квадратів із чергуванням;

BERT – Bidirectional Encoder Representations from Transformers – двонапрямлені представлення енкодера на основі трансформерів;

BJCP – Beer Judge Certification Program – програма сертифікації пивних суддів;

BoW – Bag-of-Words – модель «мішок слів»;

CBOW – Continuous Bag-of-Words – неперервна модель «мішок слів»;

CF – Collaborative Filtering – колаборативна фільтрація;

CIN – Compressed Interaction Network – мережа стиснених взаємодій;

DeepFM – Deep Factorization Machine – глибока факторизаційна машина;

FM – Factorization Machines – факторизаційні машини;

IBU – International Bitterness Units – міжнародна шкала гіркоти пива;

MAE – Mean Absolute Error – середня абсолютна похибка;

MF – Matrix Factorization – матрична факторизація;

MLP – Multilayer Perceptron – багатошаровий перцептрон;

MSE – Mean Squared Error – середньоквадратична похибка;

NCF – Neural Collaborative Filtering – нейронна колаборативна фільтрація;

NLP – Natural Language Processing – обробка природної мови;

NMF – Non-negative Matrix Factorization – невід’ємна матрична факторизація;

RMSE – Root Mean Squared Error – корінь із середньоквадратичної похибки;

SGD – Stochastic Gradient Descent – стохастичний градієнтний спуск;

SIACF – Side-Informed Attention Collaborative Filtering –колаборативна фільтрація з механізмом уваги та додатковими ознаками;

SRM – Standard Reference Method – стандартна шкала кольору пива;

SVD – Singular Value Decomposition – сингулярний розклад матриці;

SVD++ – розширена модель сингулярного розкладу з урахуванням неявного зворотного зв’язку;

TF-IDF – Term Frequency – Inverse Document Frequency – частота терміна / обернена частота документа;

VADER – Valence Aware Dictionary and sEntiment Reasoner – лексиконний алгоритм аналізу тональності;

Word2Vec – модель векторного представлення слів.

ВСТУП

У сучасних умовах стрімкого розвитку електронної комерції та цифрових сервісів користувачі стикаються з проблемою вибору товарів серед великої кількості доступних варіантів. Для розв’язання цієї проблеми широко застосовуються рекомендаційні системи, які аналізують поведінку користувачів, їхні вподобання та характеристики об’єктів, пропонуючи найбільш релевантні рекомендації.

Однією з галузей, де рекомендаційні системи можуть забезпечити суттєве покращення користувацького досвіду, є ринок крафтового та промислового пива. Сучасні онлайн-платформи містять тисячі найменувань пивної продукції, що відрізняються стилем, смаковими характеристиками, рівнем гіркоти, вмістом алкоголю та іншими параметрами. Велика кількість доступних варіантів ускладнює процес вибору для користувачів, особливо для тих, хто не має достатнього досвіду у сфері пивної культури.

Традиційні рекомендаційні системи часто базуються виключно на числових оцінках користувачів або інформації про взаємодію з об’єктами. Проте значна частина корисної інформації міститься у текстових відгуках, де користувачі описують власні враження від продукту, виділяють його переваги та недоліки, оцінюють окремі характеристики смаку, аромату та зовнішнього вигляду. Використання методів обробки природної мови та машинного навчання дає можливість автоматично аналізувати такі відгуки та враховувати отримані результати під час формування рекомендацій.

Актуальність роботи обумовлена необхідністю підвищення точності та персоналізації рекомендацій шляхом поєднання методів колаборативної фільтрації з аналізом текстових відгуків користувачів. Використання сучасних алгоритмів машинного навчання дозволяє враховувати не лише явні оцінки користувачів, а й приховані вподобання, що містяться у текстових даних. Це сприяє формуванню більш релевантних рекомендацій та покращенню якості взаємодії користувача з інформаційною системою.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

1.1 Світовий ринок пива та поведінка споживача

Світовий ринок пива є одним із найбільших сегментів харчової промисловості. За оцінками Grand View Research, у 2024 році обсяг світового ринку пива становив 839,31 млрд доларів США, і прогнозується його зростання до 1 248,3 млрд доларів США до 2030 року (рисунок 1.1) при середньорічному темпі зростання 6,8% [1].

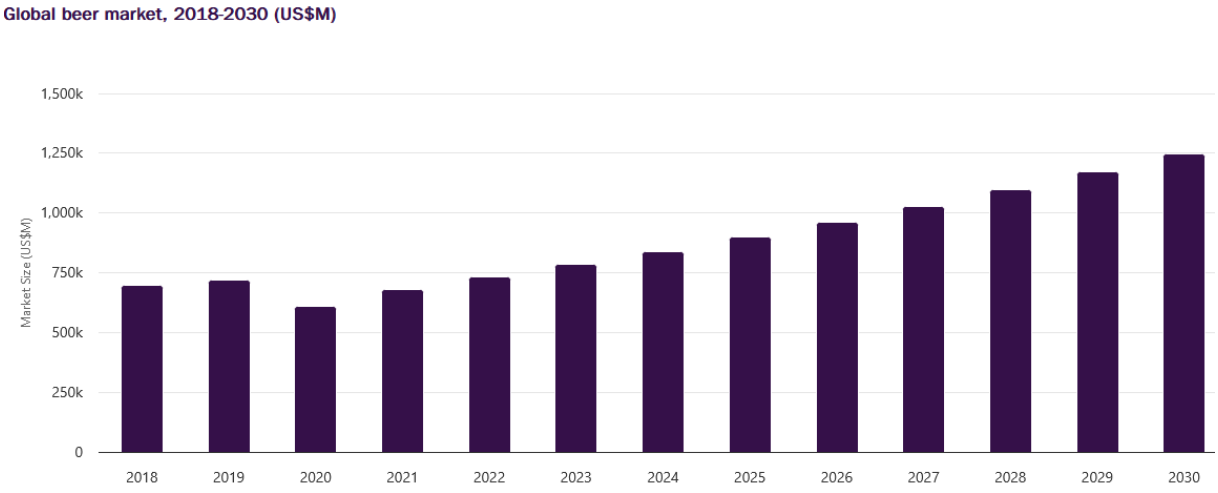
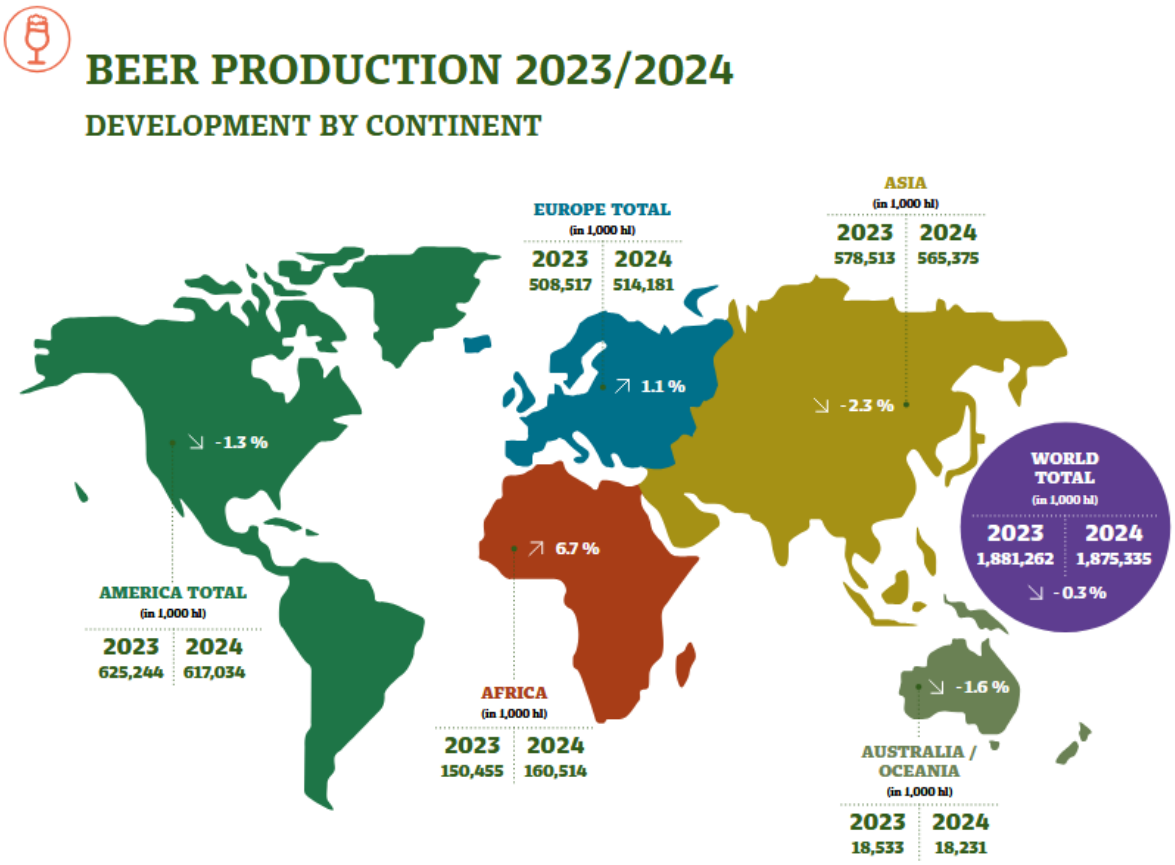


Рисунок 1.1 – Обсяг світового ринку пива у 2018–2030 рр.

За даними IWSR, у 2024 році світовий обсяг алкогольних напоїв скоротився на 1%, тоді як вартість зросла на 1%, причому преміалізація залишалася ключовим драйвером категорії пива [2].

Згідно зі звітом BarthHaas Report 2024/2025, світове виробництво пива у 2024 році склало 1 875,3 млн гектолітрів, що на 0,3% менше порівняно з попереднім роком (рисунок 1.2) [3].



	2023 1,000 hl	2024 1,000 hl	2023 +/– % rel.	2024 +/– % rel.
European Union	343,331	339,813	-3.5 %	-1.0 %
Rest of Europe	165,186	174,368	0.5 %	5.6 %
EUROPE TOTAL	**508,517**	**514,181**	**-2.2 %**	**1.1 %**
North America	356,930	349,876	-2.5 %	-2.0 %
Central America/Caribbean	24,089	24,309	1.0 %	0.9 %
South America	244,225	242,849	0.8 %	-0.6 %
AMERICA TOTAL	**625,244**	**617,034**	**-1.1 %**	**-1.3 %**

Рисунок 1.2 – Розвиток виробництва пива 2023/2024 за континентами

Сегмент крафтового пива демонструє стійке зростання, незважаючи на загальну стагнацію ринку. За даними Brewers Association, крафтове пиво в США зберегло частку ринку на рівні 13,3% у 2024 році, хоча обсяги виробництва скоротилися на 3,9% [4]. Згідно з прогнозами Mordor Intelligence, світовий ринок крафтового пива досягне 7,11 млрд літрів у 2025 році та зросте до 10,93 млрд літрів до 2030 року при середньорічному темпі зростання 9% (рисунок 1.3) [5].

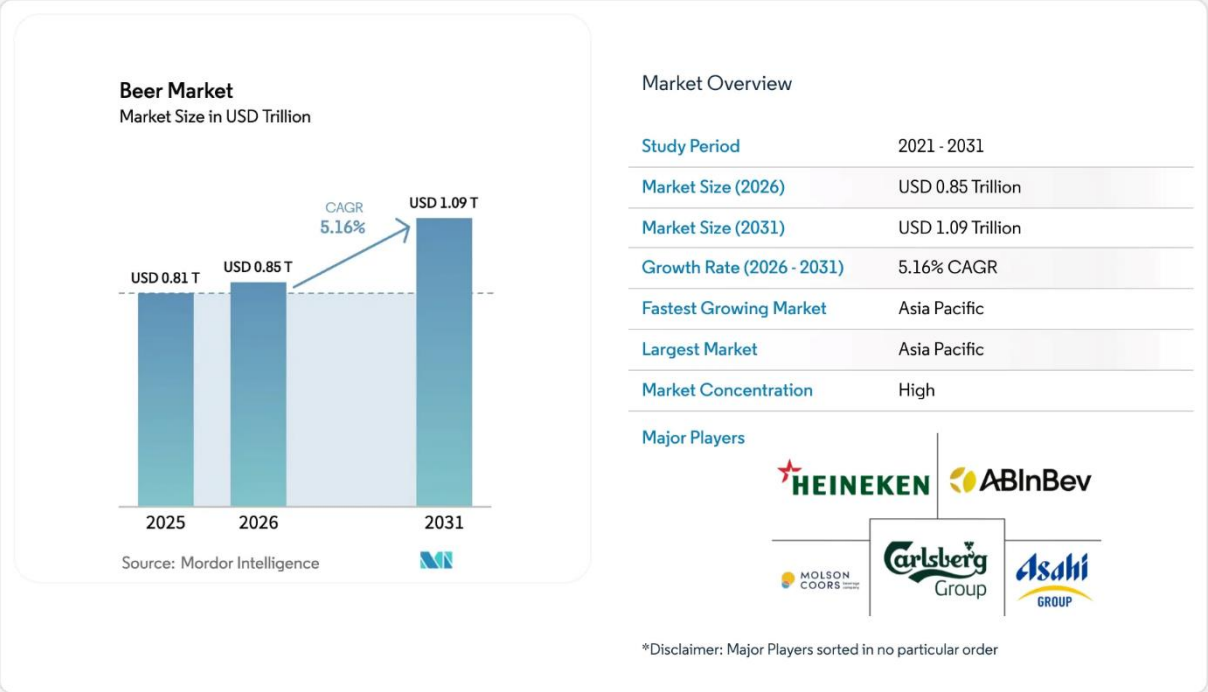


Рисунок 1.3 – Прогноз росту ринку крафтового пива до 2030 р.

Розширення асортименту пивної продукції супроводжується зростанням складності вибору для споживача. Програма сертифікації пивних суддів (Beer Judge Certification Program, BJCP) у редакції 2021 року класифікує 158 різних стилів пива, кожен з яких характеризується унікальним поєднанням ароматичних, смакових і візуальних властивостей, історичним контекстом та типовою сировинною базою [6]. Така різноманітність, з одного боку, створює передумови для задоволення індивідуальних уподобань, а з іншого – спричиняє явище, відоме як «decision fatigue» (втома від прийняття рішень), коли споживачі відчувають перевантаження через надмірну кількість опцій.

У відповідь на цю проблему сформувалися спеціалізовані цифрові платформи, які акумулюють користувацькі відгуки та рейтинги. Платформи RateBeer, BeerAdvocate та Untappd стали провідними світовими ресурсами, де споживачі публікують текстові рецензії, оцінюють пиво за органолептичними характеристиками, формують персональні колекції вподобаних сортів та обмінюються рекомендаціями. BeerAdvocate

позиціонується як еталонний ресурс для пивних рецензій, тоді як Untappd є основним застосунком для пошуку, оцінювання та обміну враженнями про пиво, забезпечуючи також інформацію про наявність сортів у локальних закладах.

Паралельно з розвитком платформ зростає споживчий запит на персоналізовані рекомендації. Згідно з дослідженням Kingfisher Drinks Boring Beer Index 2025, 77% споживачів охоче обрали б пиво, про яке вони раніше не чули, а 58,9% зазначили, що рекомендації персоналу впливають на їхній вибір під час ознайомлення з пивним меню [7]. Наведені дані вказують на високий потенціал впровадження інтелектуальних рекомендаційних сервісів у пивній галузі та обґрунтовують актуальність розроблення систем, здатних аналізувати текстові відгуки та формувати персоналізовані пропозиції.

1.2 Огляд існуючих підходів до рекомендацій напоїв на основі відгуків

Рекомендаційні системи для напоїв, зокрема пива, вина та кави, становлять окремий напрям досліджень у галузі штучного інтелекту. Специфіка предметної галузі полягає в необхідності врахування суб’єктивних органолептичних характеристик продукту, які важко формалізувати, проте які знаходять відображення в текстових відгуках користувачів. У цьому підрозділі проведено огляд наукових робіт, комерційних систем та застосунків, що реалізують рекомендації напоїв із використанням текстів відгуків або методів машинного навчання.

Платформи RateBeer, BeerAdvocate та Untappd є основними джерелами даних для академічних досліджень у галузі рекомендацій пива. На основі даних BeerAdvocate побудовано низку рекомендаційних систем, які використовують методи обробки природної мови (NLP) для вилучення ознак із текстів відгуків.

У роботі [8] представлено систему, реалізовану на основі Python з використанням бібліотек spaCy та VADER для обробки тексту та сентимент-аналізу відгуків. Користувач вводить бажані атрибути пива (наприклад, солодкість, фруктовість, насиченість), і система знаходить схожі характеристики у текстах відгуків за допомогою семантичного аналізу (spaCy). Потім за допомогою алгоритму VADER визначається тональність виявлених відгуків (позитивна чи негативна), і на цій основі система генерує рекомендації. Таким чином, інструмент поєднує контентно-орієнтований пошук по текстах з фільтрацією за емоційністю відгуків для вибору найбільш підходящих варіантів.

Система Beerlytics [9] реалізує інтерактивний підхід до рекомендації пива на основі аналізу текстових відгуків користувачів. Як джерело даних використовувалися рейтинги та текстові рецензії з відкритої онлайн-платформи для оцінювання пива. Для побудови числових представлень текстів застосовувалися два підходи до створення ембедингів: Word2Vec та StarSpace. Отримані векторні представлення документів використовувалися як вхідні ознаки для двох окремих класифікаторів на основі методу опорних векторів (SVM). Для формування персоналізованого профілю користувачеві пропонувалося вказати перелік улюблених і небажаних сортів пива, після чого система навчала класифікатор та прогнозувала перелік найбільш релевантних рекомендацій. Для оцінювання ефективності підходу результати порівнювалися з двома базовими методами: рекомендацією найпопулярніших сортів пива та рекомендацією на основі косинусної подібності між векторними представленнями продуктів. Підсумковий рейтинг формувався шляхом пізнього об’єднання результатів (late fusion) за допомогою механізму голосування. Експериментальна оцінка виконувалася на основі профілів реальних користувачів («штучних агентів»), які вже мали історію оцінювання напоїв. Результати показали, що моделі на основі Word2Vec демонструють вищу точність для вже відомих користувачеві

сортів пива, тоді як StarSpace забезпечує кращі результати при рекомендації нових продуктів та перевершує базовий підхід популярності.

Модель Beer2Vec [10] кодує пиво як вектор у багатовимірному просторі смаків. Для навчання використовуються дані з Untappd – 500 тис. чекінів за 14 600 різних крафтових пив, зокрема стандартизовані теги смаків (264 елементи). Beer2Vec спирається на аналогію зі словниковими векторними моделями: у центрі моделі – дві матриці-вбудування Θ^B і Θ^F для пив та тегів смаків (рисунок 1.4) відповідно. Навчання відбувається за схемою Skip-Gram: модель оптимізує ймовірність появи наближених за смаком тегів для кожного пива. В результаті кожне пиво та смак отримують векторне подання, що дозволяє порівнювати пива за смаковими профілями через евклідову відстань або векторні операції (додавати/віднімати смаки для зміни профілю). Автори демонструють, що Beer2Vec дає змістовні рекомендації: знайдені пива близькі за стилем та смаком до запитуваних. Модель було опубліковано з відкритим кодом і навіть реалізовано у веб-додатку для споживачів.

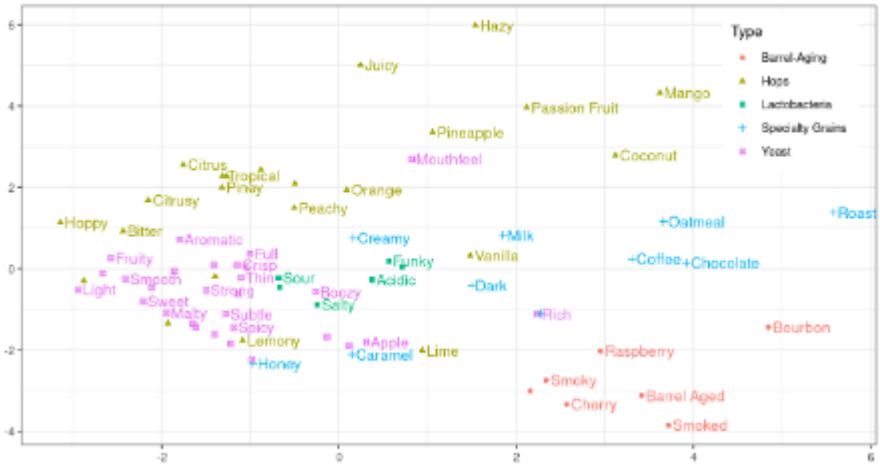


Рисунок 1.4 – 2D-проекція навченої матриці смаків

Ni, Li та Chang [11] поєднують колаборативну фільтрацію на основі рейтингів із аналізом тексту відгуків. У їхньому підході побудовано

двокомпонентну систему: перша модель генерує рекомендації пива на основі рейтингових даних (застосовуючи, наприклад, ALS або інші методи CF), а друга – класифікатор для визначення сентименту відгуків (позитивні/негативні). Дослідження охоплює порівняння десяти моделей класифікації (традиційних методів і глибинних мереж) для аналізу тональності тексту. Виявилося, що інтеграція цих двох компонент підвищує «довіру» до рекомендацій: систему можна сконфігурувати так, щоб виключати продукти з переважно негативними відгуками, що на практиці підвищує прийнятність рекомендацій для користувачів.

Робота [12] присвячена моделюванню розвитку «експертизи» користувачів. Автори запропонували латентно-факторну модель зі змінними рівнями досвіду. Кожен користувач моделюється як послідовність рівнів досвіду, що не зменшуються з часом; у міру зростання досвіду користувач переходить у відповідний підмодель (latent factor model), яка «розуміє» отримані нові смаки. На прикладі даних RateBeer (рисунок 1.5) показано, що початківці і експерти по-різному сприймають деякі стилі: наприклад, найбільш високі середні оцінки мали пива з дуже сильним хмелевим смаком, але новачкам їхня гіркота менш зрозуміла.

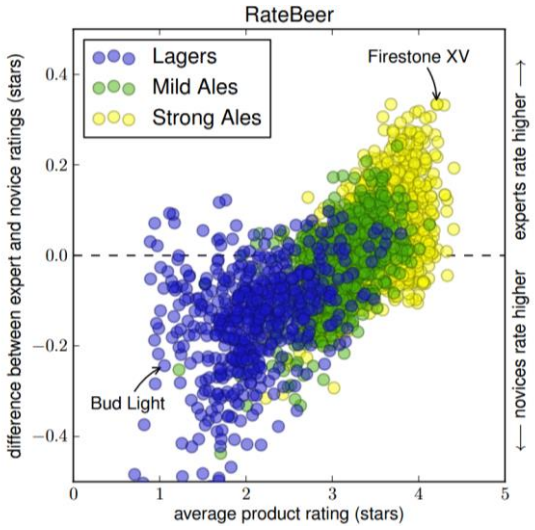


Рисунок 1.5 – Порівняння оцінок експертів і новачків для різних стилів пива на платформі RateBeer

Включення змінної «рівень досвіду» дало кращі рекомендації та дозволило виявити закономірності у «формуванні смаків» користувачів: досвідчені дегустатори дають більш високі оцінки сміливим стилям пива, чого не видно без урахування експертизи.

У галузі рекомендацій вина спостерігається активне використання гібридних підходів. Система soMLier [13] призначена для рекомендацій вина за даними платформи Vivino. Використовується гібридний підхід: item-based колаборативна фільтрація генерує списки релевантних вин, матрична факторизація (MF) прогнозує майбутні рейтинги, а контентно-орієнтовані методи на основі NLP забезпечують пояснення рекомендацій. В роботі дослідники спочатку протестували кожен компонент окремо: ІБ-CF добре знаходить схожі вина, MF дає точні оцінки рейтингу, а описові тексти вин (сорти, врожай, регіон) через NLP слугують базою для пояснення «чому» рекомендація. Оптимальне поєднання (гібридизація) показало кращі результати, ніж будь-який одиночний метод. Автори зазначають, що точність можна ще підвищити, маючи більше персональних даних про користувачів, але навіть у базовій реалізації soMLier допомагає новачкам відкривати нові вина і розуміти власні уподобання.

Дослідження [14] запропонувало архітектуру Side-Informed Attention Collaborative Filtering (SIACF) (рисунок 1.6). На відміну від класичних моделей, що використовують тільки рейтинги, тут вводяться додаткові «бокові» ознаки вина: рік збору винограду, сорт, кислотність, регіон тощо. За ідеєю авторів, дегустатори можуть звертати увагу на різні характеристики напою, тому в SIACF використовується механізм attention, який навчається «зважувати» релевантні ознаки в кожному конкретному випадку. У своїй роботі вони порівняли SIACF з класичним нейромережним колаборативним фільтром (NCF) на наборі X-Wines. SIACF показала кращу точність: наприклад, середня абсолютна помилка (MAE) знизилася на 2,98%, а RMSE – на 2,08% порівняно з базовим NCF. Автори зробили акцент на тому, що їхня модель добре інтегрує багату інформацію про

вино (через side information) і може бути адаптована до інших доменів (перевірено також на MovieLens).

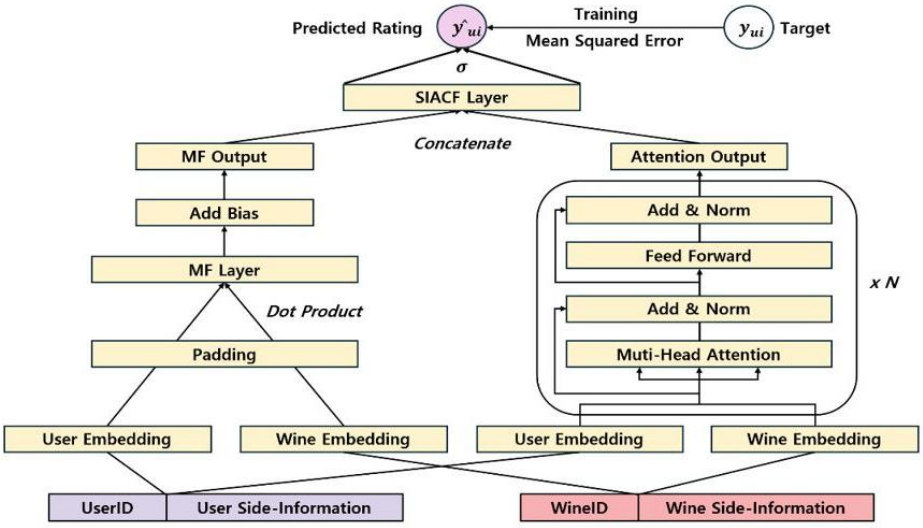


Рисунок 1.6 – Архітектура колаборативної фільтрації з побічним інформуванням (SIACF)

У фундаментальних оглядах теорії рекомендацій [15] описані методи матричної факторизації та SVD для колаборативної фільтрації. Ці моделі розкладають матрицю користувач–товар на два низькорангові простори (вбудовування), що дозволяє прогнозувати відсутні рейтинги. Такий підхід може бути безпосередньо застосований і до наборів з відгуками про напої, адже він зосереджений на латентних уподобаннях користувачів і характеристиках напоїв.

Musto та ін. [16] запропонували методи побудови пояснень до рекомендацій на основі текстів відгуків. Вони виходять з ідеї, що уривки з відгуків містять важливу інформацію про особливості продукту. Рамкова методологія працює так: на вхід подається запропоноване рекомендатором пиво і набір відгуків про нього, на виході – обґрунтування рішення природною мовою. Реалізовано три варіанти:

– NLP+sentiment: використовуються NLP-методи і аналіз тональності, щоб виділити ключові аспекти (особливості пива), що згадуються у відгуках. Потім система об’єднує фрагменти відгуків, де згадані ці аспекти, у пов’язну природно-мовну фразу, що пояснює рекомендацію;

– аспекти + саммаризація: до першого варіанту додається автоматичне вилучення аспектів та текстове узагальнення (summary). Система витягує головні характеристики продукту та створює єдиний синтезований текст обґрунтування;

– контекст-залежний: створюється «лексикон» для кожного контексту (ситуації споживання) на основі навчання, що дозволяє гнучко адаптувати формулювання пояснень до різних обставин;

Усі три моделі зроблені як «чорний ящик» поверх звичайного рекомендатора – тобто незалежно від основного алгоритму рекомендацій – і показали у користувацьких дослідженнях підвищення довіри та задоволеності користувачів завдяки прозорішим поясненням. Також у контексті їжі Musto та ін. [17] розробили спосіб генерувати порівняльні пояснення, що підкреслюють харчові властивості страв. Вони створили фреймворк, який бере користувача і дві рекомендовані страви і генерує текст, що аргументує «чому одне блюдо корисніше (або підходить більше) за інше». Для цього авторам вдалося згенерувати кілька стратегій: одні фокусувалися на поживних факторах та недоліках (наприклад, вміст цукру, жирів), інші – на загальному харчовому профілі. Проведені експерименти з користувачами (N=503) показали, що надання таких порівняльних обґрунтувань сприяє здоровішим виборам: коли система «пояснює» перевагу здоровішого блюда, люди частіше віддають йому перевагу. Цей підхід демонструє, як техніки NLP можуть сполучати відгуки/дані продукту зі створенням зрозумілих текстових пояснень у рекомендаторі.

Дослідження [18] зосередилося на прямому зв’язку між хімічними характеристиками пива та його смаком/суб’єктивною оцінкою споживачів. Вони зібрали унікальні дані: проаналізували хімічний склад 250 різних

пив (понад 200 хімічних властивостей кожного) і отримали 16-пунктову сенсорну оцінку від тренованої дегустаційної групи. Крім того, було зібрано понад 180 тисяч відгуків від споживачів (на Rating-платформах) з метою визначення прихильності людей. На основі цього великого набору даних навчено 10 моделей машинного навчання для прогнозу смаку та «апетитності» напою.

Найкраща модель (градієнтний бустинг) значно перевершила традиційні статистичні методи: вона точно передбачає складні органолептичні показники і оцінки споживачів з огляду на хімічний профіль пива (рисунок 1.7). Аналіз моделі дозволив виявити конкретні сполуки, що найбільше впливають на сприйняття смаку. Експериментальне додавання цих сполук у комерційні варіанти пива продемонструвало реальне покращення їхньої привабливості для споживачів.

Таким чином, ця робота показує, що поєднання великих датасетів «хімія + сенсорика + користувацькі відгуки» з ML-моделями дає змогу прогнозувати смаковий профіль пива і навіть підказувати, як його покращити.

Jannach та ін. [19] звернули увагу на те, що традиційна парадигма «рекомендації ≈ передбачення рейтингу» занадто вузька. Вони запропонували розглядати різні цілі з погляду споживача та провайдера. Наприклад, споживачу може бути важливо знайти щось несподіване, а продавцю – збільшити продажі певного сегменту. Автори склали схему різних завдань (long-term-переваги, оновлення каталогу, підписки, стимулювання продажів тощо), показавши, що чимало таких цілей ідентичних або подібних у різних галузях. Це означає, що у дослідженнях рекомендаторів слід враховувати не тільки точність прогнозів, а й інші метрики (наприклад, видимість нових продуктів, економічну користь, обізнаність користувачів). Підхід з урахуванням мети підкреслює необхідність розробляти та оцінювати системи за показниками, що відповідають конкретним цілям застосування.

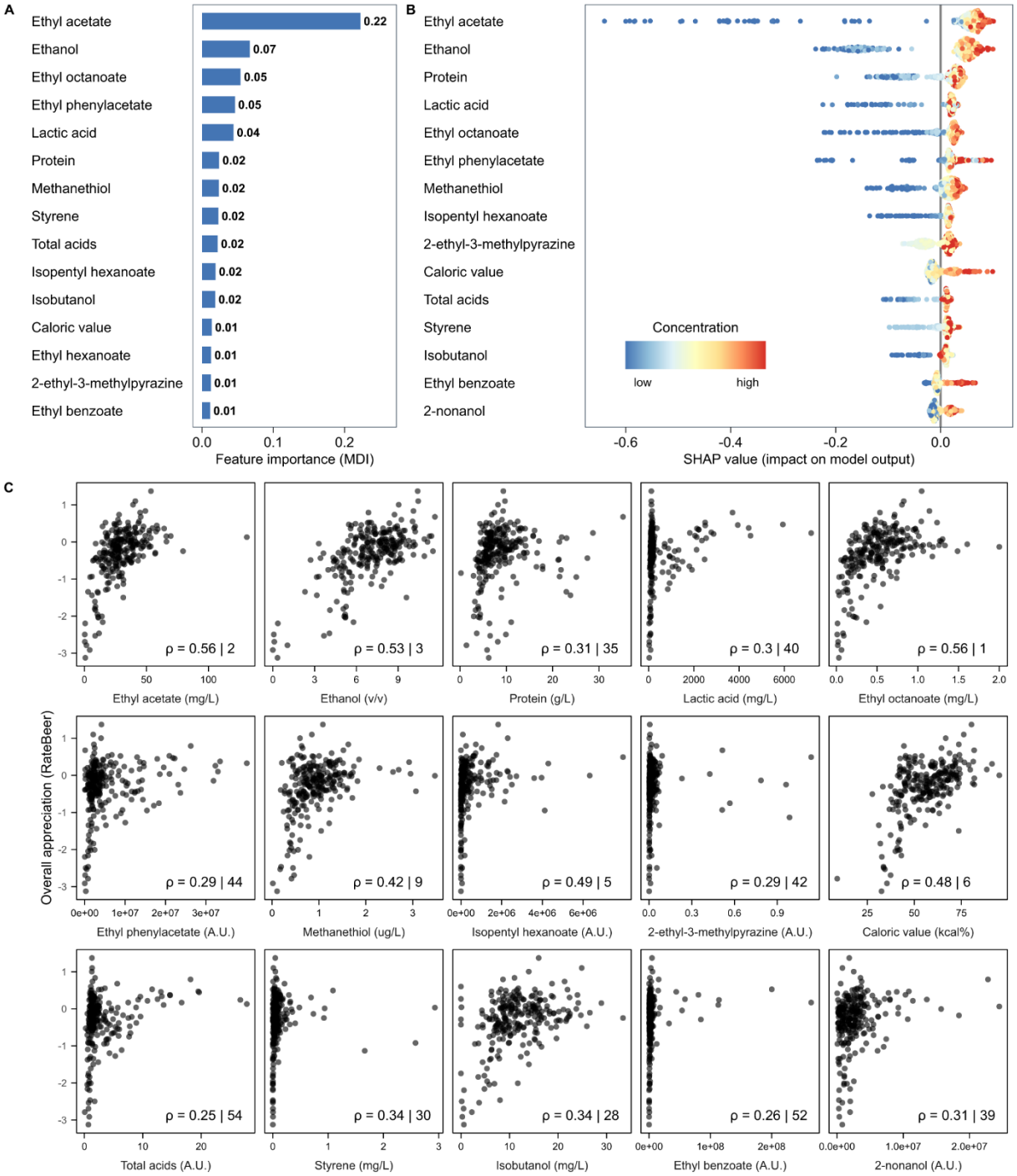


Рисунок 1.7 – Найважливіші хімічні параметри

Aggarwal та ін. [20] систематизовали широкий спектр технік для побудови рекомендаційних систем. Окрім класичної CF та контентного аналізу, він описує методи кластеризації, графові підходи і, особливо, застосування глибоких нейронних мереж. Сучасні нейронні колаборативні фільтри, розширені рекомендаційні моделі (DeepFM, Autoencoders тощо) та

NLP-техніки (наприклад, BERT для видобування особливостей із текстів відгуків) згадані як перспективні напрями.

Pierre Gutierrez [21] показав практичні реалізації нейронних моделей для рекомендацій пива. Спочатку він реалізував просту матричну факторизацію у Keras: навчає вбудовування (embedding) для користувачів і пив, а передбачення оцінки робить через скалярний добуток цих векторів. Потім він «розгорнув» модель у глибоку нейронну мережу: замінив операцію добутку на конкатенацію ембеддінгів і декілька Dense-шарів. Така модель з двома «прихованими» шарами показала значно кращу точність: якщо MF-модель досягала валідаційної MSE ≈0,42, то глибока модель – ≈0,20. Тобто використання нейромереж дозволило майже вдвічі зменшити помилку прогнозів. Цей приклад демонструє, як застосування популярних бібліотек (Python, Keras) і технік глибокого навчання може суттєво підвищити якість рекомендацій порівняно з базовою матричною факторизацією.

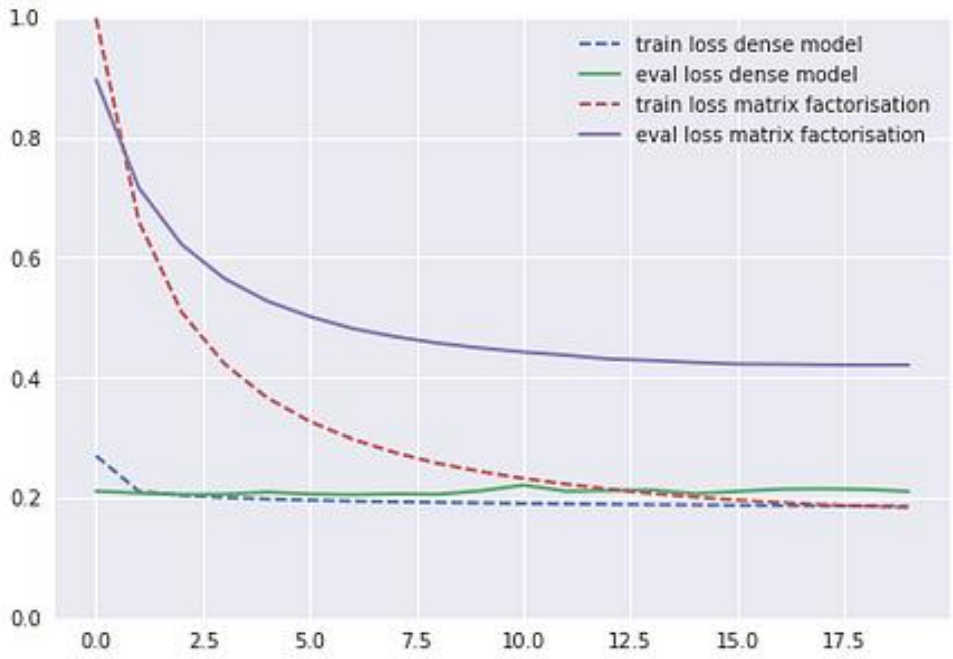


Рисунок 1.8 – Порівняння валідаційної MSE MF-моделі та Dense-моделі

У дослідженні Chinchanachokchai та ін. [22] розроблено рекомендаційну систему BeerXpert, призначену для аналізу впливу різних підходів до персоналізованих рекомендацій пива залежно від рівня експертності користувача. Система використовує дані відгуків та рейтингів, зібрані з онлайн-спільнот любителів пива, на основі яких реалізовано два окремі рекомендаційні механізми: user-based collaborative filtering та content-based filtering. Колаборативна модель формувала рекомендації шляхом пошуку користувачів зі схожими вподобаннями та прогнозування рейтингів на основі їхньої історії оцінок, тоді як контентно-орієнтований підхід використовував характеристики продуктів та інформацію з відгуків для визначення подібності між сортами пива. Для експериментальної оцінки автори створили програмний прототип BeerXpert та провели тестування за участю реальних користувачів із різним рівнем знань про пиво.

Результати показали відсутність статистично значущої різниці між підходами для початківців, однак досвідчені споживачі надавали перевагу рекомендаціям, отриманим за допомогою колаборативної фільтрації. Дослідження продемонструвало, що ефективність рекомендаційної системи залежить не лише від алгоритму, але й від рівня предметної експертизи користувача, що свідчить про доцільність адаптивного вибору рекомендаційних стратегій залежно від профілю споживача.

Аналіз існуючих робіт показує, що в рекомендаціях напоїв застосовують широкий спектр методів: від класичних CF і факторизаційних моделей до гібридних рішень і глибоких нейромереж. Багато уваги приділяється обробці текстових відгуків з метою виділення семантичних ознак (словникові моделі, BERT, аналіз тональності) і побудови інтерпретованих рекомендацій. Водночас у багатьох роботах помічаються недоліки: багато систем ігнорують специфічні сенсорні дескриптори пива (багато важить загальна тональність, але не виділено чітких смакових атрибутів згідно стандартів BJCP), а також відсутня обробка контексту споживання. Крім того, інтерпретованість моделей і довіра користувачів

залишаються непростими питаннями. Ці прогалини обґрунтовують необхідність подальшого розвитку – зокрема, створення рекомендаційних систем, що поєднують глибокий аналіз відгуків з урахуванням специфічних особливостей пивної галузі та контексту споживання (сезон, гастрономічне поєднання тощо), а також забезпечують зрозумілі обґрунтування для користувачів.

1.3 Постановка задачі дослідження

Проведений аналіз предметної галузі та огляд існуючих рекомендаційних систем для напоїв виявив низку суттєвих прогалин. Більшість наявних рішень слабко використовують специфічні сенсорні дескриптори пива, визначені міжнародними стандартами, не забезпечують достатньої інтерпретованості рекомендацій та практично не враховують контекст споживання. Подолання цих обмежень потребує створення нової системи, здатної глибоко аналізувати текстові відгуки користувачів та використовувати цю інформацію для формування персоналізованих рекомендацій на основі методів машинного навчання.

Метою даної роботи є розроблення інтелектуальної інформаційної системи рекомендацій пива, яка на основі аналізу користувацьких текстових відгуків і сучасних методів машинного навчання забезпечує формування персоналізованих пропозицій сортів пива з урахуванням індивідуальних органолептичних уподобань споживача.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

– провести аналіз предметної галузі, включаючи сучасний стан ринку пива, поведінку споживачів, існуючі цифрові платформи та наукові підходи до рекомендацій напоїв;

– зібрати та підготувати датасет текстових відгуків про пиво з відкритих джерел, виконати його очищення, нормалізацію та структурування для подальшого аналізу;

– виконати попередню обробку текстів відгуків із застосуванням методів обробки природної мови та витягти інформативні ознаки, зокрема сенсорні атрибути, тональність, стильові характеристики пива;

– дослідити та реалізувати моделі машинного навчання для формування рекомендацій, включаючи методи колаборативної фільтрації, гібридні підходи з використанням NLP-ознак, а також моделі на основі глибокого навчання;

– спроєктувати архітектуру інтелектуальної інформаційної системи, розробити її програмний прототип, що реалізує запропоновані методи та моделі.

– перевірити працездатність розробленого прототипу на тестових даних, проаналізувати отримані результати та сформулювати практичні рекомендації щодо впровадження системи.

2 АНАЛІЗ МЕТОДІВ ТА ПІДХОДІВ ДО ПОБУДОВИ РЕКОМЕНДАЦІЙНИХ СИСТЕМ НА ОСНОВІ ТЕКСТОВИХ ВІДГУКІВ

2.1 Методи колаборативної фільтрації

Колаборативна фільтрація (collaborative filtering, CF) (рисунок 2.1) є одним із найпоширеніших підходів до побудови рекомендаційних систем. Основна ідея CF полягає у використанні відомих уподобань групи користувачів для прогнозування невідомих уподобань інших користувачів. На відміну від контентно-орієнтованих методів, CF не потребує апріорних знань про властивості об’єктів, а спирається виключно на матрицю взаємодій «користувач-об’єкт», елементами якої можуть виступати явні оцінки (рейтинги), неявні сигнали (факти перегляду чи придбання) або текстові відгуки. У сучасній літературі виділяють два основні класи методів колаборативної фільтрації: методи на основі пам’яті (memory-based) та методи на основі моделі (model-based) [23].

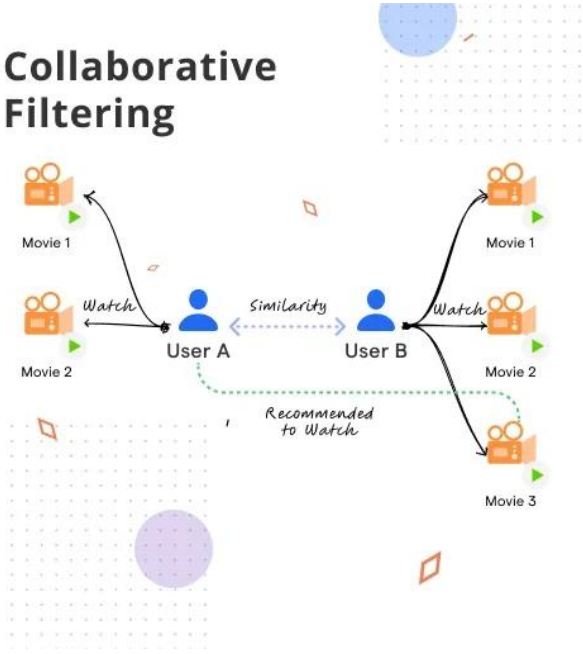


Рисунок 2.1 – Візуалізація колаборативної фільтрації

Методи на основі пам’яті, які також називають методами сусідства (neighbourhood-based), оперують безпосередньо матрицею взаємодій без побудови проміжної моделі. Їх поділяють на два різновиди: орієнтовані на користувача (user-based) та орієнтовані на об’єкт (item-based) (рисунок 2.2).

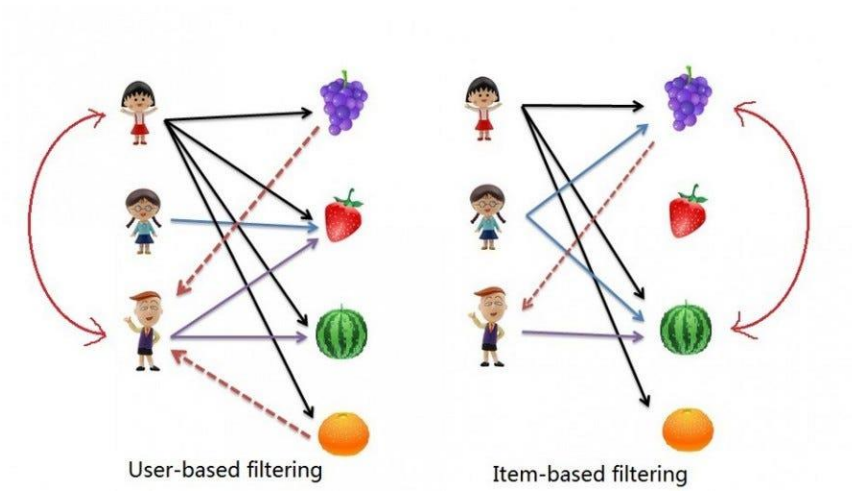


Рисунок 2.2 – Орієнтована на користувача та орієнтована на об’єкт колаборативні фільтрації

У підході user-based для цільового користувача обчислюються міри схожості з іншими користувачами на основі їхніх спільних оцінок, після чого формується сусідство з k найбільш подібних користувачів, оцінки яких агрегуються для прогнозування невідомого рейтингу.

У підході item-based аналогічна процедура виконується на множині об’єктів: для об’єкта, який потребує оцінювання, знаходяться подібні об’єкти, після чого оцінки цільового користувача для цих подібних об’єктів використовуються для прогнозу [24].

Для кількісного вираження схожості в memory-based методах застосовуються різні статистичні міри. Найчастіше використовуються косинусна схожість (cosine similarity), яка обчислює косинус кута між векторами оцінок двох користувачів або об’єктів, та коефіцієнт кореляції

Пірсона (Pearson correlation coefficient), який враховує відмінності в середніх значеннях оцінок і є менш чутливим до індивідуальних шкал оцінювання [24]. Емпіричні дослідження показують, що методи item-item із використанням кореляції Пірсона часто демонструють вищу точність порівняно з іншими конфігураціями memory-based підходів [24]. До переваг memory-based методів належать простота реалізації, прозорість та легка інтерпретованість результатів. Водночас вони мають суттєві обмеження: низьку масштабованість через квадратичну обчислювальну складність відносно кількості користувачів або об’єктів, а також вразливість до проблем розрідженості даних (data sparsity) та холодного старту (cold start) [23].

Методи на основі моделі (model-based) передбачають побудову математичної моделі, яка відображає приховані закономірності у взаємодіях між користувачами та об’єктами.

Центральне місце серед model-based підходів посідають методи матричної факторизації (matrix factorization, MF). Формально задача матричної факторизації полягає в апроксимації вихідної матриці рейтингів R розмірності m×n добутком двох матриць меншої розмірності: матриці латентних факторів користувачів P (розмірності $m \times k$) та матриці латентних факторів об’єктів Q (розмірності $n \times k$), де k – розмірність латентного простору, причому $k << min(m, n)$ [15]. Тоді прогнозований рейтинг користувача u для об’єкта і обчислюється як скалярний добуток їхніх латентних векторів:

$$\widehat{r_{ui}} = p_u^T \cdot q_i. \tag{2.1}$$

У класичному методі сингулярного розкладу (Singular Value Decomposition, SVD) матриця рейтингів представляється через її сингулярне розкладання, однак через наявність пропущених значень у реальних даних пряме застосування SVD є неможливим. Тому на практиці використовують

оптимізаційний підхід: значення латентних векторів знаходяться шляхом мінімізації функції втрат, що включає квадрати помилок на відомих рейтингах та регуляризаційні доданки для запобігання перенавчанню.

Для навчання моделей матричної факторизації застосовуються два основних методи оптимізації. Перший – стохастичний градієнтний спуск (Stochastic Gradient Descent, SGD) (рисунок 2.3), який на кожній ітерації оновлює латентні вектори на основі градієнта помилки для одного елемента матриці рейтингів. Другий – метод найменших квадратів із чергуванням (Alternating Least Squares, ALS), який почергово фіксує одну з матриць (P або Q) та розв’язує задачу найменших квадратів відносно іншої, що дозволяє ефективно розпаралелювати обчислення.

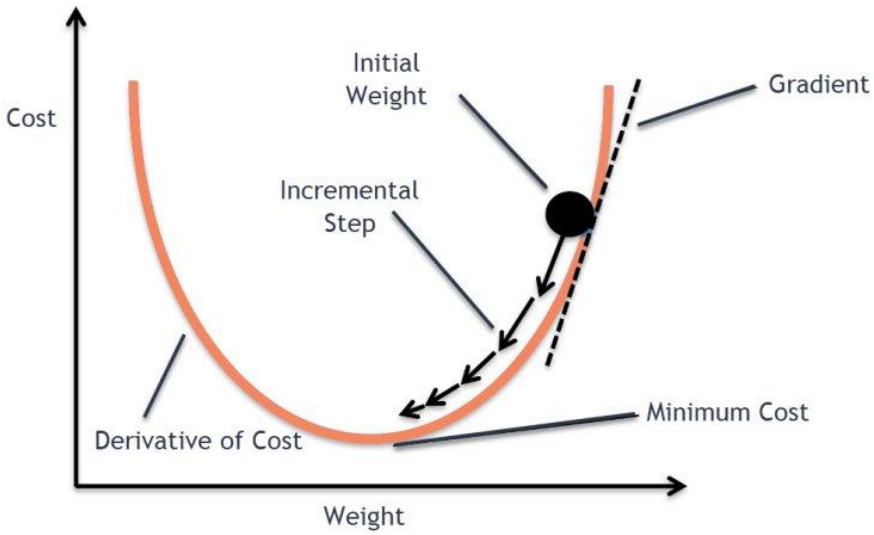


Рисунок 2.3 – SGD

Модель SVD++ є розширенням базового підходу SVD, яке додатково враховує неявний зворотний зв’язок від користувача (наприклад, які об’єкти він переглядав, але не оцінював). Це дозволяє підвищити точність прогнозів, особливо в умовах обмеженої кількості явних оцінок. Окремий напрям становить невід’ємна матрична факторизація (Non-negative Matrix Factorization, NMF), яка накладає обмеження невід’ємності на латентні

вектори (рисунок 2.4), що сприяє кращій інтерпретованості отриманих факторів, хоча й може поступатися SVD за точністю прогнозування в окремих задачах [15].

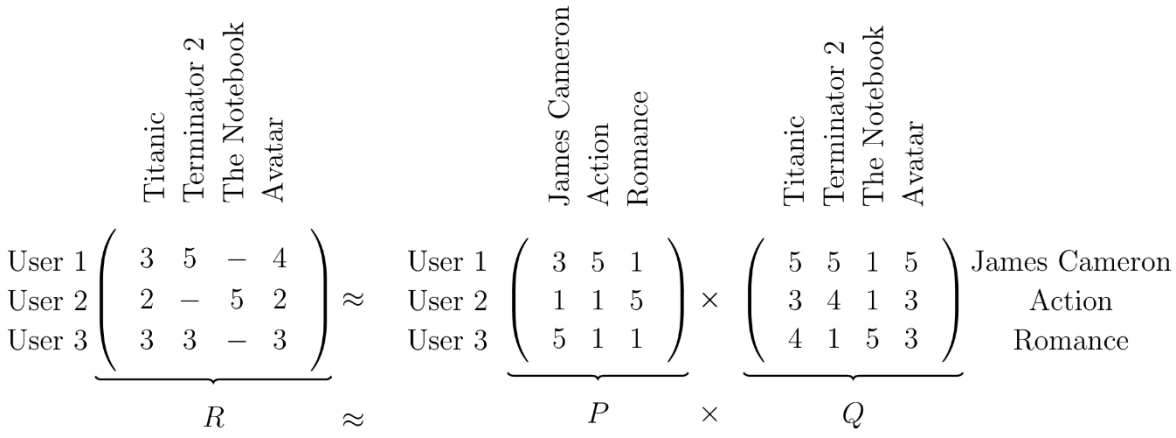


Рисунок 2.4 – NMF

Порівняно з memory-based підходами, model-based методи демонструють вищу масштабованість та стійкість до розрідженості даних [23]. Зокрема, дослідження [15] показує, що моделі SVD та SVD++ є найбільш ефективними рішеннями для роботи з розрідженими даними та в ситуаціях холодного старту користувачів.

Однак model-based методи потребують більшої обчислювальної потужності на етапі навчання, складнішої реалізації та ускладнюють інтерпретацію отриманих результатів порівняно з memory-based підходами [23].

Таким чином, колаборативна фільтрація надає потужний інструментарій для побудови рекомендаційних систем, причому вибір між memory-based та model-based методами визначається конкретними вимогами до системи: інтерпретованістю, масштабованістю, стійкістю до розрідженості даних та обчислювальними ресурсами.

2.2 Контентно-орієнтовані методи та обробка природної мови для аналізу текстів відгуків

На відміну від колаборативної фільтрації, яка спирається виключно на матрицю взаємодій, контентно-орієнтовані методи (content-based filtering) використовують описові характеристики об'єктів для формування рекомендацій (рисунок 2.5). У контексті рекомендацій пива такими характеристиками можуть виступати як структуровані метадані (стиль, вміст алкоголю, гіркота в IBU, колір у SRM), так і неструктурована текстова інформація, отримана з відгуків користувачів. Центральним завданням контентно-орієнтованого підходу є побудова профілів об'єктів та користувачів у спільному ознаковому просторі, де схожість між ними може бути кількісно оцінена [19].

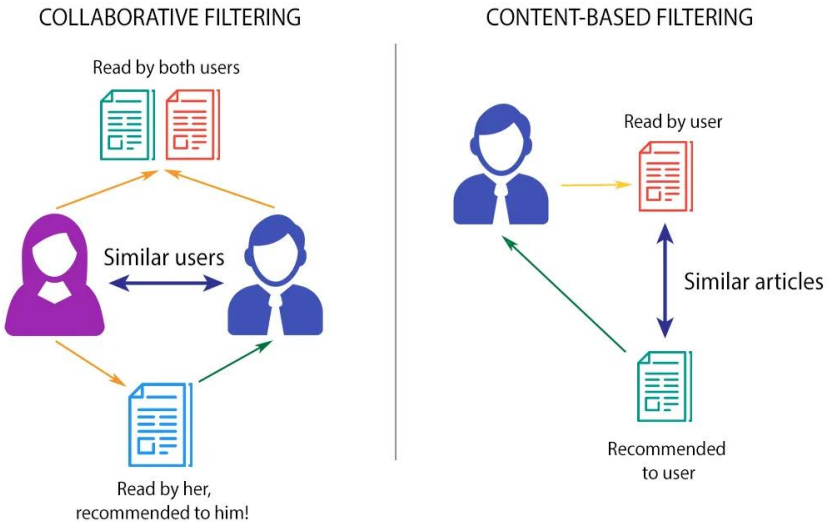


Рисунок 2.5 – Колаборативна фільтрація проти контентно-орієнтованої фільтрації

Побудова профілів об'єктів на основі текстових відгуків потребує застосування методів обробки природної мови (Natural Language Processing, NLP) для перетворення неструктурованого тексту в числові векторні

представлення. Найпростішими підходами до векторизації тексту є Bag-of-Words (BoW) та TF-IDF (Term Frequency – Inverse Document Frequency) (рисунок 2.6). Модель BoW представляє документ як вектор частот слів у ньому, ігноруючи порядок слів та їхні семантичні зв'язки. Метод TF-IDF удосконалює цей підхід шляхом зважування частот слів на обернену частоту їх появи в колекції документів, що дозволяє зменшити вплив загальновживаних слів та підсилити значущість рідкісних термінів [25].

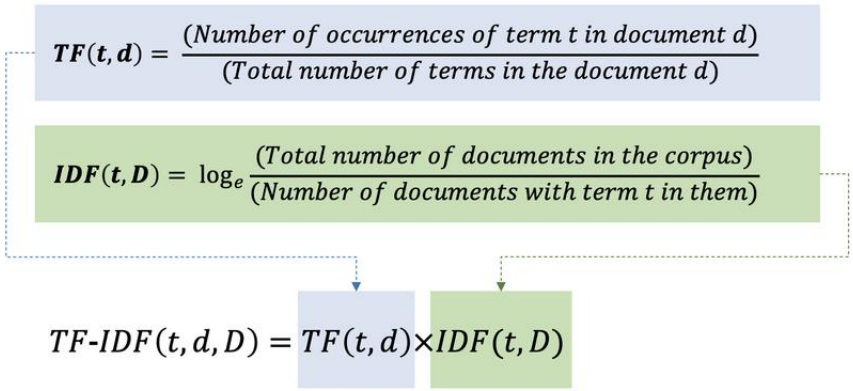


Рисунок 2.6 – TF-IDF

Суттєвим обмеженням класичних методів векторизації є нездатність враховувати семантичну близькість слів. Цю проблему вирішують дистрибутивні семантичні моделі, зокрема Word2Vec (рисунок 2.7), яка навчає векторні представлення слів на основі їхнього контекстного оточення. Word2Vec реалізує дві архітектури: Continuous Bag-of-Words (CBOW), що передбачає слово за його контекстом, та Skip-gram, що передбачає контекст за словом. Отримані векторні представлення фіксованої розмірності відображають семантичні відношення між словами, що було використано, зокрема, у системі Beerlytics для побудови рекомендацій пива [9]. Модель Beer2Vec [10] розширює цю ідею, кодуючи смакові характеристики пива безпосередньо у векторний простір на основі аналізу текстів відгуків.

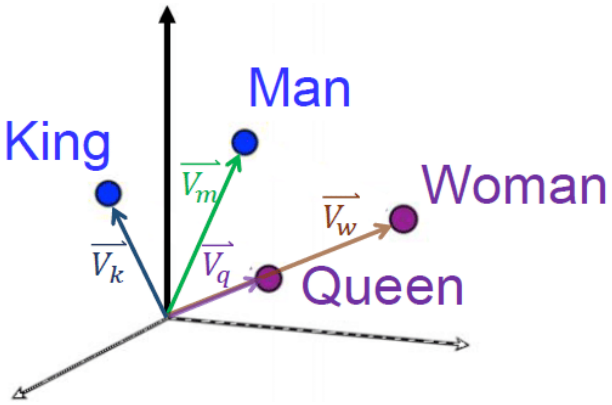


Рисунок 2.7 – Word2Vec

Подальшим розвитком дистрибутивних моделей стали контекстуалізовані векторні представлення, що враховують полісемію слів. Модель BERT (Bidirectional Encoder Representations from Transformers) та її полегшена версія DistilBERT (рисунок 2.8) використовують архітектуру трансформера з механізмом самоуваги (self-attention), що дозволяє отримувати векторні представлення, залежні від конкретного контексту вживання слова. BERT попередньо навчається на великих корпусах текстів із використанням двох задач: Masked Language Modeling (передбачення замаскованих слів) та Next Sentence Prediction (передбачення наступного речення).

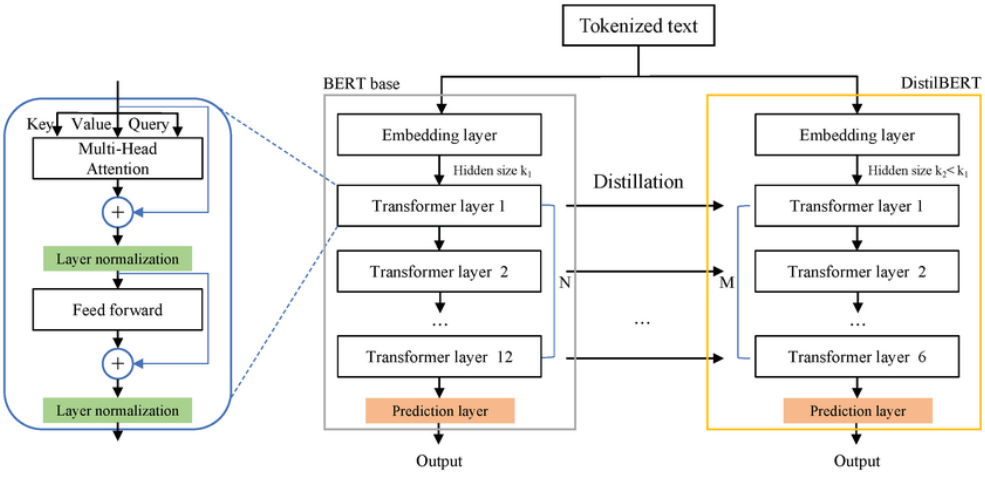


Рисунок 2.8 – Bert + Distilibert

Для отримання векторного представлення цілого відгуку може використовуватися вихід спеціального токена [CLS] або усереднення виходів усіх токенів останнього шару [26].

Окремим напрямом NLP, релевантним для аналізу відгуків про пиво, є аспектно-орієнтований аналіз тональності (Aspect-Based Sentiment Analysis, ABSA). На відміну від аналізу тональності на рівні документа, ABSA дозволяє виокремити конкретні аспекти продукту (наприклад, аромат, смак, післясмак, карбонізацію), згадані у відгуку, та визначити тональність щодо кожного з них. Для аналізу тональності на рівні документа широко застосовуються лексиконні підходи, такі як VADER (Valence Aware Dictionary and sEntiment Reasoner), що спеціально розроблений для аналізу текстів соціальних медіа та враховує інтенсифікатори, заперечення та емодзі. У роботі [8] VADER використовувався для сентимент-аналізу відгуків про пиво з метою подальшого формування рекомендацій.

Для більш точного аналізу тональності застосовується fine-tuning попередньо навчених моделей, таких як BERT, на розмічених наборах даних. Цей підхід дозволяє адаптувати загальну мовну модель до специфіки предметної галузі. У роботі [11] проведено порівняльний аналіз 10 класифікаційних моделей для задачі сентимент-аналізу відгуків про пиво, включно з fine-tuned BERT, що показало перевагу трансформерних архітектур над традиційними методами машинного навчання для цієї задачі.

Отримані в результаті NLP-обробки текстові ознаки можуть бути використані для побудови профілів продуктів у контентно-орієнтованих рекомендаційних системах. Профіль об'єкта формується як вектор, що агрегує інформацію про його сенсорні характеристики, виявлені аспекти та відповідні тональності з відгуків. Профіль користувача, у свою чергу, може бути побудований як усереднений вектор профілів об'єктів, яким користувач надавав високі оцінки, або шляхом безпосереднього аналізу текстів його відгуків для виявлення індивідуальних смакових уподобань [19]. Рекомендації формуються на основі обчислення

схожості (наприклад, косинусної) між профілем користувача та профілями об'єктів-кандидатів.

Контентно-орієнтовані методи мають низку переваг порівняно з колаборативною фільтрацією: вони не страждають від проблеми холодного старту для нових об'єктів (оскільки рекомендація базується на описі об'єкта, а не на історії взаємодій), забезпечують прозорість та інтерпретованість рекомендацій, а також не залежать від наявності великої кількості користувачів. Водночас їм притаманні обмеження: схильність до надмірної спеціалізації рекомендацій (відсутність серендипності), залежність від якості та повноти описових даних, складність врахування суб'єктивних уподобань, які не відображені в явних характеристиках продукту [19].

2.3 Гібридні рекомендаційні системи та моделі на основі глибокого навчання

Колаборативна фільтрація та контентно-орієнтовані методи мають комплементарні переваги та обмеження. Колаборативна фільтрація здатна виявляти складні патерни колективної поведінки користувачів, проте страждає від проблеми холодного старту та не забезпечує прозорості рекомендацій. Контентно-орієнтовані методи, навпаки, стійкі до холодного старту для нових об'єктів і забезпечують інтерпретованість, однак обмежені якістю описових даних та схильні до надмірної спеціалізації. Природним шляхом подолання цих обмежень є побудова гібридних рекомендаційних систем, які поєднують обидва підходи [19].

У сучасній літературі виділяють кілька основних стратегій гібридизації (рисунок 2.9). Зважене поєднання (weighted hybrid) передбачає обчислення фінальної оцінки релевантності як лінійної комбінації або іншої згортки оцінок, отриманих від окремих моделей. Каскадне поєднання (cascade hybrid) використовує одну модель для формування попереднього набору кандидатів, який потім уточнюється іншою моделлю.

Перемикаюче поєднання (switching hybrid) залежно від контексту (наявність історії користувача, повнота опису об'єкта) обирає одну з доступних моделей. Загальним принципом для всіх стратегій є використання сильних сторін кожного підходу для компенсації слабких сторін іншого [19].

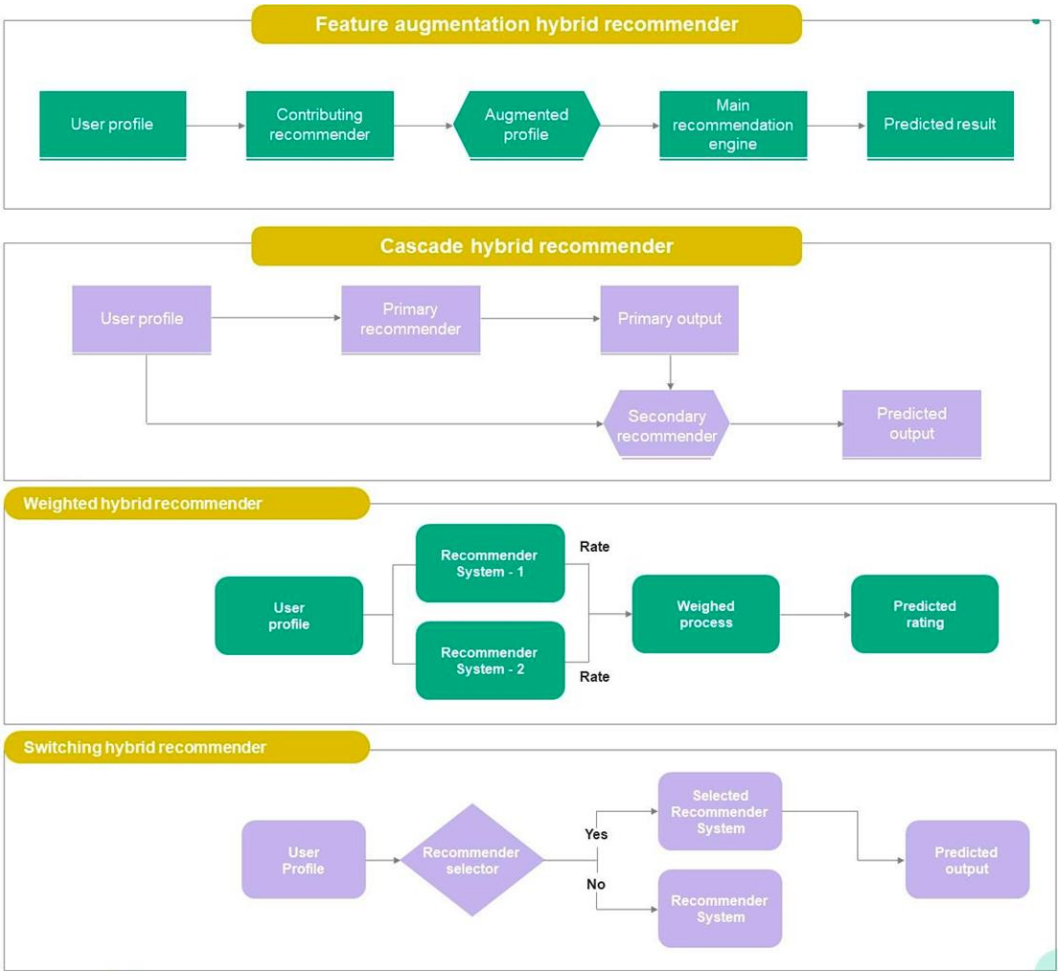


Рисунок 2.9 – Основні стратегії гібридизації

Розвиток глибокого навчання суттєво розширив можливості гібридних рекомендаційних систем. Однією з ключових архітектур є Neural Collaborative Filtering (NCF), запропонована He та ін. [27]. NCF замінює лінійну взаємодію між латентними векторами користувача та об'єкта, властиву класичній матричній факторизації, на нелінійну функцію, що моделюється багатошаровим перцептроном. Формально, узагальнена модель NCF визначається наступним чином:

$$\widehat{y_{ui}} = f\left(P^T \cdot v_u,\ Q^T \cdot v_i \mid P, Q, \Theta_f\right), \quad (2.2)$$

де f – нейронна мережа з параметрами Θ_f;

P та Q – матриці ембедингів;

v_u та v_i – one-hot вектори користувача та об'єкта.

Окремим випадком NCF є Generalized Matrix Factorization (GMF), яка відтворює логіку класичної MF, але з можливістю навчання оптимальних ваг взаємодії. Повна модель NCF поєднує GMF та MLP шляхом конкатенації їх виходів перед фінальним шаром [27].

Паралельним напрямом є факторизаційні машини (Factorization Machines, FM), які моделюють взаємодії між усіма парами ознак через факторизовані латентні вектори. На відміну від класичної MF, FM можуть включати довільні побічні ознаки (side features), такі як характеристики продукту, контекст взаємодії або демографічні дані користувача, що робить їх природним вибором для задач, де важливо враховувати описові характеристики об'єктів [14].

Глибокі варіанти факторизаційних машин, зокрема DeepFM та xDeepFM (рисунок 2.10), поєднують здатність FM моделювати парні взаємодії низького порядку з можливістю глибоких нейронних мереж вивчати складні нелінійні залежності високого порядку. DeepFM реалізує це через паралельне навчання компоненти FM та глибокої компоненти з подальшим об'єднанням їх виходів, тоді як xDeepFM запроваджує Compressed Interaction Network (CIN) для експліцитного моделювання взаємодій ознак на різних рівнях [28].

Архітектура Wide&Deep (рисунок 2.11) реалізує ідею паралельного поєднання двох компонент: широкої (Wide), яка відповідає за «запам'ятовування» (memorization) частих патернів через лінійну модель із перехресними ознаками, та глибокої (Deep), яка відповідає за «узагальнення» (generalization) через ембединги ознак та багатошаровий

перцептрон. Фінальне передбачення отримується як зважена сума виходів обох компонент [28].

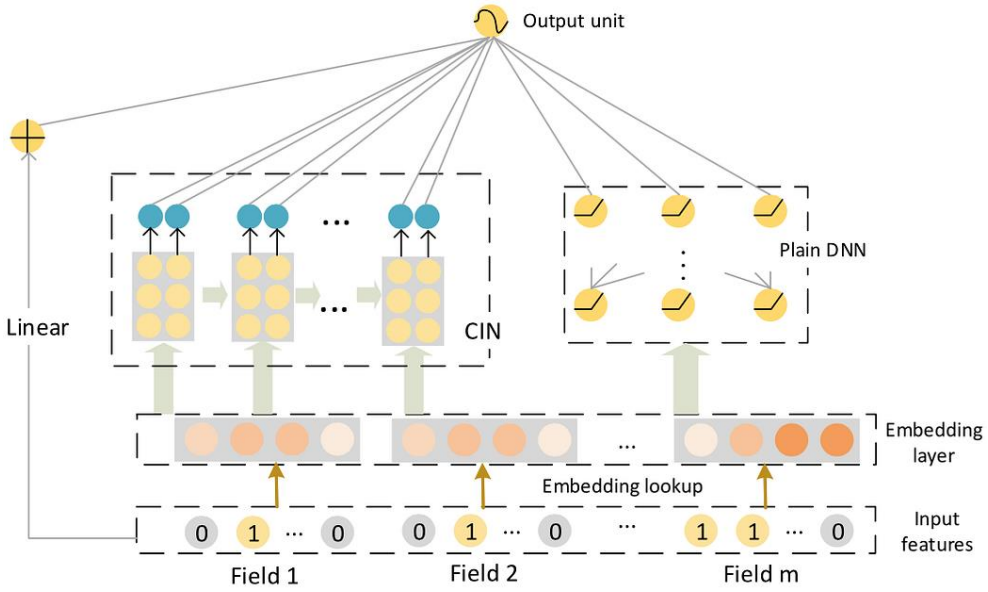


Рисунок 2.10 – xDeepFm

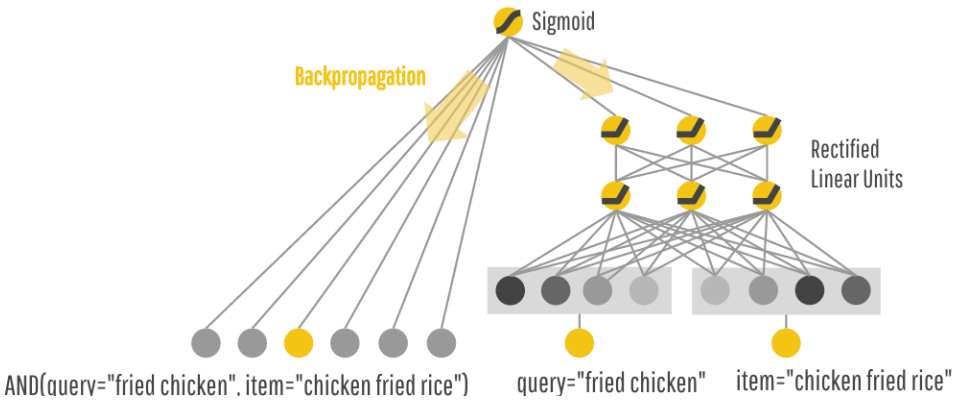


Рисунок 2.11 – Wide and deep

Окрему увагу в контексті даної роботи приділено використанню попередньо навчених мовних моделей у складі гібридних рекомендаційних архітектур. Моделі на основі BERT [26] можуть бути інтегровані як компонента вилучення ознак із текстових відгуків, продукуючи контекстуалізовані векторні представлення, які потім подаються на вхід

рекомендаційної моделі. Такий підхід дозволяє враховувати семантику текстів відгуків при формуванні рекомендацій. Зокрема, у роботі [20] запропоновано архітектуру, що поєднує методи пошуку та генерації з використанням великих мовних моделей для врахування як структурованих метаданих, так і неструктурованих текстових описів продуктів.

Підсумовуючи, гібридні рекомендаційні системи на основі глибокого навчання надають потужний інструментарій для поєднання переваг колаборативної та контентно-орієнтованої фільтрації. Для задачі рекомендацій пива на основі відгуків особливо перспективними є архітектури, здатні інтегрувати NLP-ознаки, вилучені з текстів відгуків, із колаборативними сигналами, отриманими з матриці взаємодій.

2.4 Метрики оцінювання якості рекомендаційних систем

Оцінювання якості рекомендаційних систем є критичним етапом їх розроблення, оскільки дозволяє кількісно порівнювати різні алгоритми, виявляти їхні сильні та слабкі сторони, а також перевіряти відповідність системи поставленим бізнес-цілям та очікуванням користувачів.

У літературі виділяють офлайн-оцінювання на історичних даних, онлайн-оцінювання в реальному середовищі (A/B тестування) та експертне оцінювання, проте на етапі дослідження та прототипування переважно застосовуються офлайн-метрики [20] (рисунок 2.12).

Сукупність використовуваних метрик прийнято поділяти на три категорії: метрики точності прогнозування рейтингів, метрики якості ранжування та метрики, що характеризують неточнісні властивості рекомендацій, такі як різноманітність, новизна та покриття.

Метрики точності прогнозування рейтингів призначені для оцінювання того, наскільки точно модель передбачає явні оцінки, які користувачі виставили об’єктам. Найпоширенішими серед них є

середньоквадратична помилка (Root Mean Squared Error, RMSE) та середня абсолютна помилка (Mean Absolute Error, MAE).

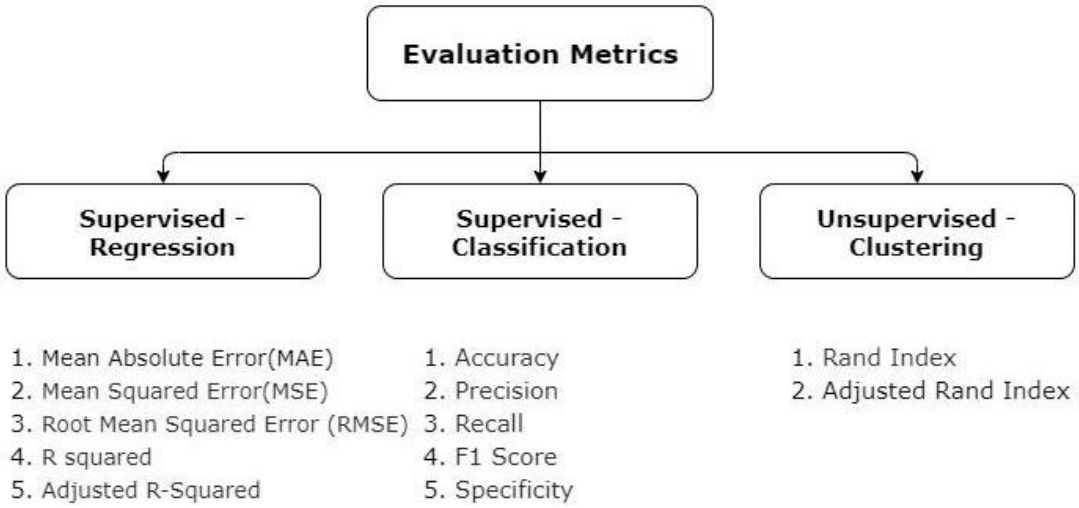


Рисунок 2.12 – Метрики оцінювання

RMSE обчислюється як квадратний корінь із середнього квадрата різниць між справжніми та передбаченими рейтингами:

$$\text{RMSE} = \sqrt{\frac{1}{|T|}\sum_{(u,i)\in T}(r_{ui} - \widehat{r_{ui}})^2}, \tag{2.3}$$

де T – множина тестових взаємодій;

r_{ui} – справжній рейтинг;

$\widehat{r_{ui}}$ – передбачений рейтинг.

MAE визначається як середнє абсолютних відхилень:

$$\text{MAE} = \frac{1}{|T|}\sum_{(u,i)\in T}|r_{ui} - \widehat{r_{ui}}|. \tag{2.4}$$

RMSE більш чутлива до великих помилок через квадрат відхилення, ніж MAE. Обидві метрики доцільно використовувати, коли рейтинги

представлені в явному вигляді, наприклад, за п’ятибальною шкалою на платформах RateBeer або BeerAdvocate. Проте для задачі рекомендацій пива, де кінцевою метою є формування впорядкованого списку пропозицій, точність передбачення окремого рейтингу має допоміжне значення, тому першочергову увагу слід приділяти метрикам ранжування.

Метрики якості ранжування оцінюють здатність системи формувати топ-k рекомендацій, релевантних для користувача. До базових належать Precision@k та Recall@k. Precision@k визначає частку релевантних об’єктів серед перших k рекомендованих:

$$\text{Precision@k} = \frac{1}{|U|}\sum_{u \in U}\frac{|L_u(k) \cap T_u|}{k}. \tag{2.5}$$

Recall@k – частку релевантних об’єктів з усієї множини релевантних, яку вдалося включити до топ-k:

$$\text{Recall@k} = \frac{1}{|U|}\sum_{u \in U}\frac{|L_u(k) \cap T_u|}{|T_u|}, \tag{2.6}$$

де U – множина користувачів;

$L_u(k)$ – список із k рекомендованих об’єктів для користувача u;

T_u – множина релевантних об’єктів у тестових даних.

Precision@k більш інтерпретовна з точки зору користувача, який бачить лише обмежений список, тоді як Recall@k важливий для оцінки повноти.

Недоліком Precision@k та Recall@k є ігнорування позиції об’єкта в списку. Усунути це дозволяє нормалізований дисконтований кумулятивний виграш (Normalized Discounted Cumulative Gain, nDCG@k), який враховує не лише факт релевантності, але й її ступінь, і штрафує за розташування релевантних об’єктів нижче в списку:

$$\mathrm{DCG}_{u@}k = \sum_{i=1}^{k} \frac{2^{rel_i}-1}{\log_2(i+1)}, \tag{2.7}$$

$$\mathrm{nDCG@k} = \frac{1}{|U|} \sum_{u \in U} \frac{\mathrm{DCG}_{u@k}}{\mathrm{IDCG}_{u@k}}, \tag{2.8}$$

де rel_i – релевантність об'єкта на позиції і (може бути двійковою або градуйованою, наприклад, фактичним рейтингом);

$\mathrm{IDCG}_{u@}k$ – ідеальне значення DCG за оптимального впорядкування.

У контексті рекомендацій пива nDCG дозволяє враховувати не лише те, чи сподобалося пиво користувачу, але й ступінь уподобання, що відповідає природі оцінювання на пивних платформах, де користувачі можуть виставляти оцінки з градацією [29].

Окрім точності, сучасні дослідження підкреслюють важливість метрик, що характеризують «неточнісні» властивості рекомендацій, оскільки максимізація лише точності може призводити до надмірно вузьких та передбачуваних рекомендацій, які не задовольняють потребу користувачів у дослідженні нового асортименту [7].

Різноманітність (diversity) рекомендацій оцінюється через показник внутрішньоспискової схожості (Intra-List Similarity, ILS). Для заданого списку L:

$$\mathrm{ILS}(L) = \frac{1}{|L|(|L|-1)} \sum_{i \in L} \sum_{j \in L, j \neq i} \mathrm{sim}(i,j), \tag{2.9}$$

де $\mathrm{sim}(i,j)$ – міра схожості між об'єктами, наприклад, косинусна схожість векторів ознак (стиль, смакові дескриптори).

Чим нижче значення ILS, тим більш різноманітними є рекомендації. У пивній галузі, де споживачі стикаються з проблемою «втоми від вибору» через величезну кількість стилів, забезпечення різноманітності є критичним: система повинна рекомендувати не лише IPAs різних виробників, а пропонувати збалансований набір стилів.

Новизна (novelty) рекомендацій характеризує, наскільки несподіваними для користувача є запропоновані об’єкти. Одним зі способів її вимірювання є обернена популярність, що базується на припущенні: чим менш популярним є об’єкт у загальній популяції, тим більш новим він є для конкретного користувача.

Формально, новизна списку може бути виражена як:

$$\text{Novelty}(L) = -\frac{1}{|L|}\sum_{i\in L}\log_2\left(\frac{|U_i|}{|U|}\right), \tag{2.10}$$

де $|U_i|$ – кількість користувачів, які взаємодіяли з об’єктом i,

$|U|$ – загальна кількість користувачів.

Це відповідає середній інформаційній ентропії. Високі значення вказують, що система пропонує менш відомі сорти, що узгоджується із бажанням 77% споживачів спробувати пиво, про яке вони раніше не чули [7], і є особливо актуальним для сегменту крафтового пива, де постійно з’являються новинки.

Покриття каталогу (catalog coverage) визначається як частка унікальних об’єктів, які хоча б раз з’явилися в рекомендаціях для певної групи користувачів:

$$\text{Coverage} = \frac{|\cup_{u\in U} L_u|}{|I|}, \tag{2.11}$$

де I – повний каталог об’єктів.

Низьке покриття свідчить про те, що значна частина асортименту залишається невидимою для користувачів, що може обмежувати бізнес-цінність системи для пивоварень та магазинів [30]. В умовах ринку крафтового пива, де асортимент налічує десятки тисяч найменувань, підтримання високого покриття є важливим для забезпечення рівномірної представленості продукції.

3 РОЗРОБКА ІНТЕЛЕКТУАЛЬНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ РЕКОМЕНДАЦІЙ ПИВА

3.1 Технологічний стек та інструментарій розроблення

Побудова інтелектуальної інформаційної системи рекомендацій пива потребує поєднання засобів для роботи з даними, методів машинного навчання, обробки природної мови та веб-розробки. Вибір конкретних інструментів визначався вимогами до продуктивності, можливості швидкого прототипування, наявністю необхідних бібліотек та зручністю інтеграції компонентів. У цьому підрозділі описано мову програмування, середовища розробки, ключові бібліотеки та їхню роль у реалізації системи.

Базовою мовою розроблення обрано Python версії 3.14. Такий вибір зумовлений насамперед надзвичайно широкою екосистемою бібліотек для наукових обчислень, машинного навчання, обробки природної мови та створення веб-сервісів, що дозволило реалізувати всі компоненти системи в межах однієї мови та уникнути зайвих інтеграційних витрат. Крім того, Python забезпечує високу читабельність вихідного коду та швидкий цикл прототипування, що є важливим у дослідницькому проєкті, де гіпотези щодо моделей та ознак потребують оперативної перевірки.

Основним інтерактивним середовищем розробки використовувався Visual Studio Code, який надає зручну підтримку віртуальних оточень Python, вбудовані засоби для налагодження, інтеграцію з системами контролю версій та широкий вибір розширень. Для виконання ресурсомістких обчислювальних завдань застосовувалося хмарне середовище Google Colab, яке надає безкоштовний доступ до графічних процесорів (зокрема NVIDIA T4). У Google Colab виконувалися дообучення моделі DistilBERT для аналізу тональності, тематичне моделювання за допомогою BERTopic, а також експерименти з побудовою TF-IDF матриць великої розмірності. Такий розподіл дозволив поєднати зручність локальної

розробки з обчислювальними можливостями хмарних GPU без додаткових фінансових витрат.

Серверна частина системи, що реалізує бізнес-логіку та надає REST API, побудована на основі веб-фреймворку FastAPI. Цей фреймворк забезпечує продуктивність, порівнянну з Node.js або Go, автоматично генерує інтерактивну документацію у форматі OpenAPI, виконує валідацію вхідних даних за допомогою моделей Pydantic та має вбудовану підтримку асинхронного виконання. FastAPI використовується для реалізації всіх ендпоінтів системи: отримання персоналізованих рекомендацій, перегляду профілів користувачів, додавання нових відгуків, пошуку пива за стилем та інших операцій.

Для зберігання даних обрано реляційну базу даних SQLite. Вона не потребує встановлення окремого сервера, зберігає всю базу в одному файлі, що спрощує розгортання та перенесення системи. Продуктивність SQLite є цілком достатньою для обсягів даних, які використовуються в проєкті (тисячі записів про користувачів, пиво, відгуки, профілі та матриці схожості). У базі зберігаються таблиці користувачів, сортів пива, відгуків, обчислених профілів, а також попередньо розраховані значення схожості між позиціями.

Для маніпуляцій із табличними даними та виконання агрегацій застосовується бібліотека Pandas. З її допомогою виконується завантаження вихідних CSV-файлів, обчислення середніх рейтингів, підрахунок частот тегів, фільтрація даних, а також формування матриць «користувач–пиво». Бібліотека NumPy використовується як основа для ефективної роботи з багатовимірними масивами, зокрема для зберігання та обробки матриці косинусної схожості між сортами пива, нормалізації вагових коефіцієнтів та виконання операцій лінійної алгебри в рекомендаційних алгоритмах.

Обробка природної мови є центральним компонентом системи, і для її реалізації залучено комплекс бібліотек. Бібліотека NLTK (Natural Language Toolkit) використовується для попередньої обробки текстів

відгуків: лематизації через WordNetLemmatizer, яка зводить словоформи до початкової форми, а також для визначення та видалення стоп-слів. Лематизація дозволяє узагальнювати слова зі спільним коренем, що є важливим для подальшого порівняння зі словниками смакових тегів.

Аналіз тональності реалізовано на основі бібліотеки Transformers компанії Hugging Face. Зокрема, попередньо навчена модель DistilBERT (distilbert-base-uncased) була дообучена на власному розміченому корпусі відгуків для класифікації тексту за трьома класами: негативний, нейтральний та позитивний. Transformers також використовується для роботи з готовими моделями тональності.

Для автоматичного розширення словника смакових характеристик застосовано бібліотеку BERTopic. Вона реалізує тематичне моделювання на основі трансформерних ембедингів та алгоритму HDBSCAN, дозволяючи виділяти з корпусу відгуків зв’язні теми (наприклад, «кава, шоколад, стаут» або «грейпфрут, сосна, IPA») та отримувати набори ключових слів для кожної теми. Ці ключові слова, після валідації, додаються до ручного словника тегів, збагачуючи систему здатністю розпізнавати ширший спектр смакових дескрипторів.

Бібліотека Scikit-learn забезпечує значну частину функціоналу машинного навчання, що не потребує глибоких нейронних мереж. TfidfVectorizer використовується для перетворення колекції текстів відгуків у матрицю ознак TF-IDF, яка дозволяє оцінити важливість окремих слів у корпусі та виявити кандидатів для поповнення словника тегів. Функція cosine_similarity застосовується для обчислення колаборативної складової: на основі матриці оцінок будується матриця косинусної схожості між усіма сортами пива, яка потім використовується для прогнозування оцінки конкретним користувачем як зваженого середнього оцінок подібних сортів. Для підготовки даних до навчання моделі тональності застосовано RandomUnderSampler, що дозволило збалансувати кількість негативних,

нейтральних та позитивних прикладів у навчальній вибірці та уникнути зсуву моделі в бік домінуючого класу.

Оптимізація вагових коефіцієнтів гібридної рекомендаційної формули виконана за допомогою бібліотеки scikit-optimize (skopt). У ній реалізовано байєсівську оптимізацію на основі гауссівських процесів (функція gp_minimize), яка дозволяє ефективно шукати оптимальне поєднання ваг п’яти компонентів (стиль, теги, рейтинг, тональність, колаборативна складова) за метрикою Precision@10, мінімізуючи кількість необхідних ітерацій порівняно з сітковим чи випадковим пошуком.

Користувацький інтерфейс реалізовано за допомогою бібліотеки Streamlit, яка дозволяє створювати інтерактивні веб-застосунки виключно на Python без необхідності написання коду на HTML, CSS або JavaScript. Інтерфейс складається з трьох основних екранів: сторінка профілю, де відображаються статистичні дані користувача та його вподобання у вигляді зважених списків стилів і смакових тегів; сторінка рекомендацій, яка виводить картки запропонованих сортів пива із зазначенням фінального скорингового балу та текстового пояснення внеску окремих компонентів; сторінка додавання нового відгуку, де демонструються результати NLP-обробки щойно введеного тексту (виявлені теги та оцінка тональності).

Таблиця 3.1 – Призначення компонентів технологічного стеку

Компонент	Роль у системі
Python 3.14	Мова реалізації всіх модулів
FastAPI	REST API для зв’язку фронтенду з бекендом
Streamlit	Веб-інтерфейс користувача
SQLite	Зберігання всіх даних (користувачі, пиво, відгуки, профілі, схожість)
Pandas / NumPy	Агрегація даних, матричні обчислення
NLTK	Лематизація текстів відгуків
Transformers	Аналіз тональності (fine-tuned DistilBERT)
BERTopic	Автоматичне розширення словника смакових тегів

Продовження таблиці 3.1

Компонент	Роль у системі
Scikit-learn	Косинусна схожість, TF-IDF, балансування вибірки
Skopt	Байєсівська оптимізація ваг компонентів рекомендаційної формули

3.2 Підготовка набору даних

Формування якісного набору даних є критичним етапом розроблення рекомендаційної системи, оскільки саме від повноти, збалансованості та коректності даних залежить здатність моделей виявляти справжні вподобання користувачів. У цьому підрозділі описано джерело початкових даних, виконані операції з їх очищення, скорочення та нормалізації, а також структуру бази даних, що забезпечує зберігання й оновлення інформації в процесі роботи системи.

Початковий датасет отримано з відкритого репозиторію Kaggle. Вихідний CSV-файл містив такі основні поля: beer/beerId, beer/name, beer/style, beer/ABV, review/overall (числова оцінка від 1 до 5), review/text (текст відгуку), user/profileName (унікальний ідентифікатор користувача). Проте структура даних мала суттєвий недолік: кожному користувачу відповідав лише один відгук, що унеможливлювало формування персоналізованих профілів на основі історії оцінювання.

Для подолання цієї проблеми було виконано процедуру створення віртуальних користувачів (рисунок 3.1). Кожному оригінальному ідентифікатору profileName випадковим чином було призначено новий user_id у діапазоні від 1 до 50. Після початкового розподілу проведено перевірку: тим користувачам, які отримали менше трьох відгуків, додано додаткові відгуки шляхом переназначення їх від інших користувачів, доки всі 50 віртуальних користувачів не отримали не менше трьох відгуків. Це забезпечило мінімальний обсяг інформації для побудови профілів.



Рисунок 3.1 – Перерозподіл відгуків на 50 користувачів

Для зменшення обчислювального навантаження та прискорення експериментів обсяг даних було скорочено. З повного переліку сортів пива залишено 165 найпопулярніших – за кількістю наявних відгуків. Загальну кількість відгуків обмежено до 1000, при цьому виконано балансування: відібрані відгуки охоплюють різноманітні оцінки та рівномірно розподілені між 50 користувачами.

Результуючий набір даних було розбито на три нормалізовані таблиці (рисунок 3.2), що відповідають ключовим сутностям предметної галузі: користувачі (users), пиво (beers) та відгуки (reviews).

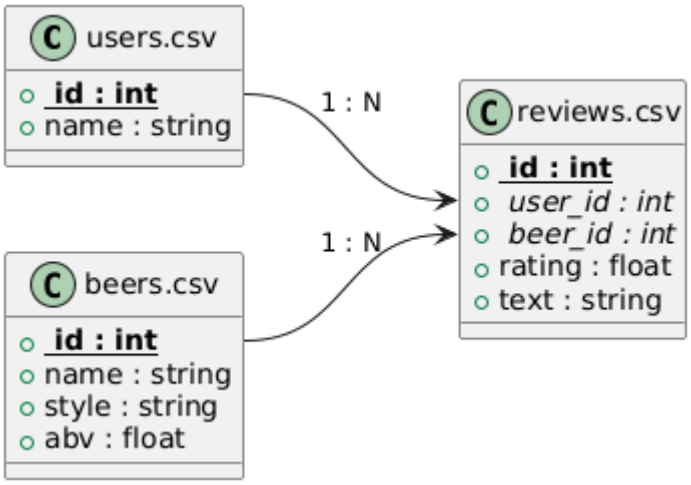


Рисунок 3.2 – Структура вхідних даних (CSV-файли)

Текстові відгуки, перш ніж використовуватися для вилучення смакових характеристик, пройшли кілька етапів очищення. Усі літери переведено в нижній регістр, видалено пунктуацію, цифри, зайві пробіли, символи табуляції та перенесення рядків. Після цього текст розбито на окремі слова, до кожного з яких застосовано лематизацію за допомогою WordNetLemmatizer із бібліотеки NLTK. Лематизація, на відміну від простого стемінгу, зводить слово до його словникової форми з урахуванням частини мови, що підвищує точність подальшого зіставлення зі словниками тегів.

Для задачі аналізу тональності оцінки review/overall було перетворено на три категорії: негативні (1–2 бали), нейтральні (3 бали) та позитивні (4–5 балів). Природний розподіл оцінок у вихідних даних виявився суттєво зміщеним у бік позитивних відгуків, що могло б призвести до навчання моделі, нечутливої до рідкісних класів. Для усунення цього дисбалансу застосовано метод RandomUnderSampler із бібліотеки scikit-learn, який випадковим чином вилучає приклади з мажоритарних класів до досягнення рівної кількості зразків у кожному. Як наслідок, навчальна вибірка для fine-tune моделі DistilBERT містила однакову кількість негативних, нейтральних та позитивних речень.

Оброблені дані зберігаються в реляційній базі даних SQLite, що складається з одного файлу і не потребує адміністрування.

Структуру бази утворюють сім таблиць, логічно пов’язаних зовнішніми ключами (рисунок 3.3). Така організація відповідає третій нормальній формі та забезпечує цілісність даних під час оновлень.

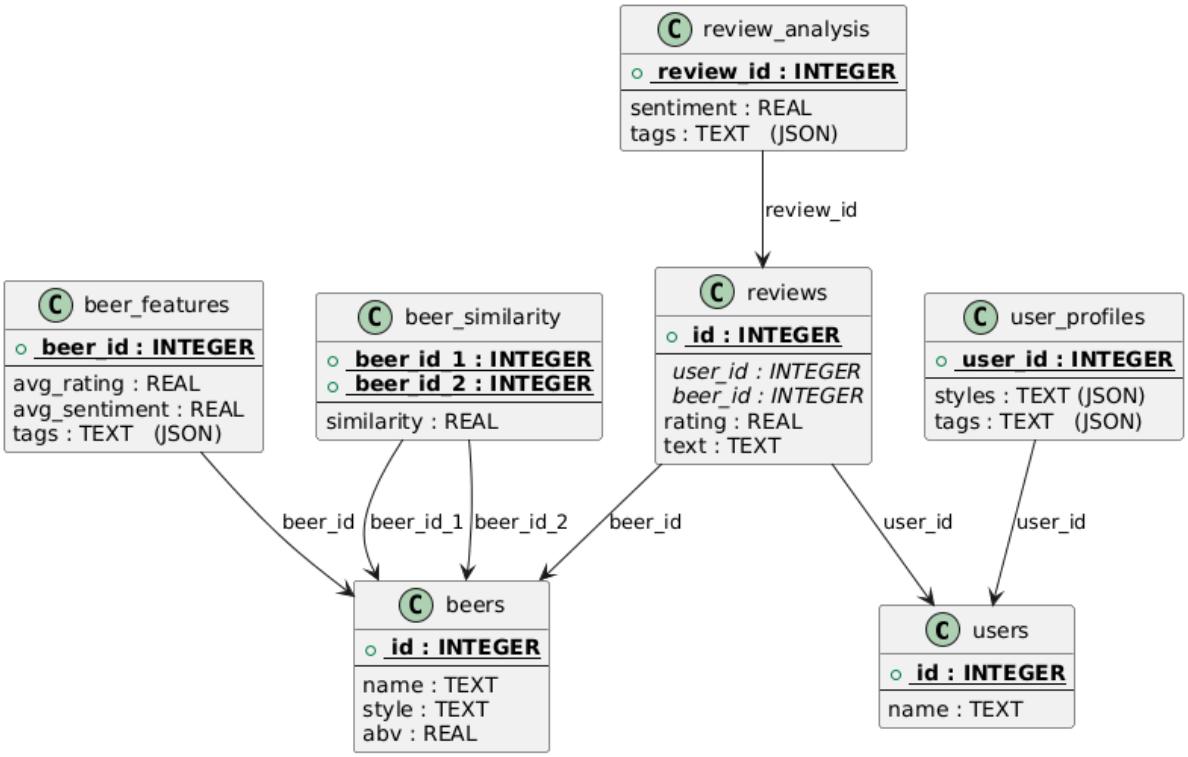


Рисунок 3.3 – Схема бази даних рекомендаційної системи

Таблиця users зберігає ідентифікатори та імена 50 віртуальних користувачів. Таблиця beers містить 165 сортів пива з їхніми назвами, стилями та вмістом алкоголю. У таблиці reviews зосереджено 1000 відгуків, кожен із яких пов’язаний із конкретним користувачем і пивом. Результати NLP-аналізу окремого відгуку фіксуються в таблиці review_analysis: значення тональності та список виявлених у тексті смакових тегів. На основі цих даних обчислюються агреговані характеристики пива (beer_features) – середній рейтинг, середня тональність та перелік із десяти найбільш частотних тегів. Профіль користувача (user_profiles) акумулює історію його

оцінювання у вигляді нормалізованих вагових коефіцієнтів для стилів і тегів. Таблиця beer_similarity зберігає попередньо розраховані значення косинусної схожості між усіма парами пива.

Життєвий цикл даних у системі передбачає послідовне виконання кількох етапів (рисунок 3.4).

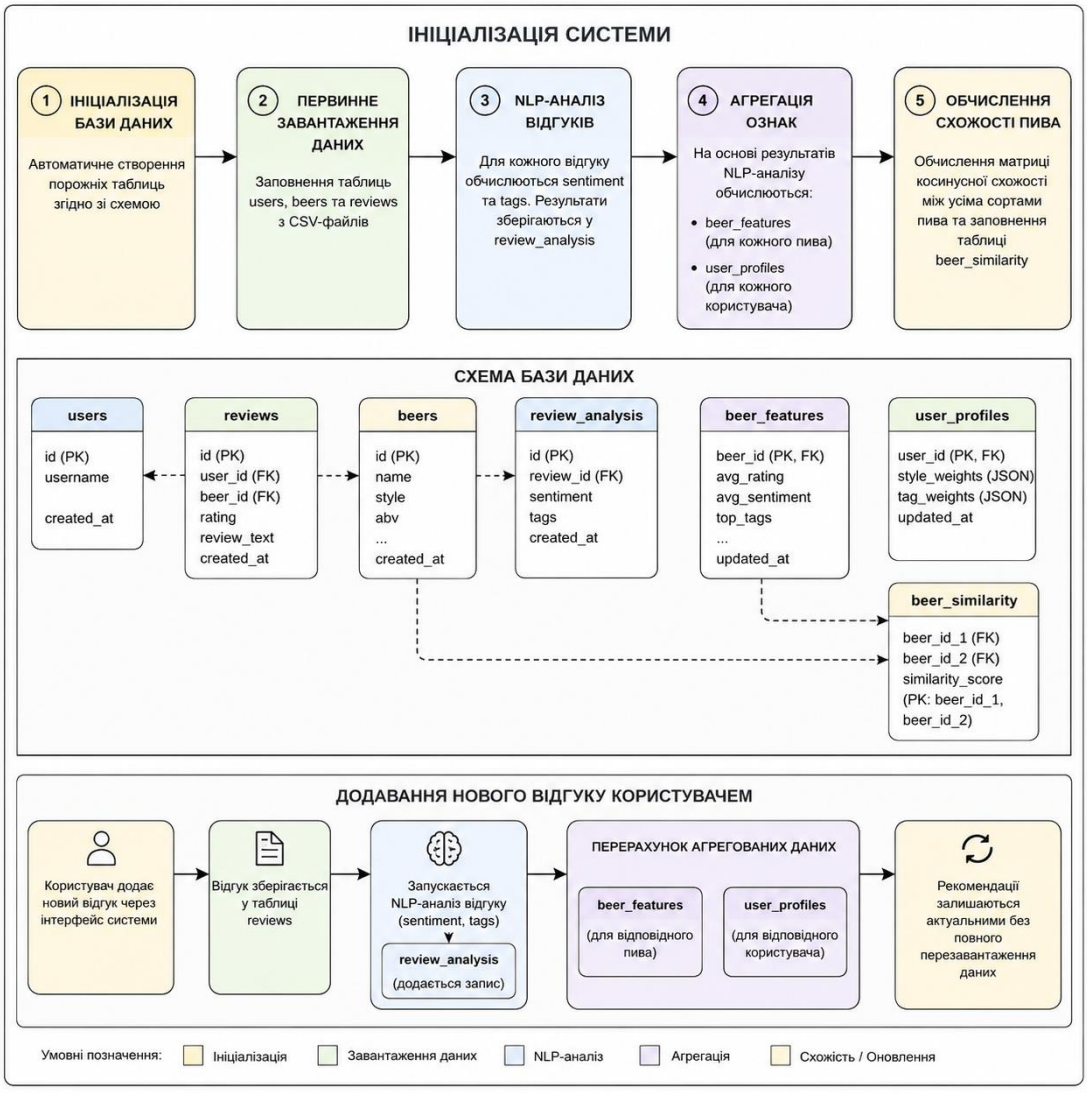


Рисунок 3.4 – Блок-схема життєвого циклу даних у системі рекомендацій

На етапі ініціалізації автоматично створюються порожні таблиці згідно з наведеною схемою. Далі виконується первинне завантаження: таблиці users, beers та reviews наповнюються даними з підготовлених CSV-файлів. Після цього для кожного відгуку запускається модуль NLP-аналізу,

який обчислює sentiment та tags, зберігаючи результат у review_analysis. На основі цих результатів агрегуються beer_features (середній рейтинг, сентимент, топ-теги) та user_profiles (зважені ваги стилів і тегів). Завершальним кроком ініціалізації є обчислення матриці косинусної схожості між усіма сортами пива та заповнення таблиці beer_similarity. У подальшому, при додаванні користувачем нового відгуку через інтерфейс системи, відбувається автоматичний перерахунок агрегованих таблиць beer_features та user_profiles для відповідних пива та користувача, що гарантує актуальність рекомендацій без повного перезавантаження даних.

Отже, підсумковий набір даних містить 50 віртуальних користувачів, 165 найпопулярніших сортів пива та 1000 відгуків, для кожного з яких виконано очищення тексту, лематизацію та визначення тональності. Нормалізована структура бази даних із зовнішніми ключами забезпечує цілісність інформації, а передбачений механізм інкрементального оновлення дозволяє системі адаптуватися до нових даних без перезапуску.

3.3 Методи попередньої обробки відгуків та формування ознак

Ефективність гібридної рекомендаційної системи значною мірою визначається якістю ознак, вилучених із текстів відгуків, та коректністю побудованих на їх основі профілів. Першим етапом обробки кожного відгуку є визначення його тональності. Для цього використано модель DistilBERT (distilbert-base-uncased), яка була дообучена на власному розміченому корпусі. Вихідні оцінки review/overall перетворено на три класи: негативний (1–2 бали), нейтральний (3 бали) та позитивний (4–5 балів) (рисунок 3.5). Оскільки природний розподіл оцінок у датасеті виявився зміщеним у бік позитивних відгуків, для усунення дисбалансу застосовано метод RandomUnderSampler, який випадково вилучає приклади з мажоритарних класів до досягнення однакової кількості зразків у

кожному. Дообучення виконувалося в середовищі Google Colab із використанням бібліотеки Transformers та інструменту Trainer. Результатом роботи моделі є числове значення тональності в діапазоні від -1 (негативний відгук) до +1 (позитивний), яке зберігається в таблиці review_analysis.



Рисунок 3.5 – Класи тональності

Другим ключовим компонентом NLP-обробки є вилучення з тексту відгуку смакових тегів – дескрипторів, що характеризують органолептичні властивості пива. Основою для цього слугує ручний словник TAG_DICT, який налічує понад 20 смакових категорій. Кожна категорія має власну вагу (наприклад, citrus, pine, tropical – 1.0; bitter, hoppy – 1.2; roasty, coffee, chocolate – 1.1; sweet, caramel – 1.0) та містить набір ключових слів-синонімів, включно з лематизованими формами. Якщо хоча б одне ключове слово категорії знайдено в попередньо обробленому тексті відгуку, відповідний тег додається до переліку.

Оскільки ручне створення словника не може охопити все різноманіття мови споживачів, було застосовано два додаткові методи його розширення. По-перше, використано бібліотеку BERTopic для тематичного моделювання на корпусі відгуків. Алгоритм автоматично виділив 17 зв’язних

тем (зокрема «coffee chocolate stout», «grapefruit pine ipa», «smoky wood», «banana clove hefe»), для кожної з яких згенеровано набір ключових слів. Ці слова, після ручної валідації, було додано до відповідних категорій TAG_DICT або оформлено як нові теги (наприклад, rye, bourbon, raisin, pumpkin_spice). По-друге, проведено аналіз TF-IDF за допомогою TfidfVectorizer із бібліотеки scikit-learn, що дозволило виявити слова з високою частотою в корпусі, які не були враховані попередніми методами. Серед таких кандидатів, після експертної перевірки, до словника додано теги на кшталт creamy, refreshing, maple.

На основі результатів аналізу окремих відгуків формуються профілі користувачів та агреговані ознаки пива. Профіль користувача будується шляхом зваженого підсумовування інформації з усіх залишених ним відгуків. Для кожної оцінки rating обчислюється вага за формулою:

$$weight = (rating - 1) / 4. \quad (3.1)$$

За такого підходу максимальна оцінка 5 отримує вагу 1.0, нейтральна оцінка 3 – вагу 0.5, а мінімальна оцінка 1 – вагу 0.0, тобто негативні відгуки фактично ігноруються під час побудови профілю. Для кожного відгуку його вага додається до відповідного стилю пива, після чого отриманий вектор стилів нормалізується діленням на максимальне значення. Аналогічна процедура виконується для смакових тегів: вага відгуку додається до кожного тегу, виявленого в його тексті, з подальшою нормалізацією. У результаті профіль користувача зберігається в таблиці user_profiles у вигляді двох JSON-об’єктів: словник стилів із вагами в діапазоні [0,1] (наприклад, {"IPA": 0.9, "Stout": 0.4}) та словник тегів із вагами (наприклад, {"citrus": 0.8, "hoppy": 0.6, "malty": 0.2}). Така структура дозволяє кількісно виразити як загальні стильові вподобання, так і конкретні смакові ноти, яким користувач надає перевагу (рисунок 3.6).

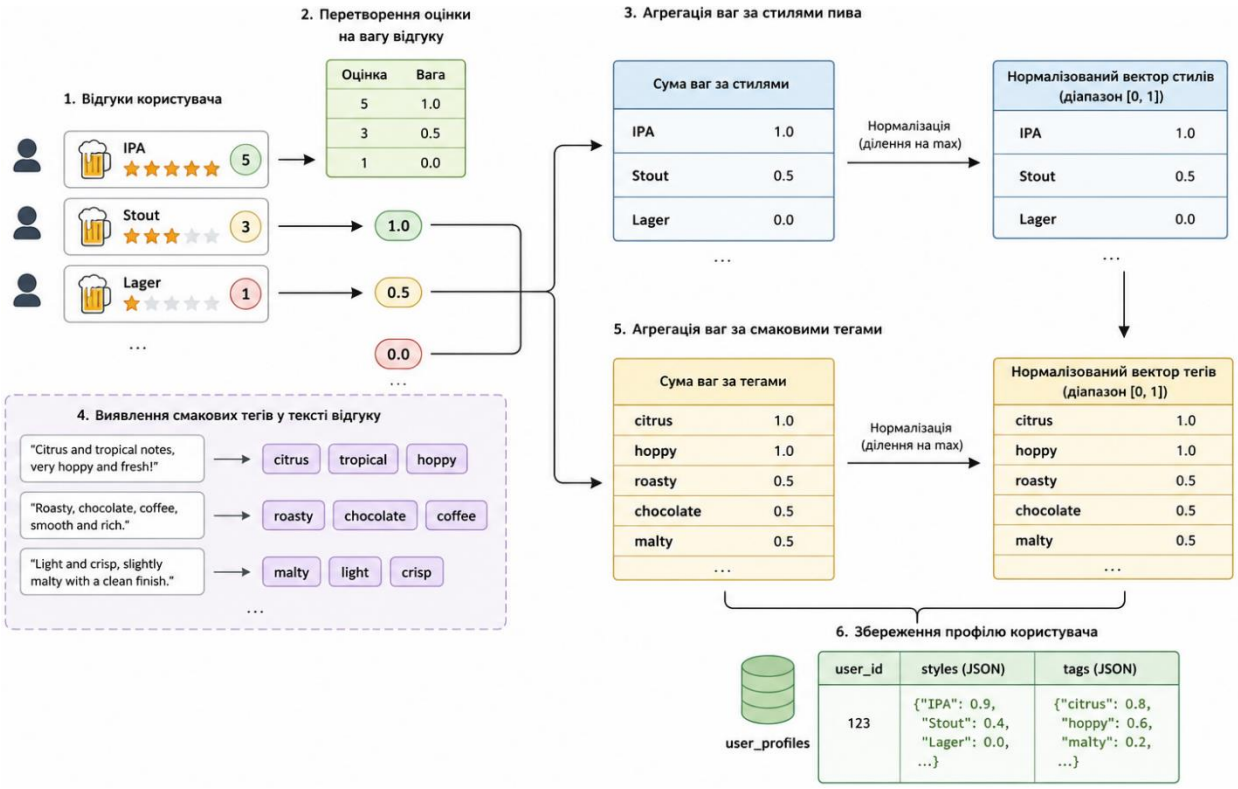


Рисунок 3.6 – Схема обробки відгуків

Паралельно з профілями користувачів обчислюються ознаки для кожного сорту пива (таблиця beer_features). На основі всіх відгуків, що стосуються конкретного пива, обчислюються три характеристики: середній рейтинг (avg_rating) як середнє арифметичне оцінок, середня тональність (avg_sentiment) як середнє арифметичне значень сентименту, та перелік із десяти найчастотніших тегів (tags), що зустрічаються у відгуках на це пиво. Отримані ознаки використовуються як контентна складова в гібридній рекомендаційній формулі.

Вибір описаних методів зумовлений їхньою здатністю забезпечувати високу якість виділених ознак в умовах обмеженого обсягу даних. Fine-tuned DistilBERT демонструє кращу точність класифікації тональності порівняно з rule-based підходами, особливо для коротких текстів, насичених сленговою лексикою та специфічною термінологією пивної галузі. BERTopic дозволяє автоматично виявляти приховані теми, що суттєво скорочує ручну працю з розширення словника тегів, тоді як TF-IDF виступає

додатковим інструментом для пошуку кандидатів у теги на основі статистичних властивостей корпусу. Зважена агрегація з урахуванням оцінки відгуку дає змогу диференціювати вподобання за інтенсивністю: теги та стилі з високо оцінених відгуків отримують більший внесок у профіль, ніж ті, що згадані в посередніх відгуках.

Таким чином, реалізований пайплайн перетворює сирі тексти відгуків у набір кількісних характеристик, які в подальшому використовуються для побудови персоналізованих рекомендацій. Поєднання fine-tuned DistilBERT для тональності, комбінованого підходу до вилучення тегів та зваженої агрегації забезпечує повноту й точність ознакового простору, необхідного для гібридної моделі.

3.4 Побудова гібридної рекомендаційної моделі

Гібридна модель рекомендацій поєднує колаборативну складову, засновану на схожості оцінок, та контентні складові, отримані з NLP-аналізу відгуків. Інтеграція різнорідних сигналів дозволяє компенсувати обмеження окремих підходів і формувати більш точні та різноманітні рекомендації.

Колаборативна складова базується на матриці «користувач–пиво», яка будується на основі таблиці reviews. Рядками матриці виступають 50 користувачів, стовпцями – 165 сортів пива, а значеннями – оцінки від 1 до 5. Усі пропущені позиції (пиво, яке користувач не оцінював) заповнюються нулями. Далі для кожної пари пив обчислюється косинусна схожість за їхніми вектор-стовпцями, що відображають профілі оцінок цих пив різними користувачами. Обчислення виконується за формулою:

$$similarity = (A \cdot B) / (||A|| * ||B||), \quad (3.2)$$

де A та B – вектори оцінок двох сортів пива.

Результати зберігаються в таблиці beer_similarity для всіх пар, чия схожість перевищує поріг 0.01, що дозволяє зменшити обсяг даних, виключаючи пари з незначущою подібністю.

Прогнозування оцінки цільового пива i для конкретного користувача виконується як зважене середнє. Для цього з таблиці beer_similarity відбираються всі сорти j, схожі на i, які користувач уже оцінив. Для кожного такого j враховується оцінка r_{uj} та схожість sim(i, j). Прогнозована оцінка обчислюється за правилом:

$$pred_rating = \Sigma\,(sim(i,j) * r_\{uj\}) \,/\, \Sigma\, sim(i,j). \quad (3.3)$$

Отримане значення знаходиться в діапазоні від 1 до 5. Для подальшого використання в гібридній формулі воно нормалізується в інтервал [0, 1] шляхом лінійного перетворення:

$$collab_score = (pred_rating - 1) \,/\, 4. \quad (3.4)$$

Контентні компоненти рекомендацій спираються на профілі користувачів та ознаки пива, сформовані на попередньому етапі. З профілю користувача отримуються ваги стилів та тегів, що відображають індивідуальні вподобання. Для кожного кандидата (пива) обчислюються чотири незалежні оцінки:

– style_score – відповідність стилю пива стильовим уподобанням користувача. Якщо стиль пива присутній у профілі користувача, береться його нормалізована вага, інакше – 0. Діапазон значень: [0, 1];

– tag_score – середній збіг тегів пива з улюбленими тегами користувача. Для кожного тегу зі списку пива (beer_features.tags) перевіряється його вага в профілі користувача, після чого обчислюється середнє арифметичне знайдених ваг. Діапазон: [0, 1];

– rating_score – нормалізований середній рейтинг пива, обчислений за формулою (avg_rating – 1) / 4, що переводить оцінку з діапазону [1, 5] у [0, 1];

– sentiment_score – нормалізована середня тональність відгуків про пиво: (avg_sentiment + 1) / 2, що відображає переважно позитивне чи негативне сприйняття продукту спільнотою. Діапазон: [0, 1].

Підсумковий рейтинг кандидата формується як лінійна комбінація зазначених компонентів із невід'ємними вагами:

$$score = style_score * w_style + tag_score * w_tag + rating_score * w_rating + sentiment_score * w_sentiment + collab_score * w_collab. \quad (3.5)$$

3.5 Архітектура системи та програмна реалізація

Програмна система реалізована за клієнт-серверною архітектурою, що передбачає чітке розділення на backend (серверну частину, яка містить бізнес-логіку, модулі машинного навчання та API для доступу до даних) та frontend (клієнтську частину, що надає користувацький інтерфейс). Усі дані зберігаються в реляційній базі даних SQLite. Взаємодія між клієнтом і сервером здійснюється через REST API за протоколом HTTP. Backend відповідає за обробку запитів, виконання NLP-аналізу, розрахунок рекомендацій та агрегацію даних, тоді як frontend забезпечує візуалізацію результатів і дає змогу користувачеві взаємодіяти з системою. Загальна архітектура системи наведена на рисунку 3.7.

Серверну частину побудовано на основі веб-фреймворку FastAPI, який працює під управлінням ASGI-сервера Uvicorn. FastAPI забезпечує високу продуктивність обробки запитів, автоматичну генерацію документації у форматі OpenAPI та валідацію даних за допомогою моделей Pydantic.

Структура backend-частини організована за модульним принципом. Каталог api містить головний файл main.py, в якому визначаються всі ендпоінти, та файл models.py із описом структур запитів і відповідей. Бізнес-логіка зосереджена в шарі сервісів services, що включає модулі beer_service.py (робота з пивом та його характеристиками), user_service.py (користувачі та профілі), review_service.py (додавання й отримання відгуків із запуском NLP) і recommendation_service.py (формування персоналізованих рекомендацій).

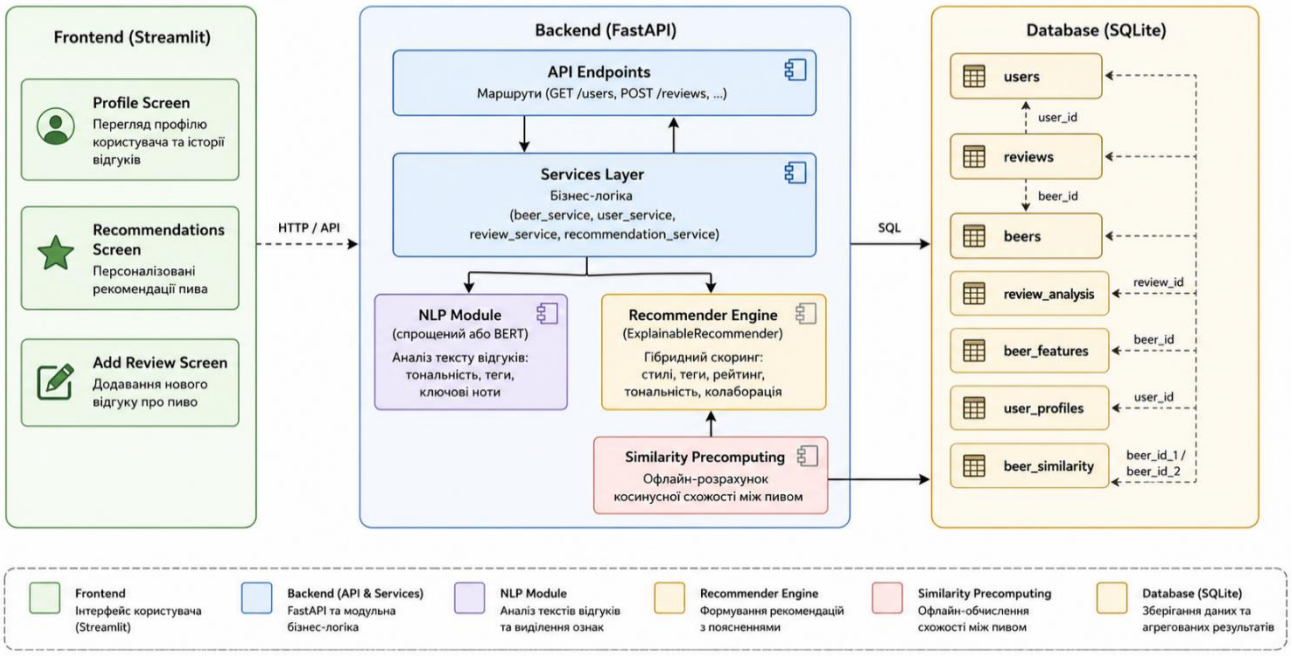


Рисунок 3.7 – Загальна архітектура системи

Допоміжні обчислювальні модулі розташовано в каталозі src, який об’єднує пакети db (ініціалізація таблиць, завантаження CSV, агрегація ознак), recommend (рекомендаційний рушій із поясненнями – explainable_engine.py) та evaluation (оцінювання якості – evaluator.py).

Backend надає набір REST-ендпоінтів, що покривають усі функціональні потреби системи (рисунок 3.8).



Рисунок 3.8 – REST API ендпоінти системи

Група ендпоінтів /users дозволяє отримати список усіх користувачів, дані конкретного користувача, його профіль (зважені стилі й теги) та статистику (кількість відгуків, середня оцінка). Ендпоінт /users/{id}/recommendations повертає персоналізований топ-N рекомендацій для заданого користувача. Група /beers надає доступ до каталогу пива: можливість отримати список із фільтром за стилем та детальну картку окремого сорту, яка містить середній рейтинг, тональність і перелік тегів. Через POST /reviews додається новий відгук; цей ендпоінт автоматично запускає NLP-обробку (визначення тональності та тегів), зберігає результати в review_analysis і ініціює перерахунок агрегованих таблиць

beer_features та user_profiles. Отримати конкретний відгук можна через GET /reviews/{id}. Уся логіка обробки запитів делегується відповідним сервісам, що забезпечує розділення відповідальності та спрощує подальше розширення системи.

Клієнтську частину реалізовано з використанням бібліотеки Streamlit, яка дозволяє створювати інтерактивні веб-інтерфейси безпосередньо на Python. Застосунок має багатосторінкову структуру: головний файл app.py відповідає за навігаційне меню, а окремі екрани розташовано в каталозі pages. До складу frontend також входять каталоги components (багаторазові візуальні елементи: чипи для тегів, індикатор тональності, картка рекомендації), services (клієнтські модулі для HTTP-запитів до backend API) та utils (допоміжні функції керування станом сесії та константи).

Система надає три основні екрани:

– екран профілю (рисунок 3.8) відображає ім’я та ідентифікатор поточного користувача й містить селектор для перемикання між усіма 50 користувачами. Улюблені стилі пива показано у вигляді прогрес-барів із зазначенням ваг, а смакові теги – у формі чипів із відсотковими значеннями. Додатково виводиться статистика: загальна кількість залишених відгуків та середня оцінка;

– екран рекомендацій (рисунок 3.9) пропонує слайдер для вибору бажаної кількості рекомендацій (від 5 до 20) та кнопку для завантаження або оновлення списку. Кожна рекомендація подається у вигляді картки, яка містить назву пива, стиль, загальний скоринговий бал, а також детальний розклад внеску окремих компонентів (стильовий, теговий, колаборативний та сентимент-скори) з текстовими поясненнями, наприклад, «відповідає вашому улюбленому стилю IPA». З картки можна перейти до детальної інформації про пиво;

– екран додавання відгуку (3.10) містить форму, в якій користувач обирає ідентифікатор пива, виставляє оцінку за допомогою слайдера та вводить текст відгуку. Після надсилання форми система відображає

результати NLP-аналізу: тональність (позитивна, нейтральна або негативна) з візуальним прогрес-баром та перелік виявлених смакових тегів у вигляді чипсів. Користувач отримує підтвердження про успішне додавання відгуку та оновлення профілю.



Рисунок 3.9 – Вигляд екрану профілю

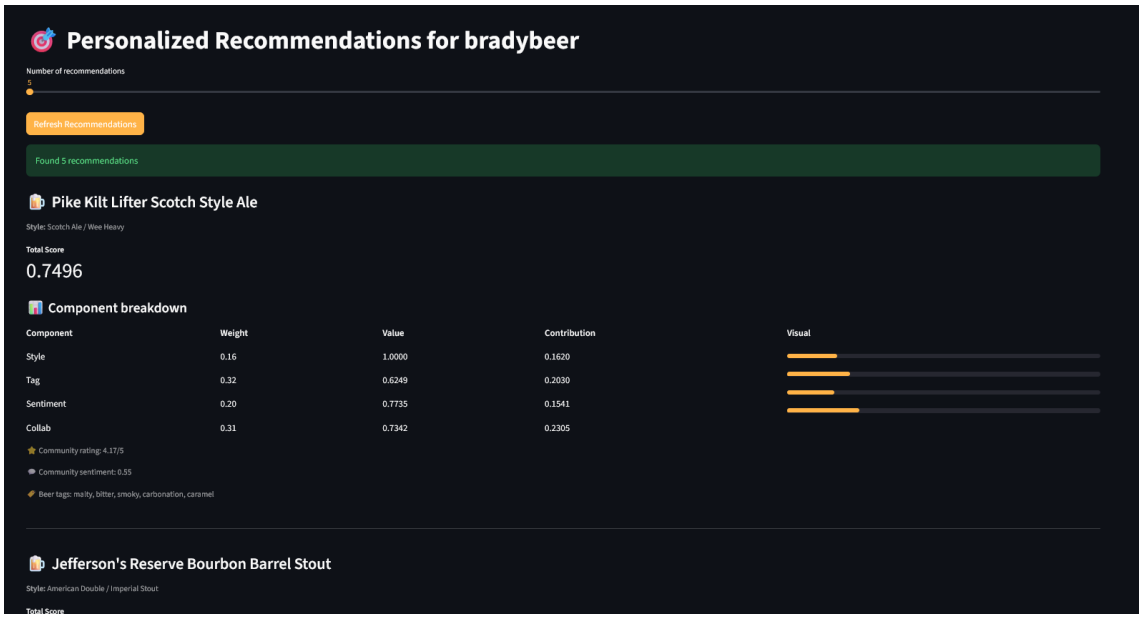


Рисунок 3.10 – Вигляд екрану рекомендацій



Рисунок 3.11 – Вигляд екрану додавання відгуків

Обмін даними між frontend і backend організовано через синхронні HTTP-запити: Streamlit-застосунок звертається до відповідних ендпоінтів FastAPI, передаючи необхідні параметри, і отримує дані у форматі JSON. Такий підхід забезпечує слабку зв’язаність компонентів: клієнтська частина не залежить від внутрішньої реалізації алгоритмів, а сервер може бути замінений або масштабований незалежно. Усі обчислення, пов’язані з машинним навчанням та NLP, виконуються на стороні сервера, що знімає навантаження з клієнтського середовища та дозволяє використовувати попередньо навчені моделі без їх перенесення.

Таким чином, архітектура системи поєднує легковажний, але функціональний інтерфейс на Streamlit із потужним бекендом на FastAPI, що реалізує весь необхідний пайплайн – від обробки природної мови до формування персоналізованих рекомендацій. Модульна структура коду полегшує тестування, супровід та потенційне розширення функціональності системи.

4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

4.1 Методика експерименту

Метою експериментального дослідження є об’єктивна оцінка якості роботи гібридної рекомендаційної системи. Оцінювання спрямоване на визначення двох ключових характеристик: точності, тобто здатності системи передбачати реальні вподобання користувача, та змістовної релевантності, яка показує, наскільки смакові профілі рекомендованих сортів пива відповідають індивідуальним уподобанням. Для цього використано спеціально підготовлену вибірку даних, процедуру leave-one-out та дві метрики: Precision@10 і Tag Overlap.

Експериментальна вибірка сформована з 20 випадково відібраних віртуальних користувачів із загальної кількості 50. Кожен із цих користувачів має не менше трьох відгуків (у середньому від п’яти до десяти), що забезпечує достатню інформацію для побудови профілю та виконання процедури оцінювання. Каталог пива складається зі 165 сортів, а загальна кількість відгуків у системі становить 1000. Оцінювання проводиться на тих самих даних, які використовуються для агрегації профілів та подібності, однак завдяки процедурі leave-one-out виключається вплив конкретного відгуку на рекомендацію, що унеможливлює некоректне завищення результатів.

Процедура leave-one-out (рисунок 4.1) є різновидом крос-валідації, адаптованим до завдань рекомендаційних систем. У контексті даного дослідження вона виконується на рівні окремих відгуків. Для кожного з 20 відібраних користувачів послідовно перебираються всі його позитивні відгуки (оцінка 4 або 5). Один такий відгук тимчасово вилучається – створюється копія бази даних без цього рядка. На основі всіх інших відгуків цього ж користувача та всіх інших користувачів система обчислює

персоналізовані рекомендації. Після цього перевіряється, чи потрапило пиво з вилученого відгуку до перших K позицій списку рекомендацій. Така перевірка повторюється для кожного позитивного відгуку, а підсумкова метрика усереднюється за всіма випробуваннями. Процедура LOO імітує реальний сценарій: система рекомендує пиво, яке користувач ще не пробував, але яке, ймовірно, оцінив би високо, і є особливо доцільною для невеликих наборів даних, де кожен відгук є цінним.

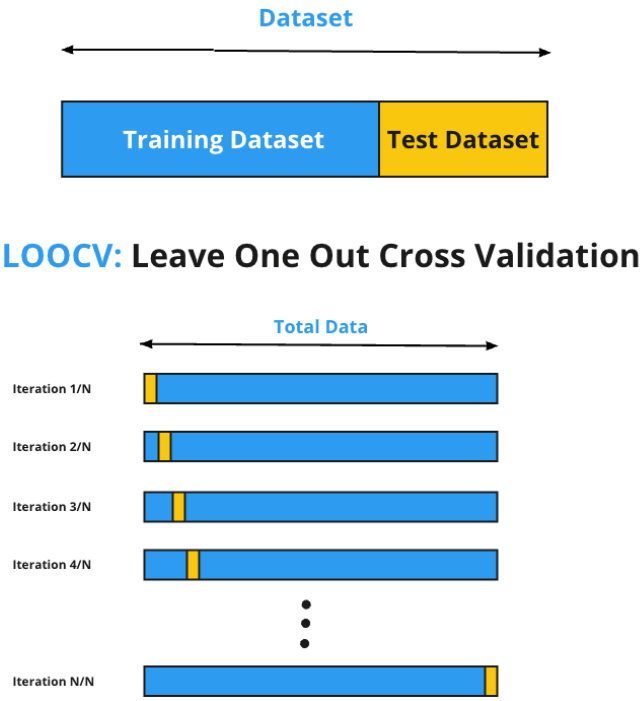


Рисунок 4.1 – Leave-one-out (LOO) [31]

Основним показником точності виступає метрика Precision@10. Вона визначається як частка релевантних (тобто високо оцінених) сортів серед десяти рекомендованих. Формально:

$$Precision@10 = (\text{кількість релевантних пив у топ} - 10) / 10. \quad (4.1)$$

Релевантним вважається те пиво, яке в реальності отримало від користувача оцінку 4 або 5. Значення K=10 обрано як типову кількість

рекомендацій, що відображається користувачеві в інтерфейсі; воно забезпечує баланс між значущістю результату та чутливістю метрики.

Другою метрикою є Tag Overlap, яка базується на коефіцієнті Жаккара (Jaccard similarity) та оцінює змістовну відповідність рекомендацій смаковому профілю користувача. Для кожного рекомендованого пива обчислюється перетин між множиною його тегів (з beer_features.tags) та множиною улюблених тегів користувача (тих, вага яких у профілі перевищує 0.3).

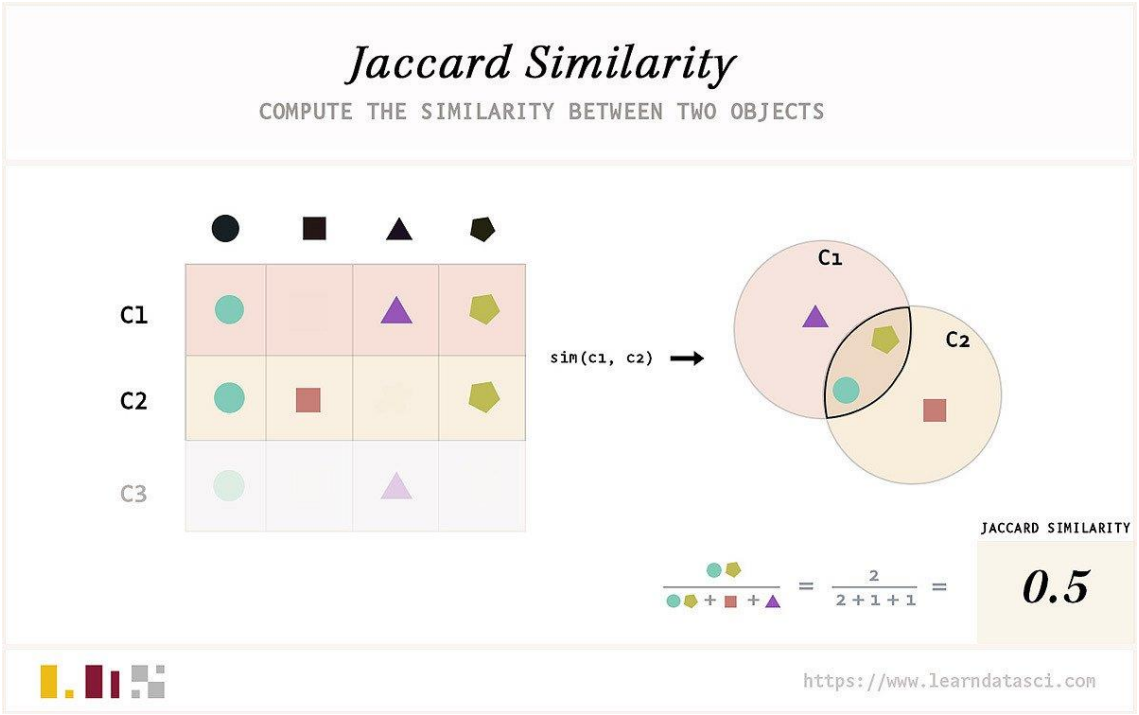


Рисунок 4.2 – Коєфіцієнт Жаккара [32]

Формально:

$$Tag\ Overlap\ =\ |A\ \cap\ B|\ /\ |A\ \cup\ B|, \tag{4.2}$$

де A – множина улюблених тегів користувача,

B – множина тегів пива.

Значення усереднюється за топ-10 рекомендацій для одного користувача, а потім – за всіма 20 користувачами. Ця метрика доповнює Precision@10, оскільки дозволяє оцінити релевантність навіть для тих сортів, які користувач не оцінював раніше, і показує, наскільки добре система використовує інформацію, отриману з текстів відгуків.

Загальну схему експерименту можна представити у вигляді такої послідовності кроків. Спочатку з 50 користувачів випадково обираються 20. Для кожного з них визначаються всі позитивні відгуки. Для кожного такого відгуку створюється копія бази даних, з якої цей відгук видалено, на основі цієї копії обчислюються топ-10 рекомендацій і фіксується факт влучання (hit). Після обробки всіх відгуків для одного користувача обчислюються індивідуальні Precision@10 та середній Tag Overlap. Процедура повторюється для всіх 20 користувачів, після чого розраховуються загальні середні значення обох метрик.

Для отримання базових показників, з якими порівнюватимуться результати подальшої оптимізації, початкові ваги п’яти компонентів гібридної формули були встановлені рівними – по 0.2 кожен. Такий рівномірний розподіл не враховує різної інформативності джерел, однак дає неупереджену точку відліку. Отримані на цьому етапі значення метрик дають змогу оцінити, наскільки подальше налаштування вагових коефіцієнтів здатне покращити якість рекомендацій.

4.2 Результати оцінювання з рівномірними вагами

Відповідно до описаної методики, на першому етапі експерименту було проведено оцінювання гібридної рекомендаційної системи з початковими ваговими коефіцієнтами, встановленими рівними для всіх п’яти компонентів: w_style = 0.2, w_tag = 0.2, w_rating = 0.2, w_sentiment = 0.2, w_collab = 0.2. Такий рівномірний розподіл не передбачає

апріорного припущення про різну значущість джерел інформації та слугує базовою точкою відліку для подальшого вдосконалення.

Середня Precision@10 за всіма користувачами склала 0.3815. Це означає, що в середньому серед десяти рекомендованих сортів пива майже чотири (3.8) виявилися такими, які користувач в реальності оцінив позитивно. Для порівняння, якби система видавала випадковий набір із 165 доступних сортів, ймовірність влучання одного релевантного пива в топ-10 була б суттєво нижчою через обмежену кількість позитивних відгуків у кожного користувача. Отже, отримане значення свідчить про наявність у системи здатності виявляти справжні вподобання, хоча абсолютний рівень точності залишає простір для покращення.

Середній Tag Overlap, обчислений за коефіцієнтом Жаккара, становив 0.4054. Це вказує на те, що множини тегів рекомендованих сортів пива та улюблених тегів користувачів перетинаються в середньому на 40%, тобто система досить послідовно пропонує пиво з близьким смаковим профілем. Помірне значення метрики пояснюється тим, що не всі рекомендації базуються виключно на тегах: колаборативна складова може пропонувати сорти, які подобалися схожим користувачам, але мають дещо інший набір смакових характеристик. Поєднання двох метрик показує, що система здатна як передбачати конкретні оцінки, так і забезпечувати певний рівень смакової відповідності.

Загалом результати базового оцінювання підтверджують працездатність гібридної архітектури: навіть без спеціального налаштування ваг система демонструє значно кращі показники, ніж випадкове вгадування. Водночас рівномірний розподіл ваг не враховує різної інформативності окремих компонентів, що обґрунтовує доцільність проведення оптимізації для підвищення точності та релевантності рекомендацій.

4.3 Оптимізація ваг гібридної формули

Попереднє оцінювання з рівномірними вагами показало, що система здатна формувати релевантні рекомендації, однак значення Precision@10 (0.3815) свідчить про значний потенціал для покращення. Оскільки гібридна формула містить п’ять компонентів, кожен із яких робить різний внесок у фінальний скоринговий бал, ручний підбір вагових коефіцієнтів є трудомістким і не гарантує оптимального поєднання. Тому було виконано автоматичну оптимізацію ваг із метою максимізації Precision@10.

Для налаштування ваг використано байєсівську оптимізацію, реалізовану бібліотекою scikit-optimize (функція gp_minimize). Цей метод будує ймовірнісну модель (гауссівський процес) залежності цільової метрики від значень вагових коефіцієнтів і на кожній ітерації обирає найбільш перспективну точку для перевірки. Такий підхід дозволяє ефективно досліджувати п’ятивимірний простір пошуку без повного перебору, для якого навіть із кроком 0.1 знадобилося б понад 160 тисяч обчислень.

Оптимізація проводилася на стратифікованій валідаційній вибірці з 10 користувачів, відібраних за кількістю відгуків так, щоб зберегти природний розподіл. Цільовою метрикою слугувала Precision@10, обчислена за процедурою leave-one-out. Оскільки оптимізатор мінімізує функцію втрат, а необхідно максимізувати точність, використовувалася функція loss = 1 – Precision@10. Простір пошуку для кожного з п’яти вагових коефіцієнтів було обмежено діапазоном [0, 1]. Початковою точкою обрано набір з ваговими коєфіцієнтами по 0,2.

У результаті оптимізації отримано вектор ваг, який після нормалізації (ділення на суму всіх коефіцієнтів) набув таких значень: style – 0.1292, tag – 0.3134, rating – 0.0302, sentiment – 0.1112, collab – 0.4162.

Відповідні частки внеску компонентів у фінальний скоринговий бал наведено в таблиці 4.1.

Таблиця 4.1 – Оптимальні нормалізовані ваги компонентів

Компонент	Вага	Внесок у score
collab	0.4162	41.6%
tag	0.3134	31.3%
style	0.1292	12.9%
sentiment	0.1112	11.1%
rating	0.0302	3.0%

Отриманий розподіл підтверджує домінуючу роль колаборативної складової (41.6%) та смакових тегів (31.3%). Разом вони забезпечують майже три чверті підсумкової оцінки. Суттєво менший внесок мають стиль (12.9%) та тональність (11.1%), тоді як середній рейтинг пива виявився практично неінформативним (3.0%). Така структура ваг узгоджується з результатами, отриманими на етапі проєктування моделі: індивідуальні смакові дескриптори та спільні патерни оцінювання між користувачами є найбільш значущими джерелами для персоналізації, тоді як усереднені показники без урахування контексту майже не додають нової інформації.

Після підстановки оптимізованих ваг у гібридну формулу було повторно обчислено метрики на тій самій валідаційній вибірці. Середня Precision@10 зросла з 0.3815 (рівномірні ваги) до 0.52, що становить відносне покращення приблизно на 36%. Зростання показника свідчить про те, що запропонована процедура оптимізації дозволила суттєво підвищити здатність системи передбачати реальні вподобання користувачів, не змінюючи при цьому саму архітектуру моделі. Отримані ваги були зафіксовані для використання у фінальному оцінюванні на незалежній тестовій вибірці.

4.4 Фінальні результати на тестових даних

Після завершення оптимізації вагових коефіцієнтів гібридної формули отриманий вектор ваг (collab – 0.4162, tag – 0.3134, style – 0.1292, sentiment – 0.1112, rating – 0.0302) було застосовано до повної тестової вибірки, що складалася з 20 користувачів, які не брали участі у валідації. Для кожного з них виконано процедуру leave-one-out на всіх позитивних відгуках, після чого обчислено середні значення Precision@10 та Tag Overlap.

Отримано такі результати:

– середня Precision@10 (LOO) – 0.41;

– середній Tag Overlap – 0.45.

Для наочності зіставлення базового та покращеного варіантів системи наведено порівняльну таблицю 4.2.

Таблиця 4.2 – Порівняння метрик до та після оптимізації ваг

Конфігурація	Precision@10	Tag Overlap
Рівномірні ваги (по 0.2)	0.3815	0.4054
Оптимізовані ваги	0.41	0.45

Аналізуючи наведені дані, можна зробити кілька важливих спостережень. По-перше, оптимізація ваг дозволила підвищити точність рекомендацій: абсолютний приріст Precision@10 склав близько 0.03, що у відносному вираженні становить приблизно 7.5%. Хоча це покращення не є радикальним, воно свідчить про те, що перерозподіл ваг на користь колаборативної та тегової складових дійсно сприяє кращому передбаченню вподобань користувачів. По-друге, спостерігається зростання показника Tag Overlap (з 0.4054 до 0.45), що вказує на підвищення смакової відповідності рекомендованих сортів профілям користувачів. Це є

закономірним, оскільки вага тегового компонента в гібридній формулі зросла з 0.2 до 0.3134.

Окремої уваги потребує питання переобучення. Під час оптимізації на валідаційній вибірці з 10 користувачів значення Precision@10 сягнуло 0.52, тоді як на незалежній тестовій вибірці з 20 користувачів воно знизилося до 0.41. Така розбіжність є класичним проявом переобучення, зумовленого малим обсягом валідаційної вибірки. Десять користувачів, навіть відібраних стратифіковано, не можуть повною мірою репрезентувати все різноманіття смакових уподобань, наявних у повному наборі даних. Оптимізатор надмірно пристосувався до особливостей саме цих 10 користувачів, що не дозволило повною мірою перенести отриманий виграш на ширшу тестову групу. Зменшення переобучення можливе шляхом збільшення валідаційної вибірки, застосування крос-валідації на кількох розбиттях або введення регуляризації в процес оптимізації. Водночас важливо зауважити, що навіть попри переобучення, оптимізована модель демонструє кращі результати порівняно з базовою, що підтверджує коректність загального напрямку налаштування.

Таким чином, фінальне тестування підтвердило працездатність системи та показало, що запропонована методика оптимізації ваг гібридної формули дозволяє покращити якість рекомендацій. Досягнуті значення Precision@10 на рівні 0.41 та Tag Overlap на рівні 0.45 свідчать про здатність системи як передбачати конкретні оцінки, так і пропонувати пиво з близьким смаковим профілем.

4.5 Аналіз якості рекомендацій та демонстрація роботи системи

Для підтвердження коректності та інтерпретованості роботи системи доцільно розглянути конкретні приклади персоналізованих рекомендацій, сформованих для різних користувачів. Нижче наведено профілі двох

користувачів із різними смаковими вподобаннями та проаналізовано відповідні їм топ-5 рекомендацій.

Перший користувач, Hex92 (рисунок 4.3), за даними його профілю, надає перевагу стилю American Pale Ale (APA), який має максимальну вагу 100%, а також цінує Rauchbier (94%) та Scotch Ale / Wee Heavy (62%). Серед смакових тегів домінують bitter (100%), malty (98%), carbonation (79%), body (71%), fruity (55%), dry (55%), balanced (54%) та creamy (52%). Користувач залишив 30 відгуків із середньою оцінкою 3.83, що свідчить про поміркованість в оцінюванні.

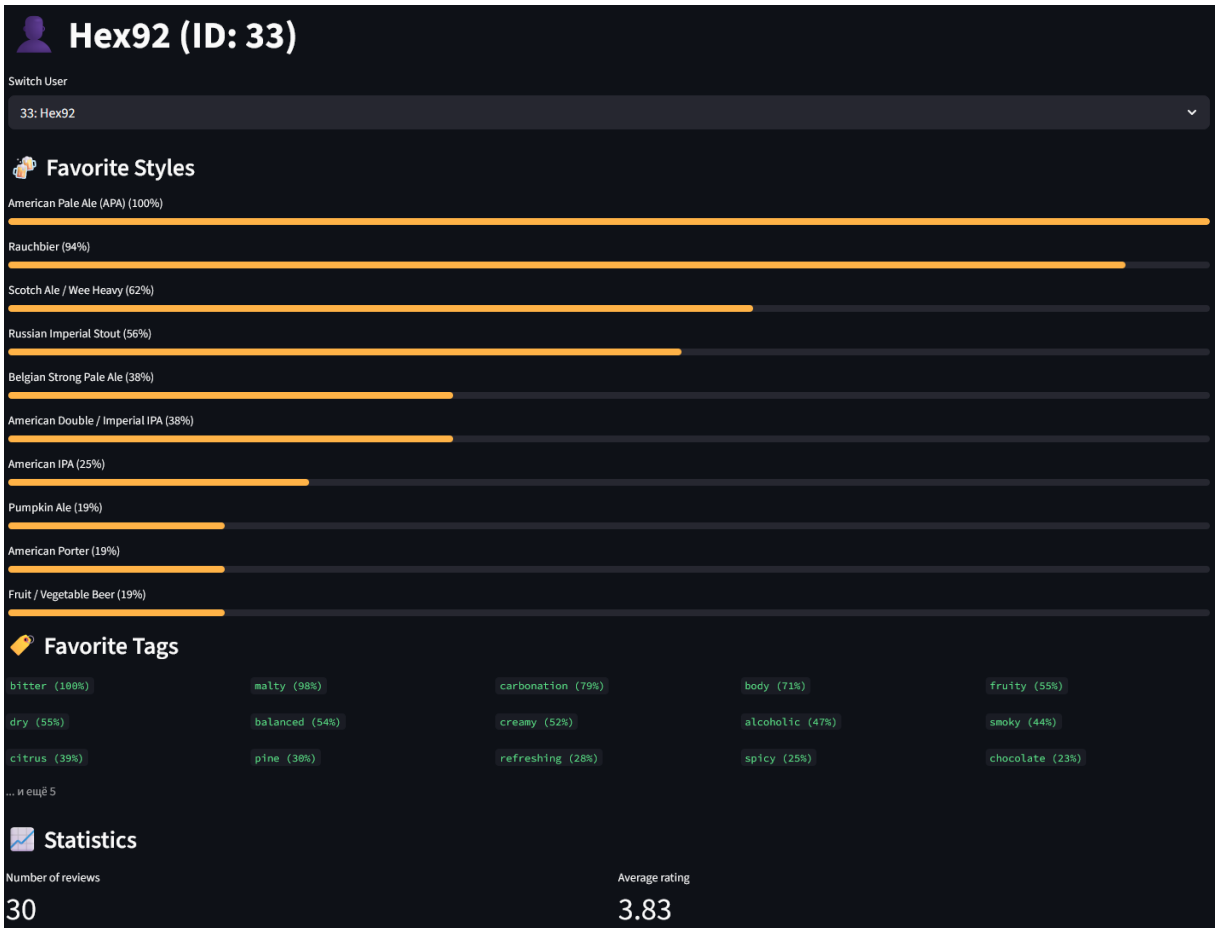


Рисунок 4.3 – Профіль користувача Hex92

Система запропонувала йому п’ять сортів пива, чотири з яких належать до стилю American Pale Ale, а один – до Rauchbier (рисунки 4.5–

4.9). Такий розподіл цілком відповідає профілю, оскільки APA є улюбленим стилем, а Rauchbier посідає другу сходинку. Загальні скорингові бали рекомендацій знаходяться у вузькому діапазоні від 0.6744 до 0.7386, що вказує на стабільність оцінок.

Найвищий бал отримало пиво Stoudts American Pale Ale (0.7386), у якого високі значення всіх компонентів: стильовий внесок – максимальний (1.0000), теговий – 0.6065, сентимент – 0.7636, колаборативний – 0.7245. Теги цього пива (bitter, citrus, malty, fruity, balanced) майже повністю збігаються з уподобаннями користувача, що пояснює лідерство.

Натомість пиво Hopdinger, хоча й належить до APA, отримало нижчий загальний бал (0.6744) через менший збіг за тегами (tag_score 0.4683) та нижчий колаборативний скор (0.6242). Це демонструє, як гібридна формула диференціює навіть сорти одного стилю, враховуючи відгуки спільноти та смакові ноти.

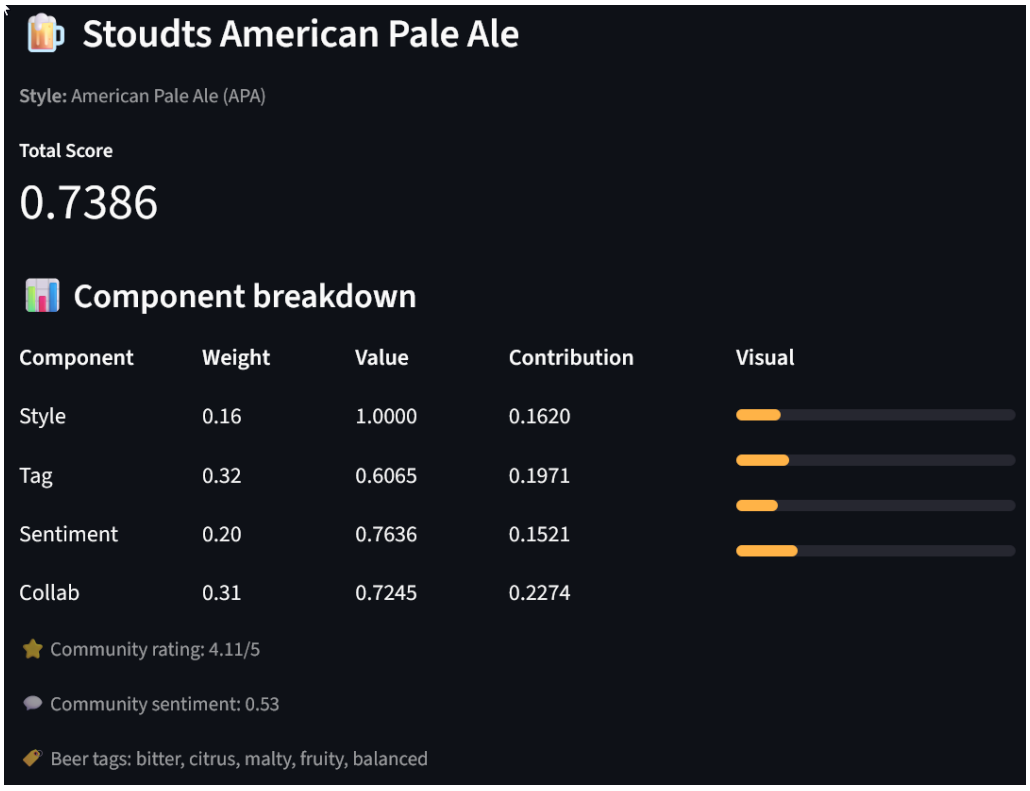


Рисунок 4.5 – Перша рекомендація для користувача Hex92

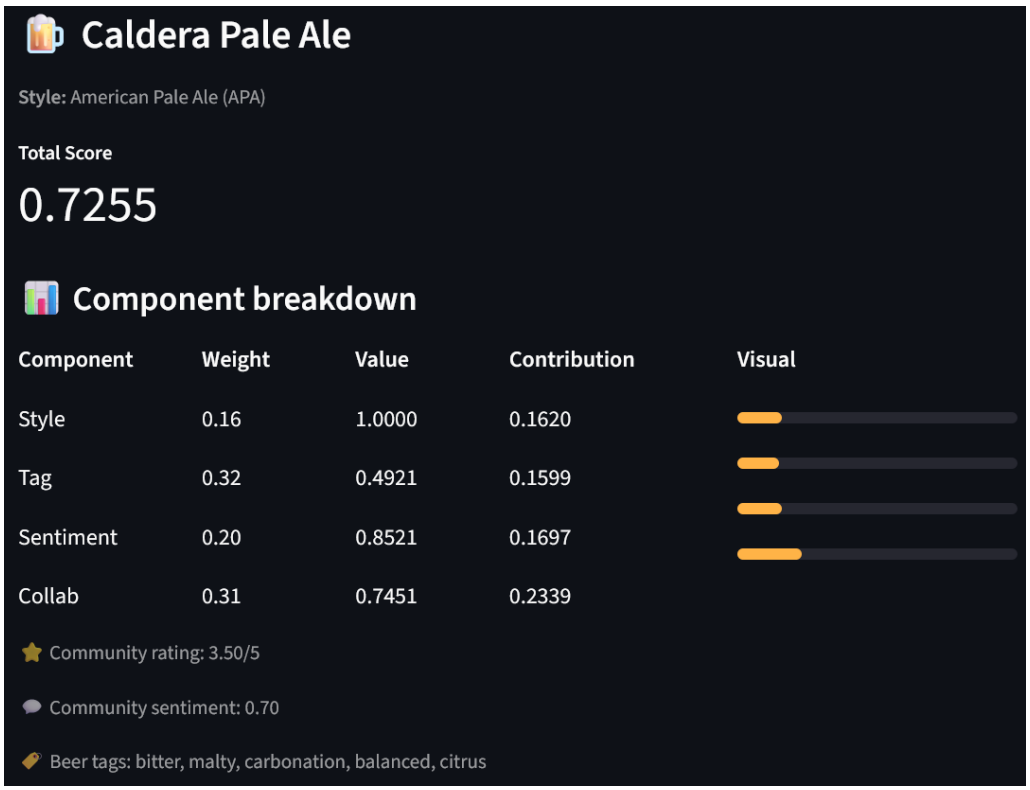


Рисунок 4.6 – Друга рекомендація для користувача Hex92

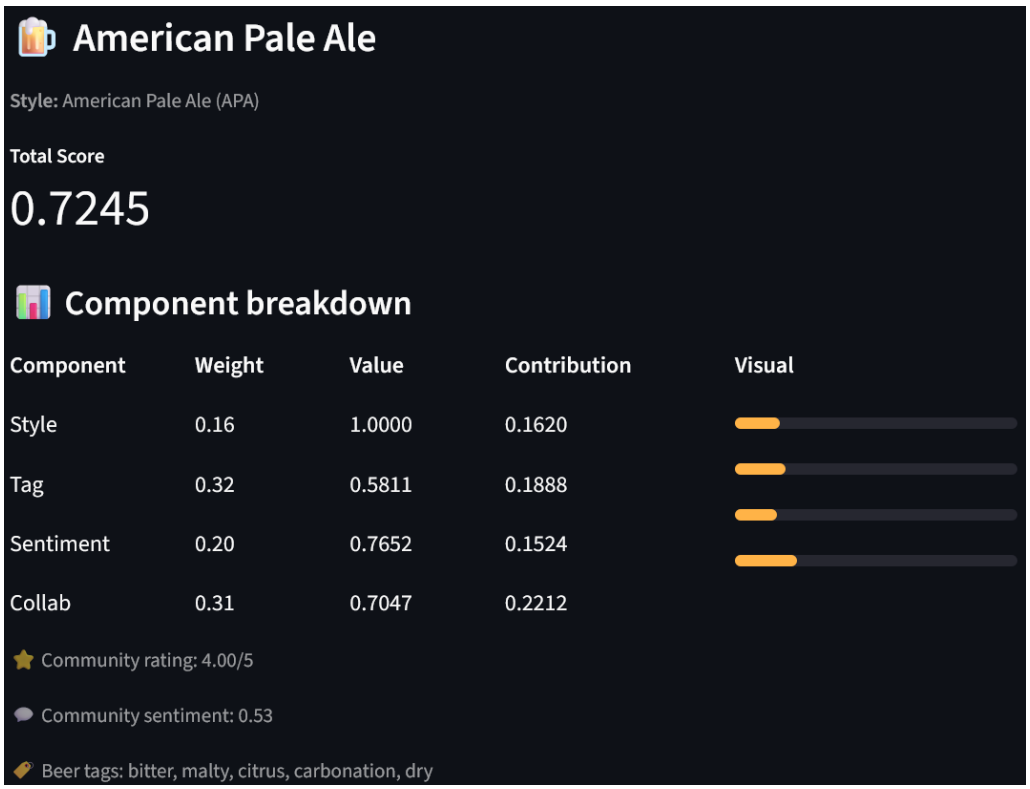


Рисунок 4.7 – Третя рекомендація для користувача Hex92

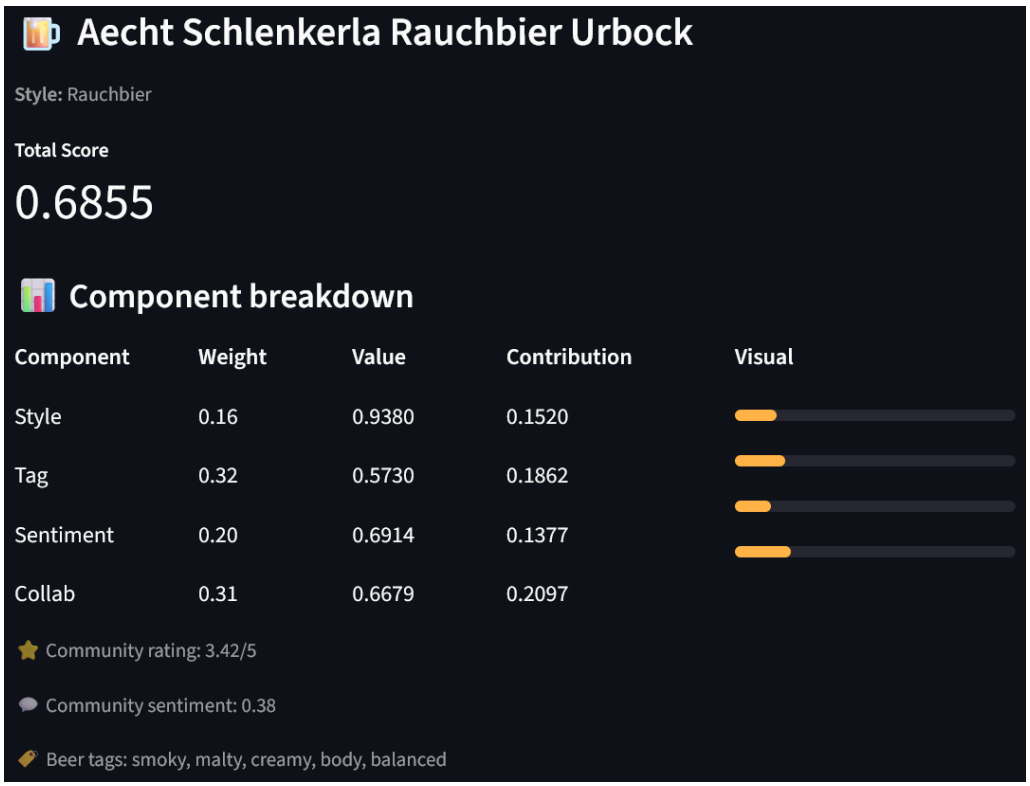


Рисунок 4.8 – Четверта рекомендація для користувача Hex92

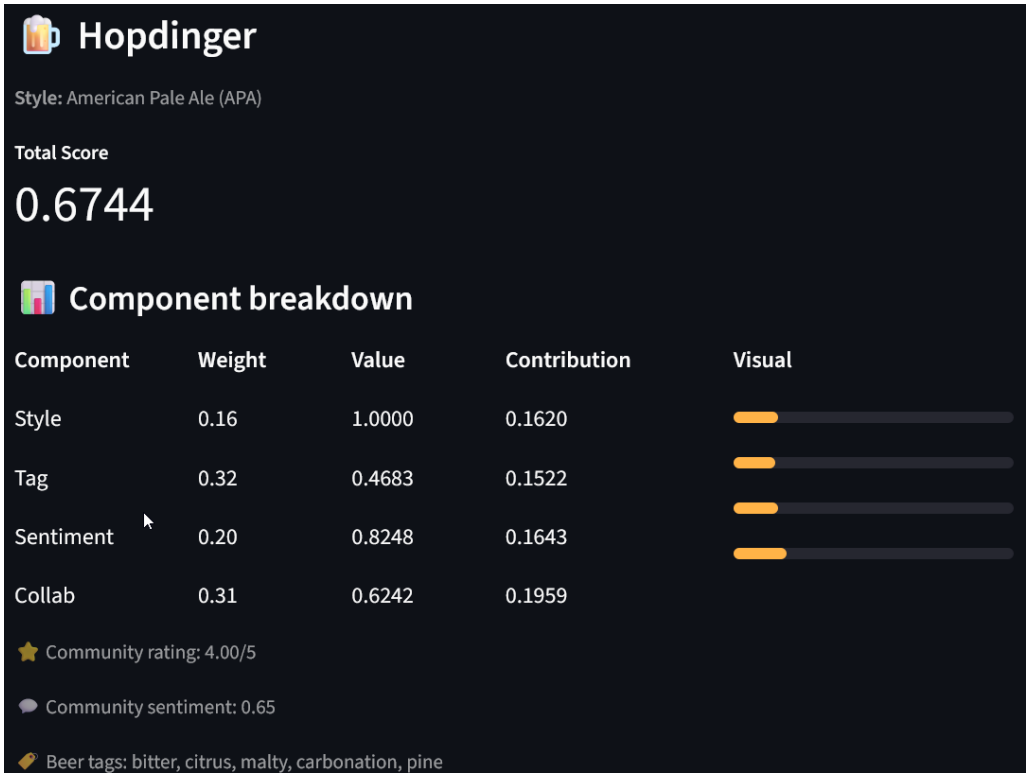


Рисунок 4.9 – П’ята рекомендація для користувача Hex92

Другий користувач, bradybeer (рисунок 4.10), має суттєво інший профіль. Його улюблені стилі – American Porter, American Double / Imperial Stout та Scotch Ale / Wee Heavy (усі з вагою 100%), а також Rye Beer (67%) та Rauchbier (56%). Домінуючі теги: malty (100%), alcoholic (65%), bitter (81%), smoky (62%), body (78%), roasty (62%), carbonation (78%), caramel (56%), creamy (71%), chocolate (46%), bourbon (28%). Користувач залишив 16 відгуків із середньою оцінкою 3.88.



Рисунок 4.10 – Профіль користувача bradybeer

Система запропонувала йому пиво (рисунки 4.11–4.15) виключно тих стилів, які відповідають його вподобанням: Scotch Ale / Wee Heavy (Pike Kilt Lifter), American Double / Imperial Stout (Jefferson's Reserve Bourbon Barrel Stout, Founders CBS Imperial Stout) та American Porter (Barrel Aged Smoke, Silk Porter).

Найвищий бал отримало Pike Kilt Lifter Scotch Style Ale (0.7598), для якого характерний високий теговий збіг (0.6250) і колаборативний скор (0.7665), а його теги (malty, bitter, smoky, carbonation, caramel) точно відповідають профілю.



Рисунок 4.11 – Перша рекомендація для користувача bradybeer

Цікавим є приклад пива Barrel Aged Smoke (Smoked Porter), яке має дуже високий сентимент (0.9715) завдяки максимальному рейтингу спільноти 5.00 та тональності 0.94, що дозволило йому піднятися на третю позицію, незважаючи на нижчий колаборативний скор (0.5000). Це демонструє, як сентимент може підсилювати рекомендації для пива з винятковими відгуками.

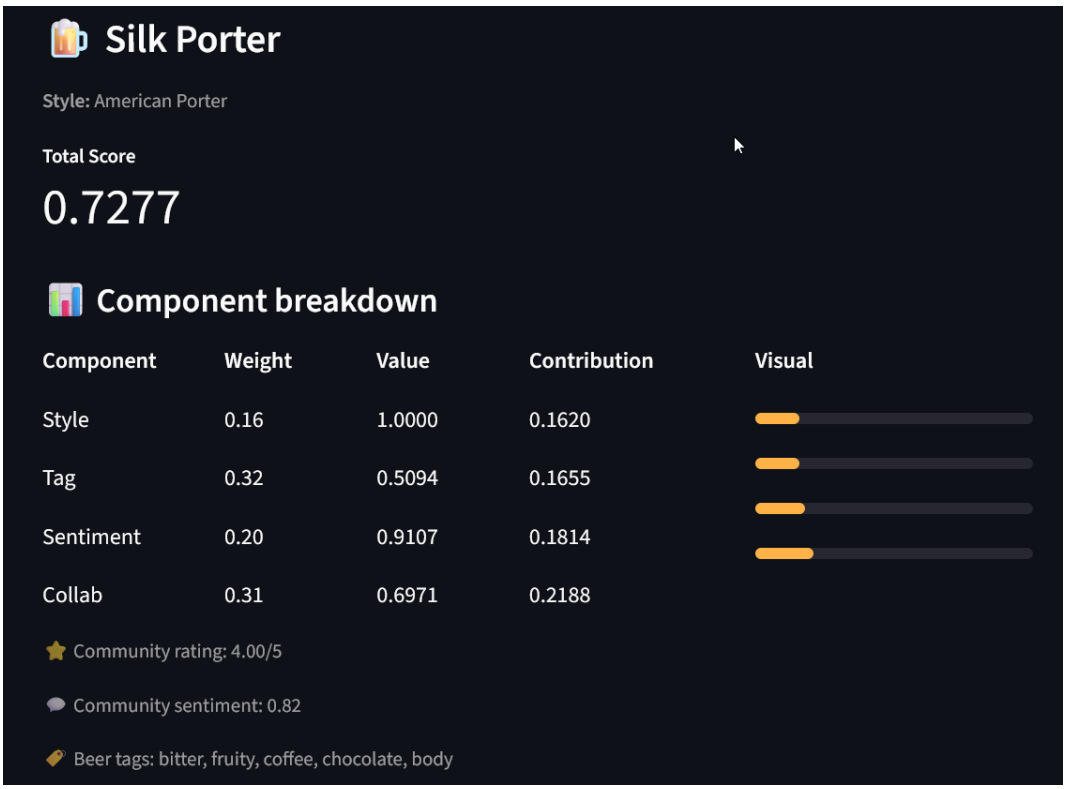


Рисунок 4.12 – Друга рекомендація для користувача bradybeer



Рисунок 4.13 – Третя рекомендація для користувача bradybeer

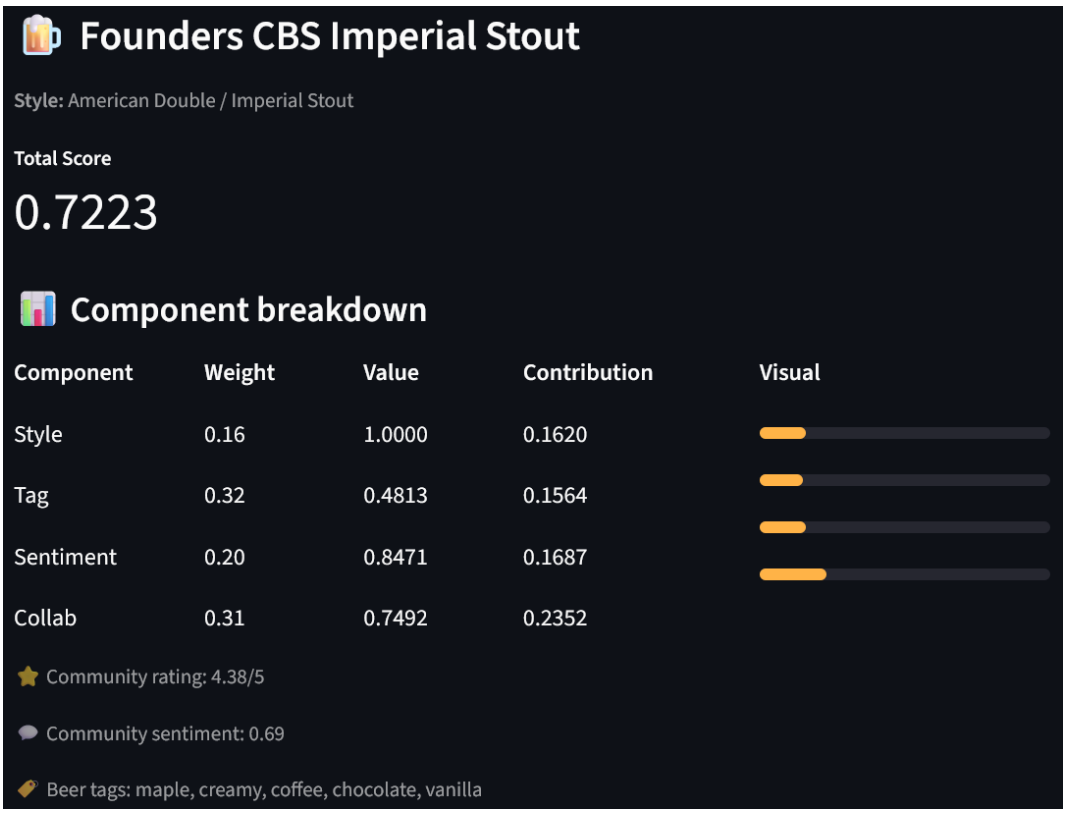


Рисунок 4.14 – Четверта рекомендація для користувача bradybeer

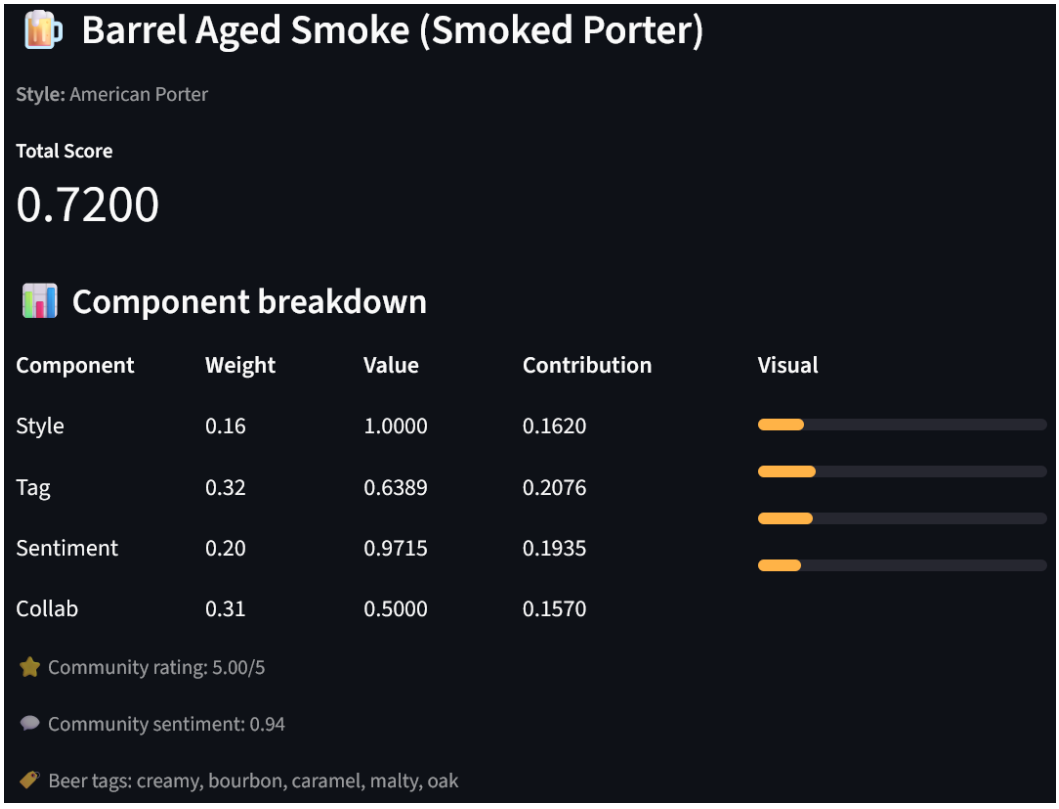


Рисунок 4.15 – П’ята рекомендація для користувача bradybeer

Порівняння рекомендацій для двох користувачів підтверджує, що система забезпечує персоналізацію: списки не перетинаються і повністю відповідають стильовим та смаковим уподобанням. Слід зазначити, що значення компонентів та внесків, показані на скріншотах, дають змогу простежити логіку формування кожної рекомендації. Це підвищує прозорість системи, що є важливою перевагою порівняно з «чорними скриньками» глибоких нейронних мереж.

Таким чином, аналіз конкретних прикладів демонструє, що система здатна генерувати релевантні та інтерпретовані рекомендації, які узгоджуються з історією оцінювання та смаковими профілями користувачів. Візуальний інтерфейс надає всю необхідну інформацію для розуміння причин тієї чи іншої рекомендації, що сприяє довірі користувачів до системи.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи було розроблено інтелектуальну інформаційну систему рекомендацій пива на основі аналізу користувацьких відгуків та методів машинного навчання.

Під час виконання роботи проведено аналіз предметної області та досліджено особливості сучасних рекомендаційних систем. Розглянуто основні підходи до формування рекомендацій, зокрема методи колаборативної фільтрації, контентно-орієнтованого підходу та гібридних моделей. Визначено переваги використання текстових відгуків як додаткового джерела інформації про вподобання користувачів.

У роботі проаналізовано сучасні методи обробки природної мови та машинного навчання, що застосовуються для аналізу користувацьких відгуків. Досліджено можливості використання текстової інформації для підвищення точності рекомендацій та формування більш повного профілю користувача.

Було спроєктовано архітектуру інформаційної системи, визначено її основні компоненти та реалізовано програмне забезпечення, що забезпечує збір, обробку та аналіз даних, а також формування персоналізованих рекомендацій. Реалізована система надає користувачам можливість отримувати рекомендації пива на основі власних оцінок, історії взаємодії та аналізу текстових відгуків.

Проведене тестування підтвердило працездатність розробленого програмного забезпечення та коректність реалізованих алгоритмів. Отримані результати демонструють можливість ефективного використання методів машинного навчання та аналізу текстових даних для побудови рекомендаційних систем у сфері вибору пивної продукції.

Практичне значення роботи полягає у створенні програмної системи, яка може бути використана в онлайн-магазинах, сервісах оцінювання напоїв та інших інформаційних платформах для підвищення якості

персоналізованих рекомендацій. Запропоноване рішення сприяє покращенню користувацького досвіду та спрощує процес вибору продукції серед великої кількості доступних варіантів.

Таким чином, мету кваліфікаційної роботи досягнуто, а поставлені завдання виконано в повному обсязі. Розроблена інтелектуальна інформаційна система рекомендацій пива може слугувати основою для подальшого розвитку та вдосконалення рекомендаційних сервісів із використанням сучасних методів штучного інтелекту та машинного навчання.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1) Beer market size, share & trends analysis report by product, by packaging, by production, by distribution channel, by region, and segment forecasts. *Grand View Research*. URL: https://www.grandviewresearch.com/industry-analysis/beer-market (date of access: 14.05.2026).

2) IWSR reports mixed results for global beverage alcohol industry in 2024. *Beverage Industry | Trends, Formulations, Technology & Products*. URL: https://www.bevindustry.com/articles/97544-iwsr-reports-mixed-results-for-global-beverage-alcohol-industry-in-2024 (date of access: 14.05.2026).

3) BarthHaas report 2024/2025. *Hops are our passion | BarthHaas*. URL: https://www.barthhaas.com/company/news/news-article/bh/barthhaas-report-2024-2025-on-the-2024-hop-harvest (date of access: 14.05.2026).

4) Tipranks. Beer industry struggles as shifting consumer preferences depress growth. *The Globe and Mail*. URL: https://www.theglobeandmail.com/investing/markets/stocks/DEO/pressreleases/32074168/beer-industry-struggles-as-shifting-consumer-preferences-depress-growth/ (date of access: 14.05.2026).

5) Mordor Intelligence. Beer – market share analysis, industry trends & statistics, growth forecasts (2025 – 2030). *Research and markets*. URL: https://www.researchandmarkets.com/reports/4536278/beer-market-share-analysis-industry-trends-and (date of access: 14.05.2026).

6) BJCP beer style guidelines – 2021 edition. *Beer Judge Certification Program – Promoting beer literacy, recognizing beer tasting and evaluation skills*. URL: https://www.bjcp.org/wp-content/uploads/2025/02/2021_Guidelines_Beer_1.25.pdf (date of access: 14.05.2026).

7) Kingfisher drinks – 2025 boring beer index. *Kingfisher Drinks*. URL: https://kingfisherdrinks.com/boringbeer2025/ (date of access: 14.05.2026).

8) Aditya N. Beer-recommendation-system: recommend beer based on customer preferences after doing sentiment analysis on beer reviews. *GitHub*. URL: https://github.com/NittalaAditya/Beer-Recommendation-System (date of access: 14.05.2026).

9) Sarovska D. Human resources recommender system based on discrete variables : master's thesis. 2021. URL: http://hdl.handle.net/10362/129696 (date of access: 14.05.2026).

10) Baillargeon J.-T., Garneau N. Beer2Vec: extracting flavors from reviews for thirst-quenching recommandations. *SSRN electronic journal*. 2022. URL: https://doi.org/10.2139/ssrn.4247241 (date of access: 15.05.2026).

11) Ni P., Li Y., Chang V. Recommendation and sentiment analysis based on consumer review and rating. *International journal of business intelligence research*. 2020. Vol. 11, no. 2. P. 11–27. URL: https://doi.org/10.4018/ijbir.2020070102 (date of access: 15.05.2026).

12) McAuley J., Leskovec J. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. *arXiv*. 2013. URL: https://doi.org/10.48550/arXiv.1303.4402 (date of access: 14.05.2026).

13) Redelinghuys J., Er Ş. SoMLier: a South African wine recommender system. *Journal of wine research*. 2023. P. 1–27. URL:https://doi.org/10.1080/09571264.2023.2184333 (date of access: 15.05.2026).

14) Kim B., Chu M., Song H. Side-Informed attention collaborative filtering for enhanced wine rating predictions. *Human-centric computing and information sciences*. 2026. Vol. 16, no. 15. URL: https://doi.org/10.22967/HCIS.2026.16.015 (date of access: 14.05.2026).

15) Koren Y., Rendle S., Bell R. Advances in collaborative filtering. *Recommender systems handbook*. New York, NY, 2021. P. 91–142. URL: https://doi.org/10.1007/978-1-0716-2197-4_3 (date of access: 15.05.2026).

16) Generating post hoc review-based natural language justifications for recommender systems / C. Musto et al. *User modeling and user-adapted*

interaction. 2020. URL: https://doi.org/10.1007/s11257-020-09270-8 (date of access: 15.05.2026).

17) Exploring the effects of natural language justifications in food recommender systems / C. Musto et al. *UMAP '21: 29th ACM conference on user modeling, adaptation and personalization*, Utrecht Netherlands. New York, NY, USA, 2021. URL: https://doi.org/10.1145/3450613.3456827 (date of access: 15.05.2026).

18) Predicting and improving complex beer flavor through machine learning / M. Schreurs et al. *Nature communications*. 2024. Vol. 15, no. 1. URL: https://doi.org/10.1038/s41467-024-46346-0 (date of access: 15.05.2026).

19) Jannach D., Adomavicius G. Recommendations with a Purpose. *RecSys '16: tenth ACM conference on recommender systems*, Boston Massachusetts USA. New York, NY, USA, 2016. URL: https://doi.org/10.1145/2959100.2959186 (date of access: 15.05.2026).

20) Aggarwal C. C. Recommender systems. Cham : Springer International Publishing, 2016. URL: https://doi.org/10.1007/978-3-319-29659-3 (date of access: 15.05.2026).

21) Gutierrez P. Deep beers: playing with deep recommendation engines using Keras. *Medium*. URL: https://medium.com/data-from-the-trenches/deep-beers-playing-with-deep-recommendation-engines-using-keras-part-1-1efc4779568f (date of access: 15.05.2026).

22) Chinchanachokchai S., Thontirawong P., Chinchanachokchai P. *A tale of two recommender systems: The moderating role of consumer expertise on artificial intelligence based product recommendations*. Journal of Retailing and Consumer Services. 2021. Vol. 61. URL: https://doi.org/10.1016/j.jretconser.2021.102528 (date of access: 15.05.2026).

23) Schmidtke M., Neeb J., Wöhner T. Collaborative filtering recommender systems: methods, strengths and weaknesses. Jena : Verlag Ernst-Abbe-Hochschule, 2025. 32 p.

URL: https://www.econstor.eu/bitstream/10419/315753/1/1923542559.pdf(date of access: 15.05.2026).

24) Users' rating predictions using collaborating filtering based on users and items similarity measures / S. Nudrat et al. *Computational intelligence and neuroscience*. 2022. Vol. 2022. P. 1–13. URL: https://doi.org/10.1155/2022/2347641 (date of access: 15.05.2026).

25) Mahmood F., Hussain N., Ben-Zaied Y. A comprehensive guide to machine learning: from introduction to model selection. *Handbook of financial econometrics, statistics, technology, and risk management*. 2025. P. 1765–1793. URL: https://doi.org/10.1142/9789819809950_0052 (date of access: 15.05.2026).

26) BERT: Pre-training of deep bidirectional transformers for language understanding / J. Devlin et al. *arXiv*. 2018. URL: https://doi.org/10.48550/arXiv.1810.04805 (date of access: 15.05.2026).

27) Neural collaborative filtering / X. He et al. *WWW '17: 26th international world wide web conference*, Perth Australia. Republic and Canton of Geneva, Switzerland, 2017. URL: https://doi.org/10.1145/3038912.3052569 (date of access: 15.05.2026).

28) DeepFM: A factorization-machine based neural network for CTR prediction / H. Guo et al. *arXiv*. 2017. URL: https://doi.org/10.48550/arXiv.1703.04247 (date of access: 15.05.2026).

29) How good your recommender system is? A survey on evaluations in recommendation / T. Silveira et al. *International journal of machine learning and cybernetics*. 2017. Vol. 10, no. 5. P. 813–831. URL: https://doi.org/10.1007/s13042-017-0762-9 (date of access: 15.05.2026).

30) Cremonesi P., Koren Y., Turrin R. Performance of recommender algorithms on top-n recommendation tasks. *The fourth ACM conference*, Barcelona, Spain, 26–30 September 2010. New York, New York, USA, 2010. URL: https://doi.org/10.1145/1864708.1864721 (date of access: 15.05.2026).

31) Khadka N. How leave-one-out cross validation (LOOCV) improve's model performance. dataaspirant.com. URL: https://dataaspirant.com/leave-one-out-cross-validation-loocv/ (date of access: 11.06.2026).

32) Karabiber F. Jaccard similarity. Learn Data Science – Tutorials, Books, Courses, and More – LearnDataSci. URL: https://www.learndatasci.com/glossary/jaccard-similarity/ (date of access: 11.06.2026).