

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Західноукраїнський національний університет**  
**Факультет комп'ютерних інформаційних технологій**  
Кафедра інформаційно-обчислювальних систем і управління

**СТАРИХ Олег Васильович**

**Інтелектуальний метод рекомендації робочих  
вакансій / Intelligent Method for Job Vacancy Recommendation**

спеціальність: 122 - Комп'ютерні науки  
освітньо-професійна програма - Комп'ютерні науки

Кваліфікаційна робота

Виконав студент групи КНм-21  
О. В. Старих

---

Науковий керівник:  
к.т.н., доцент Д.І. Загородня

---

Кваліфікаційну роботу  
допущено до захисту:

«\_\_\_» \_\_\_\_\_ 20\_\_\_ р.

В.о. завідувача кафедри

\_\_\_\_\_ Н.В. Дзюбановська

**ТЕРНОПІЛЬ - 2025**

**Факультет комп'ютерних інформаційних технологій**  
Кафедра інформаційно-обчислювальних систем і управління  
Освітній ступінь «магістр»  
спеціальність: 122 – Комп'ютерні науки  
освітньо-професійна програма – Комп'ютерні науки

ЗАТВЕРДЖУЮ  
В.о. завідувача кафедри  
\_\_\_\_\_ Н.М. Васильків  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**  
**НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**  
**СТАРИХ Олег Васильович**

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи

**Інтелектуальний метод рекомендації робочих вакансій / Intelligent Method for Job Vacancy Recommendation**

керівник роботи к.т.н., доцент Д.І. Загородня

затверджені наказом по університету від 20 грудня 2024 року № 938.

2. Строк подання студентом закінченої кваліфікаційної роботи 1 грудня 2025 р.

3. Вихідні дані до кваліфікаційної роботи: завдання на кваліфікаційну роботу студента, наукові статті, технічна література.

4. Основні питання, які потрібно розробити

- огляд предметної області та аналіз систем пошуку вакансій;
- аналіз рекомендаційних технологій та методів NLP для обробки текстів;
- постановка задачі дослідження;
- розробка інтелектуального методу рекомендацій на основі гібридної моделі;
- проектування архітектури рекомендаційної системи та бази даних;
- результати експериментальних досліджень та аналіз ефективності методу.

5. Перелік графічного матеріалу у роботі

- класифікація систем пошуку вакансій;
- схема формування векторних представлень текстів;
- загальна архітектура системи рекомендації вакансій;
- схема взаємодії модулів системи та структура бази даних;
- блок-схема алгоритму персоналізованого ранжування;
- діаграми порівняльного аналізу ефективності моделей;
- графік залежності якості ранжування від параметрів моделі.

6. Консультанти розділів кваліфікаційної роботи

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

7. Дата видачі завдання 20 грудня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів кваліфікаційної роботи	Примітка
1	Затвердження теми кваліфікаційної роботи, ознайомлення з літературними джерелами та складання плану роботи.	до 01.01. 2025 р.	
2	Написання 1 розділу кваліфікаційної роботи	до 01.03. 2025 р.	
3	Написання 2 розділу кваліфікаційної роботи	до 20.05.2025 р.	
4	Написання 3 розділу кваліфікаційної роботи	до 28.10. 2025 р.	
5	Представлення попереднього варіанту кваліфікаційної роботи, перевірка та внесення змін керівником	до 11.11.2025 р.	
6	Опрацювання зауважень та представлення завершеного варіанту кваліфікаційної роботи. Підготовка супроводжуючих документів.	до 25.11.2025 р.	
7	Перевірка кваліфікаційної роботи на оригінальність тексту.	до 01.12.2025 р.	
8	Оформлення кваліфікаційної роботи та отримання допуску до захисту	до 04.12.2025 р.	
9	Подання кваліфікаційної роботи до захисту на засіданні атестаційної комісії.	до 14.12. 2025 р.	

Студент \_\_\_\_\_ О.В. Старих  
підпис

Керівник роботи \_\_\_\_\_ к.т.н., доцент Д.І. Загородня  
підпис

## РЕЗЮМЕ

Кваліфікаційна робота на тему «Інтелектуальний метод рекомендації робочих вакансій» на здобуття освітнього ступеня «Магістр» зі спеціальності 122 «Комп'ютерні науки» освітньої програми «Комп'ютерні науки» написана обсягом в 80 сторінок і містить 21 ілюстрацію, 8 таблиць, 3 додатки та 38 використаних джерел.

Метою роботи є розробка інтелектуального методу рекомендації робочих вакансій, здатного формувати персоналізовані пропозиції на основі семантичного аналізу текстів вакансій, профілю користувача та поведінкових даних.

У межах дослідження побудовано математичну модель предметної області, запропоновано двоетапну гібридну архітектуру рекомендацій, що поєднує семантичний пошук та методи персоналізованого ранжування, а також реалізовано прототип веб-застосунка для практичного застосування методу.

Методи дослідження: веб-скрапінг і робота з API, методи обробки природної мови (NLP), інформаційний пошук (TF-IDF, BM25), семантичні векторні представлення (Sentence-BERT), алгоритми машинного навчання для ранжування, статистичні методи оцінювання якості рекомендацій.

Результати дослідження:

- удосконалено процес формування профілю користувача та векторизації текстів;
- розроблено двоетапну гібридну модель рекомендацій;
- проведено експериментальне порівняння лексичних, семантичних і гібридних методів рекомендацій.

Результати можуть бути інтегровані в онлайн-агрегатори вакансій, HR-системи або платформи кар'єрного консалтингу з метою покращення релевантності пошуку та автоматизації підбору персоналу.

Ключові слова: ІНТЕЛЕКТУАЛЬНИЙ МЕТОД, РЕКОМЕНДАЦІЙНА СИСТЕМА, ВАКАНСІЇ, СЕМАНТИЧНИЙ АНАЛІЗ, SENTENCE-BERT, АГРЕГАТОР ВАКАНСІЙ, NLP.

## ABSTRACT

Qualification work on the topic «Intelligent Method for Job Vacancy Recommendation» for Master's degree on speciality 122 «Computer Science» educational and professional program «Computer Science» is written on 80 pages and it contains 21 figures, 8 tables, 3 annexes and 38 sources.

The goal of this work is to develop an intelligent method for recommending job vacancies that can generate personalized suggestions based on semantic analysis of job descriptions, user profiles, and behavioral data.

Within the scope of the paper, a mathematical model of the subject area was constructed, a two-stage hybrid recommendation architecture combining semantic search and personalized ranking methods was proposed, and a prototype web application for practical application of the method was implemented.

Research methods: web scraping and API interaction, natural language processing (NLP) methods, information retrieval (TF-IDF, BM25), semantic vector representations (Sentence-BERT), machine learning algorithms for ranking, statistical methods for evaluating the quality of recommendations.

Research results:

- the processes of user profile construction and text vectorization were improved;
- a two-stage hybrid recommendation model was developed;
- an experimental comparison of lexical, semantic, and hybrid recommendation methods was conducted.

The results can be integrated into online job aggregators, HR systems, or career consulting platforms to improve search relevance and automate recruitment.

Keywords: INTELLIGENT METHOD, RECOMMENDATION SYSTEM, JOB VACANCIES, SEMANTIC ANALYSIS, SENTENCE-BERT, JOB VACANCY AGGREGATOR, NLP.

## ЗМІСТ

Перелік умовних позначень .....	7
Вступ.....	8
1 Аналіз інтелектуального методу рекомендацій робочих вакансій .....	11
1.1 Загальна характеристика предметної області .....	11
1.2 Класифікація систем пошуку вакансій та їхні приклади .....	12
1.3 Рекомендаційні технології та виклики їх застосування .....	17
1.4 Постановка задачі дослідження .....	21
Висновки до розділу 1 .....	22
2 Інтелектуальний метод рекомендації вакансій .....	24
2.1 Опис інтелектуального методу .....	24
2.2 Методи представлення текстів і профілів. ....	28
2.3 Архітектура рекомендаційної системи .....	32
2.4 Метод персоналізованого ранжування .....	36
2.5 Методи оцінювання ефективності рекомендаційних систем .....	40
Висновки до розділу 2 .....	44
3 Експериментальне дослідження та аналіз ефективності методу .....	46
3.1 Методологія та експериментальне середовище .....	46
3.2 Базове порівняння ефективності моделей .....	48
3.3 Детальний аналіз причин переваги гібридної моделі.....	51
3.4 Вплив параметрів гібридної моделі.....	54
3.5 Аналіз продуктивності.....	56
Висновки до розділу 3 .....	58
Висновки .....	60
Список використаних джерел .....	62
Додаток А Апробація отриманих результатів.....	66
Додаток Б Потік формування ознак для профілю користувача та вакансії ....	79
Додаток В Архітектура двоетапної гібридної моделі рекомендацій .....	80

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ANN (Approximate Nearest Neighbors) — наближений пошук найближчих сусідів.

API (Application Programming Interface) — програмний інтерфейс застосунку.

BM25 (Best Matching 25) — ймовірнісна функція ранжування, що використовується в інформаційному пошуку.

BPR (Bayesian Personalized Ranking) — байєсівське персоналізоване ранжування (метод навчання ранжуванню).

CTR (Click-Through Rate) — показник клікабельності.

HTTP (HyperText Transfer Protocol) — протокол передачі гіпертексту.

IDF (Inverse Document Frequency) — обернена частота документа.

JSON (JavaScript Object Notation) — текстовий формат обміну даними.

MAP (Mean Average Precision) — середня точність (метрика якості ранжування).

MRR (Mean Reciprocal Rank) — середній обернений ранг (метрика якості ранжування).

NDCG (Normalized Discounted Cumulative Gain) — нормований дисконтований кумулятивний приріст (метрика якості ранжування).

NLP (Natural Language Processing) — обробка природної мови.

REST (Representational State Transfer) — архітектурний стиль взаємодії компонентів розподіленої програми.

SBERT (Sentence-BERT) — модифікація мережі BERT для створення семантичних векторів речень.

TF (Term Frequency) — частота терміна в документі.

TF-IDF (Term Frequency - Inverse Document Frequency) — статистична міра для оцінки важливості слова в контексті документа.

UI (User Interface) — інтерфейс користувача.

## ВСТУП

**Актуальність теми.** Сучасний ринок праці є надзвичайно динамічним і характеризується постійним зростанням кількості вакансій, які щоденно публікуються на різних онлайн-платформах, корпоративних сайтах і в соціальних мережах. Водночас відбувається збільшення кількості претендентів на робочі місця, що посилює конкуренцію між кандидатами. У таких умовах традиційні методи пошуку роботи, засновані на ручному відборі та перегляді великого обсягу оголошень, стають малоефективними та витратними за часом. Існуючі агрегатори вакансій зазвичай обмежуються наданням інструментів фільтрації та пошуку за ключовими словами, але рідко впроваджують механізми глибокої персоналізації, здатні враховувати індивідуальні особливості користувача, його попередній досвід, навички та вподобання [4, 17, 19, 23, 36, 38].

Впровадження інтелектуальних методів у функціонал агрегаторів відкриває нові можливості для підвищення ефективності пошуку роботи. Такі підходи дозволяють автоматично аналізувати текстові описи вакансій, визначати їх релевантність для конкретного користувача, будувати персоналізовані рекомендації та ранжувати пропозиції за ступенем відповідності [5, 7, 11, 15, 30, 33, 34]. Це особливо актуально в умовах швидкої зміни ринкових вимог, коли швидкість і точність пошуку можуть суттєво вплинути на успішність працевлаштування.

**Мета та завдання дослідження.** Метою роботи є розробка та дослідження інтелектуального агрегатора вакансій, який поєднує автоматизоване збирання пропозицій з різних джерел та механізм персоналізованого ранжування, що формує індивідуально орієнтований список вакансій для кожного користувача. Для досягнення мети передбачено вирішення таких завдань [1]:

1. Виконати огляд і аналіз сучасних систем пошуку роботи та агрегаторів вакансій, визначити їхні переваги та недоліки.

2. Сформулювати вимоги до інтелектуального агрегатора, включаючи необхідні функціональні та нефункціональні характеристики.

3. Розробити архітектуру системи, що передбачає модулі збору даних, збереження та оновлення інформації про вакансії, побудови профілю користувача та рекомендаційного механізму [2].

4. Запропонувати метод персоналізації рекомендацій, який враховує як структуровані дані (локація, досвід, зарплата), так і семантичний аналіз текстових описів вакансій.

5. Реалізувати прототип веб-застосунка з інтерфейсом, що дозволяє переглядати ранжований список вакансій та переходити на оригінальні сторінки оголошень.

6. Провести експериментальні дослідження для оцінки ефективності запропонованого методу, використовуючи метрики точності, повноти та якості ранжування.

7. Виконати порівняння результатів з базовими методами пошуку і сформулювати рекомендації для подальшого розвитку системи.

**Об'єктом дослідження** є процес підбору вакансій у цифровому середовищі, що охоплює пошук, відбір і представлення пропозицій користувачеві, зокрема в межах агрегаторів вакансій.

**Предметом дослідження** є інтелектуальні методи та алгоритми формування персоналізованих рекомендацій вакансій у межах системи-агрегатора, а також архітектурні рішення, що забезпечують їх інтеграцію та ефективне використання.

**Методи дослідження.** У роботі застосовуються методи збору та обробки інформації з відкритих джерел (веб-скрапінг, API-запити), методи обробки природної мови (NLP) для аналізу текстових описів вакансій, алгоритми інформаційного пошуку та персоналізованого ранжування, а також статистичні методи для оцінки якості роботи рекомендаційних систем. Для реалізації програмного забезпечення використовуються сучасні веб-технології, реляційні та

нереляційні бази даних, фреймворки для бекенд- та фронтенд-розробки [1, 2, 14, 24, 26, 28, 29].

**Наукова новизна одержаних результатів.** Наукова новизна полягає в інтеграції механізму інтелектуального ранжування у функціонал вакансійного агрегатора з урахуванням семантичної подібності текстів вакансій і профілю користувача [3]. Запропонований підхід дозволяє поєднати переваги широкого охоплення ринку праці (через агрегацію вакансій із різних джерел) із точністю та релевантністю персоналізованих рекомендацій. У роботі передбачається удосконалення підходів до побудови профілю користувача та застосування методів аналізу тексту для глибшого виявлення відповідності вакансій [4, 5, 21].

**Практичне значення одержаних результатів.** Практичне значення полягає у створенні робочого прототипу веб-застосунку, який може бути використаний як самостійний сервіс або інтегрований у вже існуючі платформи з працевлаштування. Розроблена система дозволить значно скоротити час пошуку вакансій та підвищити точність їх підбору для користувачів. Збереження можливості переходу на оригінальні джерела вакансій робить систему сумісною з політиками більшості платформ і відкриває можливості для партнерської взаємодії з роботодавцями та агрегаторами [4, 14, 17, 19, 29].

**Публікації та апробація КР.** Результати кваліфікаційної роботи апробовані та опубліковані у матеріалах:

– Міжнародної науково-практичної конференції «Interdisciplinary approaches to solving global issues», Остін, США, 6 листопада 2025р.

– ІХ Міжнародної студентської наукової конференції «Глобалізація наукових знань: міжнародна співпраця та інтеграція галузей наук», Черкаси, Україна, 7 листопада 2025р.

Копії публікацій представлені в додатку А.

Кваліфікаційна робота складається із вступу, трьох розділів, висновків, списку використаних джерел та додатків.

# 1 АНАЛІЗ ІНТЕЛЕКТУАЛЬНОГО МЕТОДУ РЕКОМЕНДАЦІЙ РОБОЧИХ ВАКАНСІЙ

## 1.1 Загальна характеристика предметної області

Сучасний ринок праці, особливо на тлі цифрової трансформації та постійних змін в економіці, демонструє суттєве збільшення попиту на онлайн-рішення для рекрутингу. В Україні зберігається високий потенціал трудових ресурсів, зокрема серед внутрішньо переміщених осіб, які активно шукають роботу через цифрові платформи [19, 36, 38]. Це створює нові виклики та водночас відкриває можливості для впровадження інтелектуальних методів рекомендацій, що здатні підвищити ефективність підбору вакансій [11, 15, 30].

Онлайн-платформи для пошуку роботи стали ключовими точками взаємодії між кандидатами та роботодавцями. Проте, збільшення кількості вакансій та резюме ускладнило ефективний пошук релевантних пропозицій. У цьому контексті агрегатори вакансій — системи, що автоматично збирають оголошення з різних джерел (портали, корпоративні сайти, соціальні мережі) та представляють їх в уніфікованому форматі — стали важливим інструментом як для шукачів, так і для роботодавців [4, 17, 19].

Агрегатори полегшують користувачам доступ до великої кількості вакансій, дозволяють економити час і підвищують видимість пропозицій для роботодавців. Однак, навіть найпопулярніші агрегатори мають недоліки. Часто зустрічається дублювання вакансій, відсутність глибокої персоналізації під пропозиції конкретного користувача, а також необхідність переходу на зовнішні ресурси для подачі заявки. Також традиційні агрегатори найчастіше пропонують широкий перелік вакансій, але не враховують індивідуальні переваги, досвід та навички користувача - що обмежує релевантність результатів [30, 32, 34].

Водночас науковий та прикладний інтерес все більше зосереджується на рекомендаційних підходах до пошуку роботи. Це включає контентно-орієнтовані, колаборативні та гібридні системи, що дозволяють адаптувати результати під індивідуального користувача. У сфері e-recruitment існують роботи, що аналізують

виклики, методи на кшталт NLP-аналізу описів вакансій, а також поведінкові дані користувача для побудови рекомендацій [11, 15, 30, 32]. Зокрема, гібридні підходи, які інтегрують семантичні моделі та класифікаційні алгоритми, показують більшу ефективність у ранжуванні вакансій [6, 13, 37].

У контексті України актуальність таких інтелектуальних рішень додатково зростає через необхідність адаптації систем до локальних особливостей ринку — врахування специфіки IT-ринку, регіональних особливостей, мовних нюансів, поведінкових моделей користувачів тощо. Це підкреслює важливість створення систем, які об'єднують функцію агрегатора (широке охоплення вакансій) з інтелектуальним ранжуванням (персоналізація, релевантність).

Таким чином, предметна область охоплює:

- динаміку ринку праці, зростання онлайн-активності й роль агрегаторів;
- переваги агрегаторів, зокрема економія часу, широкий доступ;
- недоліки, що обмежують ефективність: дублювання, низька релевантність, відсутність персоналізації;
- напрямки розвитку, пов'язані з методами NLP, гібридним ранжуванням, аналізом поведінкових моделей.

Це обґрунтовує необхідність створення — інтелектуального методу рекомендацій робочих вакансій, який поєднує широту охоплення з глибиною персоналізованого ранжування [5, 13, 15, 30, 34].

## 1.2 Класифікація систем пошуку вакансій та їхні приклади

Системи для пошуку роботи відіграють важливу роль на сучасному ринку праці, полегшуючи взаємодію між тими, хто шукає роботу, та роботодавцями, забезпечуючи зручний доступ до вакансій та кандидатів. Їх можна поділити на три основні категорії, зважаючи на їх функціонування та охоплення: портали з вакансіями, агрегатори вакансій та корпоративні/спеціалізовані ресурси [4, 17, 19, 23, 36, 38].

### 1.2.1 Портали з вакансіями

Портали вакансій — це онлайн-платформи, де роботодавці безпосередньо публікують свої оголошення. Вони зосереджені на формуванні власної бази даних вакансій та резюме, виступаючи як посередник між кандидатами та роботодавцями. В Україні найбільш відомі приклади: Work.ua та Rabota.ua [36, 38].

Work.ua функціонує з 2006 року та є однією з найбільш відвідуваних платформ в Україні. Даний сайт пропонує різні категорії вакансій, має розвинені фільтри пошуку(за містом, зарплатою, досвідом, зайнятістю), а також дає змогу створювати резюме. На рисунку 1.1 показано головну сторінку сайту Work.ua.

Rabota.ua, заснована 2000 року і має понад 3 млн зареєстрованих резюме та пропонує роботодавцям додаткові інструменти для пошуку персоналу.

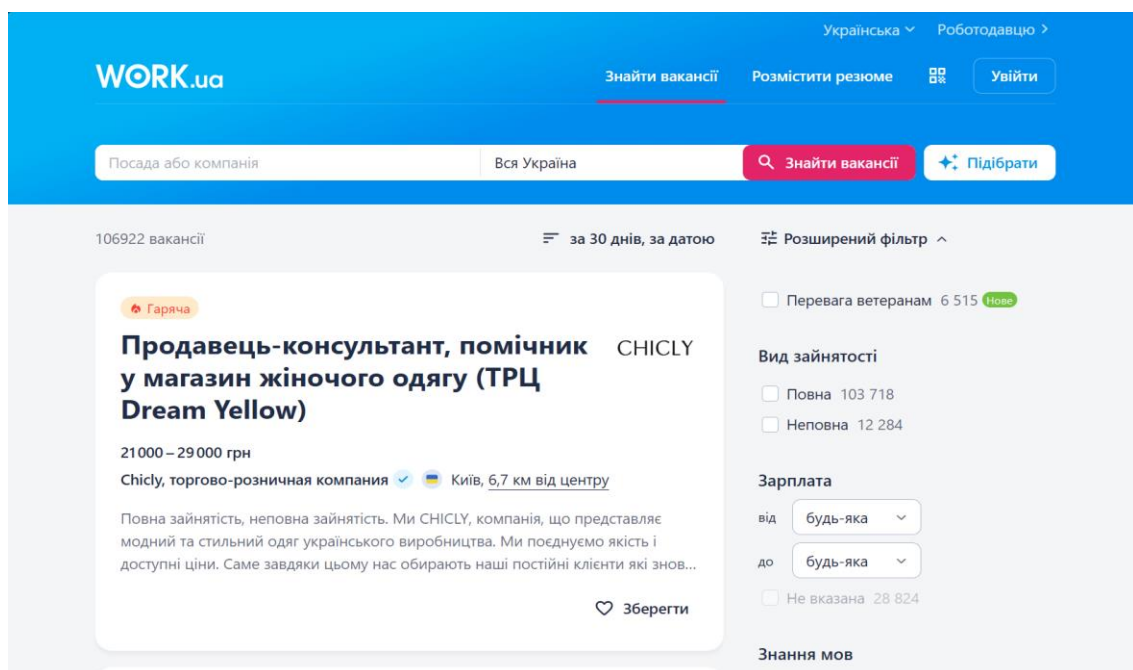


Рисунок 1.1 - Головна сторінка порталу Work.ua з пошуковим рядком і прикладами фільтрів

На міжнародному ринку популярні Indeed (понад 250 млн унікальних відвідувачів щомісяця) та Monster, які активно працюють у США, Європі та Азії [17].

Характерні риси порталів:

- можливість створити профіль та розмістити резюме;
- фільтри для пошуку;
- іноді — додаткові сервіси (тести, консультації, навчальні матеріали).

#### Переваги:

- прямий зв'язок між роботодавцем та кандидатом;
- актуальність вакансій;
- зручність використання.

#### Недоліки:

- обмежене охоплення (тільки вакансії, розміщені на порталі);
- конкуренція серед кандидатів за популярні вакансії;
- відсутність хороших механізмів персоналізації.

### 1.2.2 Агрегатори вакансій

Агрегатори вакансій — це пошукові системи, що збирають інформацію з багатьох джерел: порталів, корпоративних сайтів, соціальних мереж. Вони дозволяють користувачам переглядати вакансії з різних платформ в єдиному інтерфейсі [4, 19].

#### Відомі приклади:

– Jooble - працює більш ніж у 70 країнах світу, охоплює понад 140 000 джерел, в Україні є одним із найвідвідуваніших ресурсів для пошуку роботи. На рисунку 1.2 зображено сторінку результатів пошуку на Jooble.

– Adzuna — позиціонує себе як "Google для вакансій", поєднує пошук з аналітикою ринку праці (наприклад, середню зарплату за професією чи регіоном).

– Indeed — хоча спочатку був порталом, зараз виконує функції агрегатора, імпортуючи вакансії з різних джерел.

#### Характерні риси агрегаторів:

- пошук в реальному часі по сотнях та тисячах сайтів;
- автоматичне оновлення бази даних;
- доступ до унікальних пропозицій, яких може не бути на великих порталах.

## Переваги:

- широке охоплення ринку праці;
- економія часу на пошук;
- додаткові аналітичні інструменти

## Недоліки [11, 13, 15, 30]:

- можливе дублювання вакансій;
- різна якість даних;
- обмежені можливості персоналізації.

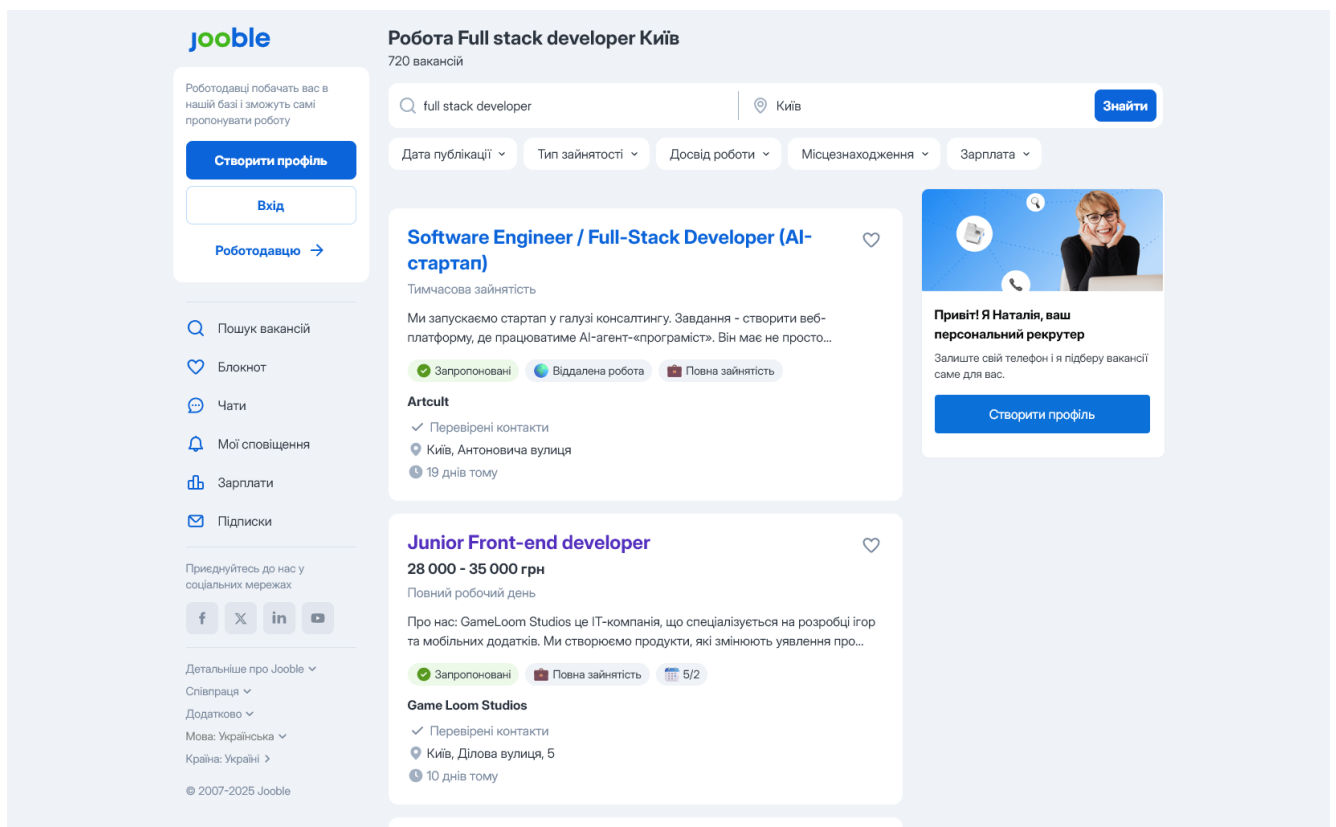


Рисунок 1.2 - Сторінка результатів пошуку на Jooble

### 1.2.3 Корпоративні та спеціалізовані ресурси

Цей тип включає кар'єрні сторінки компаній та платформи з вузькою спеціалізацією.

Найяскравіший приклад — LinkedIn — глобальна професійна мережа, що поєднує в собі пошук вакансій, нетворкінг та рекрутинг. Станом на 2025 рік тут

налічується понад 1 млрд користувачів, і понад 35 млн людей вже знайшли роботу завдяки їй [23]. Рисунок 1.3 наочно відображає, як виглядає інтерфейс пошуку вакансій у LinkedIn.

Кар'єрні сторінки компаній (наприклад, SoftServe, EPAM, Google Careers) дають можливість переглядати вакансії безпосередньо від роботодавця.

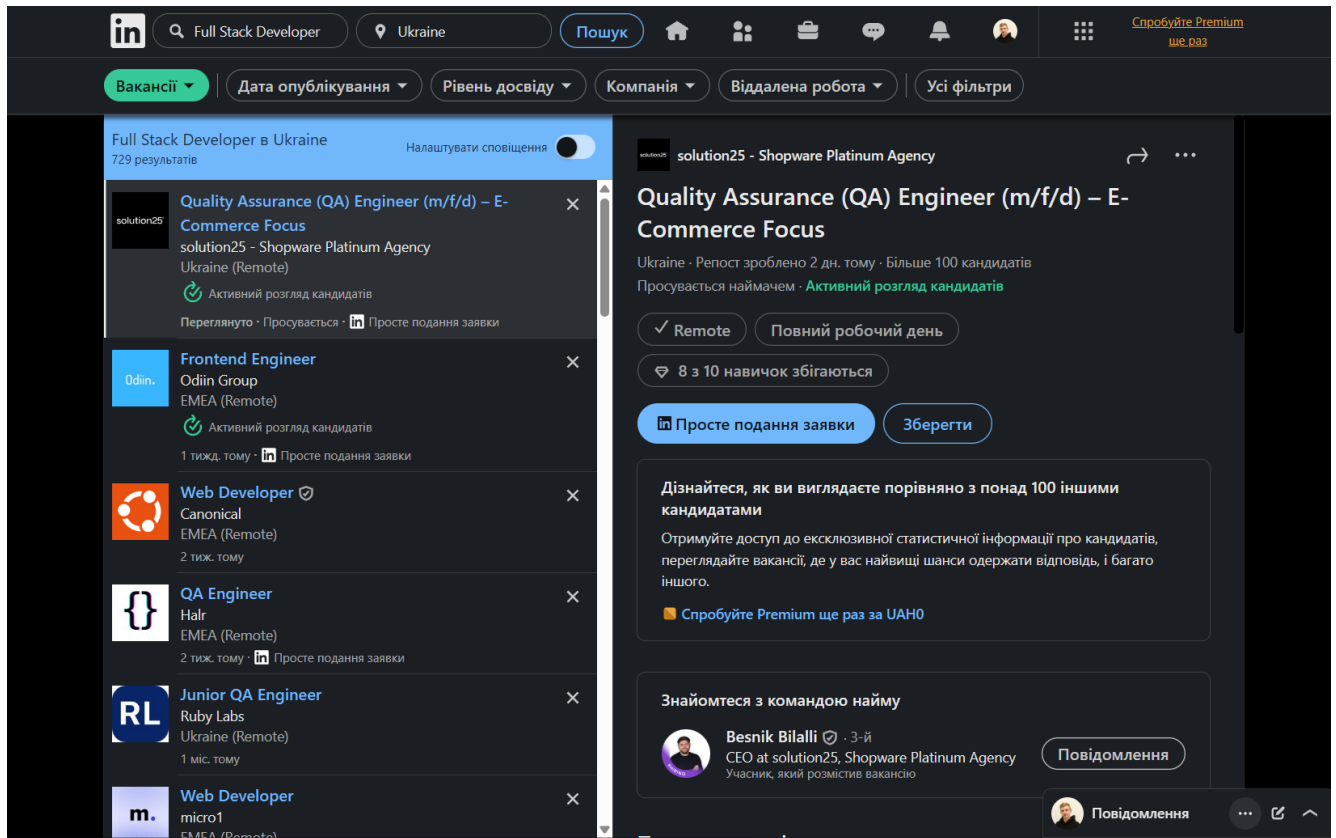


Рисунок 1.3 - Інтерфейс пошуку вакансій у LinkedIn (розділ Job Search)

Характерні риси:

- вакансії публікуються роботодавцями напряму;
- часто орієнтовані на певну галузь або нішу;

Переваги:

- висока цільова точність;
- відсутність посередників;
- додаткові можливості для професійного розвитку.

Недоліки [8, 11, 15, 30, 34]:

- обмежений набір вакансій;
- необхідність відвідувати багато сайтів;
- складність масштабного пошуку.

### 1.3 Рекомендаційні технології та виклики їх застосування

Рекомендаційні технології — ключовий елемент сучасних онлайн-платформ. Вони забезпечують швидкий доступ користувачів до актуальної інформації, що відповідає їхнім інтересам, а компаніям дозволяють покращити взаємодію з цільовою аудиторією. У сфері пошуку роботи ці технології слугують основою для інтелектуального підбору вакансій, беручи до уваги досвід, навички та професійні вподобання кандидата [11, 15, 30, 32].

#### 1.3.1 Контентно-орієнтовані підходи

Контентно-орієнтовані системи генерують рекомендації на основі аналізу описів вакансій і профілів користувачів. Алгоритми визначають ступінь подібності, аналізуючи ключові слова та характеристики (посада, вимоги, набір навичок) [35, 37]. Найчастіше використовують такі методи, як TF-IDF та косинусна міра.

На рисунку 1.4 подано процес формування рекомендацій на основі даних із профілю користувача та опису вакансії. Вхідними параметрами є ключові навички кандидата і вимоги роботодавця, що надходять на етап обчислення схожості. Результатом цього процесу є формування рекомендації вакансії, найбільш релевантної користувачу.

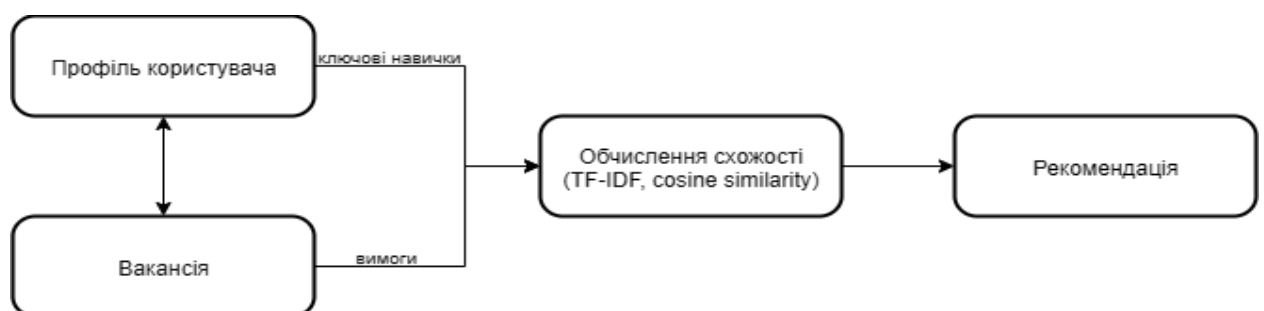


Рисунок 1.4 - Контентно-орієнтований підхід до рекомендації вакансій

Переваги: легкість впровадження, швидкість обчислень, ефективність при добре структурованих описах.

Недоліки: залежність від якості текстів вакансій, не враховуються поведінкові фактори, слабка адаптивність до зміни потреб користувача.

### 1.3.2 Колаборативна фільтрація

Колаборативна фільтрація базується на принципі: користувачі зі схожими вподобаннями часто виявляють інтерес до одних і тих же об'єктів [5]. У сфері пошуку роботи це означає, що якщо група кандидатів з аналогічними профілями реагує на вакансії певного типу, інші користувачі з подібними характеристиками отримують аналогічні рекомендації [5, 11,15].

Рисунок 1.5 відображає взаємозв'язки між користувачами та вакансіями. Суцільними лініями показано реальні дії користувачів (перегляд чи відгук), а пунктирними — рекомендації, згенеровані системою. Наприклад, якщо два користувачі цікавилися однаковою вакансією, то іншому з них може бути рекомендована вакансія, що зацікавила першого. Таким чином, система використовує колективний досвід для формування нових рекомендацій.

Перевагами є: врахування поведінкових даних, можливість надавати рекомендації навіть без повного опису вакансії.

До недоліків відносять: проблема «холодного старту» (новий користувач чи вакансія не мають історії), велика потреба у значних обсягах даних.

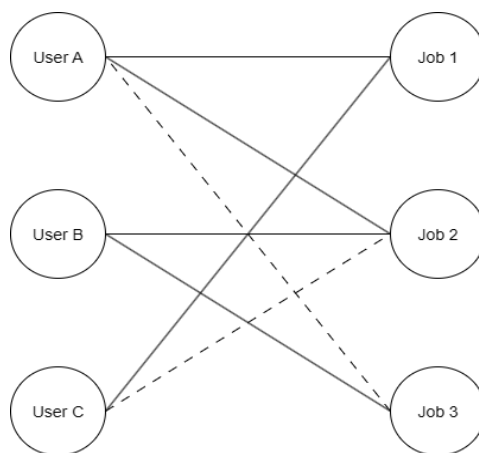


Рисунок 1.5 - Колаборативна фільтрація у системах пошуку вакансій

### 1.3.3 Гібридні підходи

Гібридні системи об'єднують переваги контентно-орієнтованих і колаборативних методів. Це дає змогу аналізувати зміст вакансій, а також враховувати поведінкові шаблони користувачів. Наприклад, алгоритм може за допомогою NLP оцінити текст вакансії, а потім скоригувати результат з урахуванням попередніх відгуків кандидата [13, 30, 34].

На рисунку 1.6 представлено блок-схему, де показано два паралельні потоки: аналіз тексту вакансій та аналіз поведінки користувачів. Обидва напрями інтегруються в єдиний модуль, який поєднує результати та формує персоналізовану рекомендацію.

Переваги: висока гнучкість, здатність подолати обмеження окремих підходів, покращена персоналізація.

Недоліки: складність реалізації, значні обчислювальні витрати.



Рисунок 1.6 - Гібридний підхід до формування рекомендацій

### 1.3.4 Використання NLP та машинного навчання

Методи обробки природної мови (NLP) дозволяють аналізувати описи вакансій не лише на основі ключових слів, а й за змістовними зв'язками. Технології word embeddings (Word2Vec, GloVe) та sentence embeddings (наприклад, SBERT) здатні виявляти семантичну подібність між текстом вакансії та профілем користувача [9, 12, 28, 31].

Машинне навчання, зокрема нейронні мережі, використовується для побудови багатовимірних моделей, які враховують професію, освіту, досвід, місце проживання, набір навичок і навіть динаміку поведінки користувачів у системі.

Переваги використання: висока точність рекомендацій, можливість комплексного аналізу даних, глибока персоналізація.

Недоліки: потреба у великих обсягах якісних даних, складність налаштування, значні обчислювальні витрати.

### 1.3.5 Виклики застосування рекомендаційних технологій

Попри численні переваги, сучасні рекомендаційні системи стикаються з рядом проблем:

- низька релевантність результатів. Простий збіг ключових слів не завжди відповідає потребам користувача;
- масштабованість. Обробка мільйонів вакансій у режимі реального часу вимагає оптимізації алгоритмів та високопродуктивних обчислювальних ресурсів;
- холодний старт. Нові користувачі чи вакансії часто не мають історії взаємодії, що ускладнює формування рекомендацій;
- якість даних. Описи вакансій часто бувають неповними, надто узагальненими або неструктурованими;
- персоналізація. Більшість агрегаторів і порталів обмежуються стандартними фільтрами, уникаючи створення глибоких моделей профілю користувача [13, 27, 30, 34].

#### 1.4 Постановка задачі дослідження

У попередніх підрозділах було розглянуто основні типи систем пошуку роботи: портали вакансій, агрегатори та корпоративні ресурси. Аналіз показав, що переважна більшість існуючих платформ сфокусовані на зборі та розміщенні вакансій, однак мають низку суттєвих недоліків. Серед них — дублювання оголошень, недостатня персоналізація, а також обмеженість пошуку, який зазвичай базується лише на ключових словах [4, 19, 36, 38]. Це суттєво ускладнює користувачам пошук саме тих вакансій, що відповідають їхнім професійним вмінням, досвіду та індивідуальним побажанням.

Огляд сучасних рекомендаційних технологій показав, що інтеграція інтелектуальних методів у системи пошуку роботи здатна значно покращити якість та відповідність результатів [11, 15, 30]. У зв'язку з цим виникає потреба в розробці інтелектуального агрегатора вакансій, що поєднує масштаби збору даних з різних джерел та гнучкість персоналізованого ранжування результатів.

Запропонований проєкт передбачає створення веб-базовану систему, що реалізує інтелектуальний метод рекомендацій робочих вакансій з такими ключовими функціональними можливостями:

- автоматизований збір вакансій з різних інтернет-джерел (портали, корпоративні сайти, соціальні мережі);
- формування рекомендацій на основі профілю користувача та аналізу описів вакансій;
- побудова персоналізованого ранжованого списку пропозицій;
- забезпечення переходу на оригінальні сторінки вакансій для подання заявки [11, 30];
- створення простого та інтуїтивно зрозумілого веб-інтерфейсу;
- можливість базового доступу без обов'язкової реєстрації (мінімальний поріг входу для нових користувачів).

Таким чином, метою кваліфікаційної роботи є розробка інтелектуального агрегатора вакансій, який поєднує функціонал збору даних із різних джерел із механізмом персоналізованого ранжування, що підвищує релевантність результатів для конкретного користувача [11, 15, 30, 32].

Для досягнення зазначеної мети необхідно виконати такі завдання:

- провести аналіз предметної області та сучасних систем пошуку вакансій;
- визначити основні функціональні та нефункціональні вимоги до системи;
- розробити архітектуру програмного комплексу, включно з модулями збору даних, збереження вакансій, побудови профілю користувача та рекомендаційного механізму [2, 24];
- створити сценарії взаємодії користувачів із системою та розробити дизайн інтерфейсу;
- реалізувати робочий прототип веб-застосунку;
- провести експериментальну оцінку ефективності рекомендаційного модуля;
- сформулювати висновки та надати пропозиції щодо подальшого розвитку системи.

## Висновки до розділу 1

1. Проведено аналіз сучасного ринку праці та встановлено, що стрімке зростання кількості онлайн-вакансій призвело до проблеми інформаційного перевантаження. Існуючі інструменти пошуку, зокрема агрегатори, хоч і забезпечують широке охоплення ринку, не повною мірою задовольняють потреби користувачів через надмірну кількість нерелевантних пропозицій та відсутність механізмів адаптації видачі під індивідуальний профіль кандидата.

2. Виконано класифікацію та порівняльний аналіз існуючих платформ пошуку роботи. Розглянуто особливості функціонування кар'єрних порталів (Work.ua, Rabota.ua), агрегаторів (Jooble, Indeed) та професійних соцмереж (LinkedIn). Виявлено їх спільний концептуальний недолік — опора на лексичний

пошук. Такий підхід вимагає точного збігу слів у запиті та описі вакансії, ігноруючи контекст і синоніми, що суттєво знижує точність пошуку.

3. Проаналізовано сучасні рекомендаційні технології та методи їх реалізації. Здійснено огляд контентно-орієнтованих, колаборативних та гібридних підходів. Встановлено, що чисті методи мають суттєві обмеження: контентні — не враховують популярність, а колаборативні — страждають від проблеми "холодного старту". Обґрунтовано, що для задачі працевлаштування найбільш ефективним є гібридний підхід, який поєднує глибокий аналіз змісту вакансій (NLP) із врахуванням поведінкових даних користувача.

4. Визначено, що головною перешкодою для підвищення якості рекомендацій є проблема "семантичного розриву" — нездатності традиційних алгоритмів розуміти, що різні слова можуть мати однакове значення. Також виокремлено необхідність розв'язання проблеми "холодного старту" для нових вакансій та забезпечення глибокої персоналізації, яка виходить за межі простих фільтрів за локацією чи зарплатою.

5. На основі проведеного аналізу визначено, що метою роботи є розробка інтелектуального агрегатора вакансій. Для досягнення цієї мети необхідно створити систему, яка використовує гібридний метод персоналізованого ранжування, що базується на семантичному аналізі текстів (для подолання semantic gap) та врахуванні контекстних факторів для підвищення релевантності результатів.

## 2 ІНТЕЛЕКТУАЛЬНИЙ МЕТОД РЕКОМЕНДАЦІЇ ВАКАНСІЙ

### 2.1 Опис інтелектуального методу

Метою даного дослідження є розробка інтелектуального методу рекомендації робочих вакансій, який дозволяє автоматично формувати персоналізовані пропозиції на основі профілю користувача, текстового опису вакансій та поведінкових даних. Для досягнення цієї мети необхідно створити формальну модель, що описує сутність об'єктів системи, взаємозв'язки між ними та математичні принципи оцінювання релевантності вакансій.

#### 2.1.1 Математичний опис задачі

Нехай  $U$  — множина користувачів, а  $V$  — множина вакансій.

Кожен користувач  $u \in U$  має профіль  $p_u$ , що містить структуровані ознаки (освіта, досвід, місце проживання, професійна сфера, очікування щодо умов праці) та текстові дані (ключові навички, цілі, опис досвіду).

Кожна вакансія  $v \in V$  описується текстом  $d_v$  (посадові обов'язки, вимоги, умови роботи) і метаданими  $a_v$  (галузь, рівень, формат роботи, зарплата, локація).

Завдання полягає у побудові функції релевантності  $s(u, v)$ , яка визначає ступінь відповідності вакансії профілю користувача. На основі цієї функції система формує впорядкований список вакансій  $\pi_u$ , у якому топ- $k$  позицій є найбільш релевантними [5, 30]. Формально,

$$s(u, v) = f(\psi_u, \phi_v) \in R, \quad (2.1)$$

де  $\psi_u$  — вектор ознак користувача,

$\phi_v$  — вектор ознак вакансії.

Ранжування визначається як:

$$\pi_u = \operatorname{argsort}_{v \in V} s(u, v), \text{ топ-}k = \pi_u[1:k], \quad (2.2)$$

Вектори  $\psi_u$  і  $\phi_v$  формуються з текстових і структурованих характеристик за допомогою методів інформаційного пошуку (TF-IDF, BM25) та семантичних векторних представлень (Sentence-BERT), що детально розглядаються у підрозділі 2.2 [9, 28, 35, 37]. У додатку Б продемонстровано процес перетворення сирих даних у векторні представлення для обчислення релевантності [9, 12, 31, 35, 37].

### 2.1.2 Двоетапна гібридна модель.

Для підвищення якості рекомендацій запропоновано двоетапну гібридну модель, що поєднує семантичний пошук і перевпорядкування за допомогою машинного навчання [6, 11, 15]. Архітектуру двоетапної гібридної моделі рекомендацій зображено в додатку В.

#### 1. Етап відбору кандидатів (Candidate Generation).

На цьому етапі застосовується двовежова модель (dual-encoder architecture): кодувальник  $E_u$  перетворює  $(p_u, I_u)$  у вектор  $z_u$ , а  $E_v - d_v, a_v$  у  $z_v$ .

Подібність між ними визначається скалярним добутком або косинусною мірою. Навчання здійснюється контрастивною функцією втрат InfoNCE, яка максимізує схожість позитивних пар (користувач-вакансія) та мінімізує подібність до негативних. Результатом цього етапу є набір із 100-300 потенційно релевантних вакансій (Candidate Set).

#### 2. Етап перевпорядкування (Re-ranking).

На цьому етапі виконується точне ранжування вакансій із набору кандидатів. Для кожної пари  $(u, v)$  формується вектор ознак:

$$x_{uv} = [ \text{sim}_{\text{emb}}(u, v), \text{BM25}(u, v), g_{\text{struct}}(u, v), h_{\text{beh}}(u, v), \rho(v) ], \quad (2.3)$$

де BM25 — контентна подібність;

$g_{\text{struct}}$  — відповідність структурних характеристик (локація, рівень, зарплата);

$h_{\text{beh}}$  — поведінкові ознаки (частота кліків, відгуків, взаємодій користувача);

$\rho(v)$  — часовий коефіцієнт актуальності вакансії.

Підсумкове значення релевантності обчислюється як:

$$s(u, v) = F_{\theta}(x_{uv}), \quad (2.4)$$

де  $F_{\theta}$  — лінійна або бустингова модель для ранжування [22].

Такий підхід дозволяє поєднати семантичну подібність, поведінкові сигнали та контекст користувача [6, 9, 11, 12, 15, 28, 30, 31].

### 2.1.3 Навчальна мета (Pairwise Learning)

Для навчання системи використовується парне навчання (Bayesian Personalized Ranking, BPR) [32]. Його мета — забезпечити, щоб для кожного користувача  $u$  позитивна вакансія  $v^+$  мала вищий рейтинг, ніж негативна  $v^-$ .

$$L = \sum_u \sum_{v^+, v^-} \log(1 + e^{-(s(u, v^+) - s(u, v^-))}) + \lambda \|\theta\|^2, \quad (2.5)$$

де  $\lambda$  — коефіцієнт регуляризації,

$\theta$  — параметри моделі.

Цей підхід широко застосовується у промислових рекомендаційних системах (LinkedIn, Indeed, Glassdoor).

### 2.1.4 Модель даних і припущення

У розробленій системі використовується три типи даних:

– текстові — описи вакансій і резюме кандидатів, які аналізуються за допомогою TF-IDF, BM25 та Sentence-BERT;

– структуровані — локація, рівень досвіду, формат роботи, заробітна плата;

- поведінкові — кліки, перегляди, збереження вакансій, відгуки (implicit feedback).

Основні припущення моделі:

1. Семантично схожі тексти мають близькі векторні представлення.
2. Релевантність залежить не лише від тексту, а й від структури та часу публікації.
3. Поведінкові дані допомагають уточнити інтереси користувача.
4. Модель повинна масштабуватись для великої кількості користувачів і вакансій у реальному часі. Ключові показники, що будуть використані для оцінювання якості ранжування та релевантності сформованих рекомендацій, систематизовано в таблиці 2.1.

Таблиця 2.1 - Основні метрики оцінювання якості рекомендаційних систем

<b>Метрика</b>	<b>Коротке означення</b>
Precision@k	Частка релевантних вакансій серед перших k результатів для користувача u.
Recall@k	Частка знайдених релевантних вакансій серед усіх релевантних для користувача u.
MRR	Середнє значення 1/позиція першої релевантної вакансії у списку для кожного користувача.
MAP	Середнє значення Precision по кожній релевантній позиції, усереднене по користувачах.
NDCG@k	Нормована сума виграшів зі зменшенням за позицією: $DCG@k / IDC@k$ , чутлива до порядку.

## 2.2 Методи представлення текстів і профілів

Ефективність будь-якої рекомендаційної системи значною мірою залежить від того, наскільки точно вона здатна представити текстову інформацію у числовій формі. У випадку систем пошуку роботи текстові дані — це описи вакансій, резюме кандидатів, переліки навичок, посадових обов’язків тощо. Щоб зробити такі тексти придатними для автоматизованої обробки, необхідно перетворити їх у векторні представлення, що відображають значення слів і семантичні зв’язки між ними [9, 12, 31, 35, 37].

Процес подання тексту у вигляді вектора називається векторизацією, і саме він є основою контентного аналізу в системах рекомендацій.

Векторизація повинна забезпечувати:

- інваріантність до довжини документів;
- збереження ключового змісту та контексту;
- можливість порівняння текстів за мірою подібності;
- масштабованість при роботі з великими обсягами вакансій.

### 2.2.1 Метод TF-IDF

Одним із найпоширеніших класичних підходів для подання текстів є TF-IDF (Term Frequency - Inverse Document Frequency) [37]. Цей метод базується на припущенні, що найважливішими є ті слова, які часто зустрічаються в одному документі, але рідко — у всій колекції документів.

Формула обчислення має вигляд:

$$\text{TFIDF}(t, d, D) = \text{TF}(t, d) \times \text{IDF}(t, D), \quad (2.6)$$

де  $\text{TF}(t, d) = \frac{n_{t,d}}{\sum_k n_{k,d}}$  — частота терміна  $t$  у документі  $d$ ;

$\text{IDF}(t, D) = \log \frac{N}{df(t)}$  — обернена частота документа, де  $N$  — загальна кількість документів, а  $df(t)$  — кількість документів, що містять термін  $t$ .

TF-IDF дозволяє оцінити, наскільки унікальне слово є значущим для конкретного документа (наприклад, опису вакансії). Результатом є векторна матриця розміром  $m \times n$ , де  $m$  — кількість документів,  $n$  — кількість унікальних термінів. На рисунку 2.1 показано схему формування матриці TF-ID.

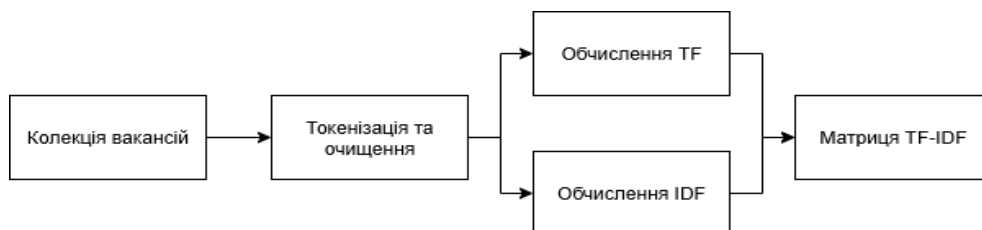


Рисунок 2.1 - Схема формування матриці TF-IDF для колекції вакансій

### 2.2.2 Метод BM25

Метод BM25 (Best Matching 25) є вдосконаленням TF-IDF і застосовується у більшості сучасних пошукових систем (у тому числі в Elasticsearch) [10, 35]. BM25 враховує довжину документа та дає змогу зменшити вплив дуже частих слів.

Релевантність між запитом  $q$  та документом  $d$  обчислюється за формулою:

$$BM25(q, d) = \sum_{t \in q} IDF(t) \cdot \frac{f(t, d) \cdot (k_1 + 1)}{f(t, d) + k_1 \cdot \left(1 - b + b \cdot \frac{|d|}{avgdl}\right)}, \quad (2.7)$$

де  $f(t, d)$  — частота слова  $t$  у документі  $d$ ;

$|d|$  — довжина документа;

$avgdl$  — середня довжина документів.

Параметри  $k_1=1.5$  і  $b=0.75$  контролюють насичення та нормалізацію за довжиною. Вплив параметрів зображено на рисунку 2.2.

BM25 показує вищу точність у порівнянні з TF-IDF завдяки врахуванню контексту документа. У системі рекомендацій вакансій він дозволяє коректно ранжувати результати навіть при різній довжині описів вакансій.

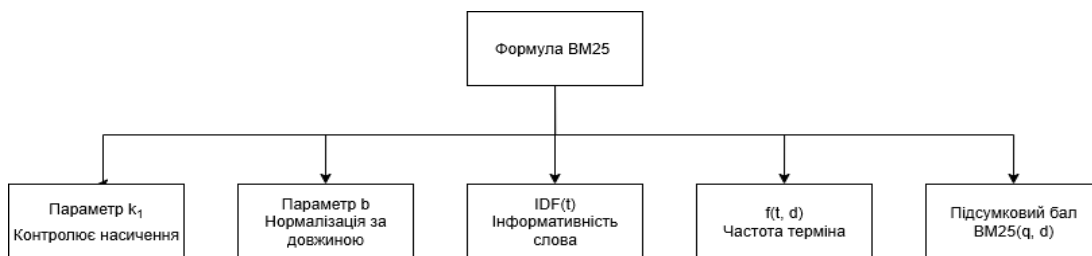


Рисунок 2.2 - Вплив параметрів  $k_1$  та  $b$  у формулі BM25 на оцінку релевантності вакансій

### 2.2.3 Семантичні векторні моделі

Хоча TF-IDF та BM25 є ефективними для базового пошуку, вони не враховують семантичні зв'язки між словами. Наприклад, терміни «developer» і «programmer» мають близьке значення, але ці методи трактують їх як різні. Щоб усунути цю проблему, використовуються семантичні моделі подання текстів, такі як:

- Word2Vec — моделює слова на основі контексту в реченнях;
- FastText — розширення Word2Vec, що враховує морфологію слів, зокрема для української мови;
- Sentence-BERT — сучасна нейронна модель для створення векторів речень і текстів, що зберігає їх семантику [9, 12, 28, 31]. На рисунку 2.3 показано порівняння базових (TF-IDF, BM25) та семантичних (Sentence-BERT) підходів.

Семантичні представлення дозволяють обчислювати подібність між вакансіями та профілями за допомогою косинусної міри:

$$\text{sim}(u, v) = \frac{\psi_u \cdot \phi_v}{\|\psi_u\| \cdot \|\phi_v\|}, \quad (2.8)$$

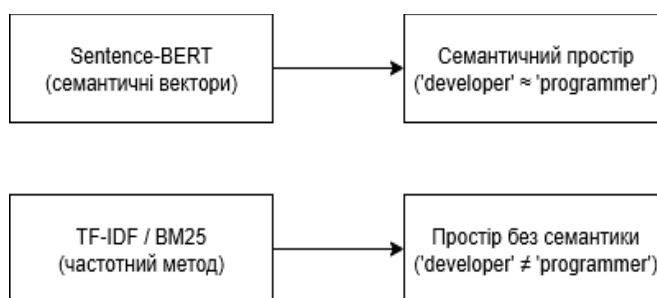


Рисунок 2.3 - Порівняння базових (TF-IDF, BM25) та семантичних (Sentence-BERT) підходів

### 2.2.4 Формування векторного профілю користувача

Оскільки користувач може мати кілька джерел інформації (резюме, навички, історія пошуку, відгуки), необхідно сформувати єдиний векторний профіль користувача [11, 13, 15, 30]. Це досягається шляхом агрегування векторів окремих документів або фрагментів:

$$\psi_u = \frac{1}{|S_u|} \sum_{d_i \in S_u} \phi(d_i), \quad (2.9)$$

де  $S_u$  — набір текстових елементів користувача, а  $\phi(d_i)$  — векторне представлення кожного елемента.

Результуючий вектор  $\psi_u$  відображає усереднений зміст профілю користувача і може бути використаний для обчислення подібності з вакансіями. Практичну реалізацію цього підходу, яка демонструє трансформацію різнорідних текстових даних (резюме, навички, пошукові запити) у єдине векторне представлення, проілюстровано на прикладі в таблиці 2.2.

Таблиця 2.2 - Приклад побудови профілю користувача з набору текстів.

Джерело даних	Приклад тексту / ключові слова	Векторне подання (усереднене значення $\phi(d_i)$ )
Резюме	Python, Django, REST API, SQL	[0.42, 0.57, 0.61, 0.48, ...]
Навички	Machine Learning, NLP, TensorFlow	[0.51, 0.63, 0.55, 0.59, ...]
Пошукові запити	Python developer remote	[0.46, 0.53, 0.58, 0.50, ...]
Вектор користувача $\psi(u)$	—	[0.46, 0.58, 0.58, 0.52, ...]

### 2.2.5 Порівняльна характеристика методів подання тексту

Для узагальнення розглянутих підходів у таблиці 2.3 наведено їхні основні характеристики за критеріями точності, швидкодії, масштабованості та здатності відображати семантику [9, 12, 21, 31].

Таблиця 2.3 - Порівняння методів векторизації текстів у системах пошуку вакансій

Метод	Точність	Швидкодія	Семантична релевантність	Масштабованість
TF-IDF	Середня	Висока	Низька	Висока
BM25	Висока	Висока	Низька	Висока
Word2Vec	Висока	Середня	Середня	Середня
FastText	Висока	Середня	Середня-Висока	Середня
Sentence-BERT	Найвища	Середня	Висока	Середня-Низька

### 2.3 Архітектура рекомендаційної системи

Розробка інтелектуальної системи рекомендації вакансій передбачає створення архітектури, яка забезпечує збирання, обробку, зберігання та аналіз великих обсягів інформації. Основна мета архітектури — простота, надійність та масштабованість, що дозволяють системі стабільно функціонувати навіть за обмежених ресурсів [11, 15, 30, 32].

#### 2.3.1 Загальна архітектура системи

Система побудована за модульним принципом і складається з таких основних компонентів (рисунок 2.4):

##### 1. Модуль збору вакансій (Data Collector).

Виконує автоматичний збір вакансій із відкритих джерел: кар'єрних порталів, корпоративних сайтів та API-платформ. Отримані дані зберігаються у форматі JSON для подальшої обробки.

##### 2. Модуль обробки та нормалізації текстів (Text Processor).

Виконує очищення, токенізацію, лематизацію та нормалізацію текстів. Після обробки текстові дані перетворюються у векторні представлення за допомогою TF-IDF, BM25 або Sentence-BERT [9, 12, 31, 35, 37].

##### 3. База даних (MongoDB)

Використовується як єдине сховище для текстових, структурованих і векторних даних. MongoDB підтримує нативний векторний пошук (`$vectorSearch`), що дозволяє зберігати `embeddings` та виконувати швидкий пошук схожих вакансій без додаткових зовнішніх модулів [24].

#### 4. Модуль рекомендацій (Recommendation Engine).

Реалізує двоетапний гібридний підхід (Candidate Generation → Re-ranking), описаний у підрозділі 2.1.

На першому етапі з MongoDB вибираються вакансії з найбільшою косинусною схожістю з профілем користувача; на другому — виконується ранжування результатів за допомогою моделі машинного навчання, яка враховує поведінкові та контекстні ознаки [6, 11, 12, 15, 30, 32].

#### 5. Інтерфейс користувача та API (Web UI / REST API).

Забезпечує зручний пошук, перегляд і отримання рекомендацій. Фронтенд реалізований на React, а API-запити передаються у форматі JSON [14, 24].

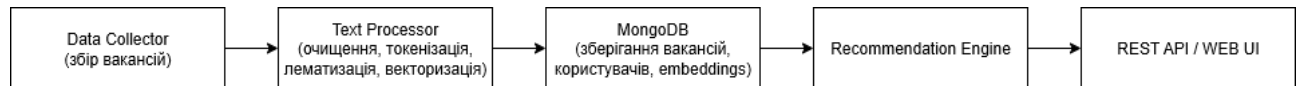


Рисунок 2.4 - Загальна архітектура системи рекомендації вакансій

### 2.3.2 Взаємодія між модулями

Взаємодія компонентів системи здійснюється через HTTP-запити та обмін JSON-повідомленнями. Основні потоки даних (рисунок 2.5) включають:

1. Збір вакансій. Data Collector формує єдину колекцію `vacancies_raw` у MongoDB.

2. Попередня обробка. Text Processor бере нові документи, виконує очищення й додає векторне поле `embedding`.

3. Збереження структурованих даних. Після обробки створюються основні колекції `users`, `vacancies` і `logs`.

4. Формування рекомендацій. Recommendation Engine отримує профіль користувача, виконує векторний пошук у колекції vacancies, проводить Re-ranking і повертає результати через REST API [11, 15, 24, 30].



Рисунок 2.5 - Взаємодія між модулями системи на основі MongoDB

### 2.3.3 Структура бази даних

MongoDB дозволяє зберігати дані у вигляді документів формату JSON, тому вся інформація про користувачів і вакансії може бути представлена у зручній і гнучкій формі [24]. На рисунку 2.6 можна побачити структуру бази даних у MongoDB.

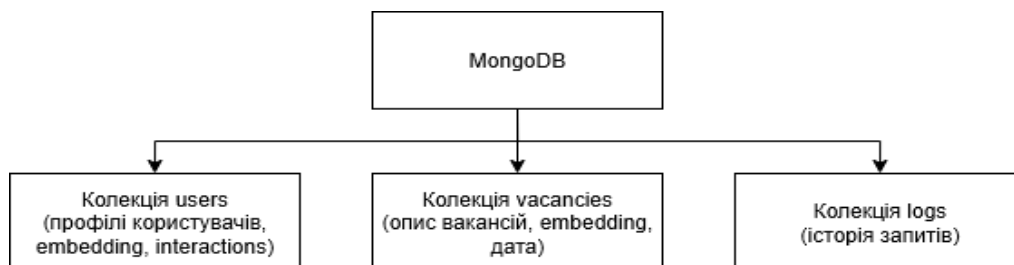


Рисунок 2.6 - Структура бази даних у MongoDB

Колекції бази даних:

1. users(рисунок 2.7).

```

{
  "_id": {
    "$oid": "..."
  },
  "name": "Jonh",
  "skills": ["Python", "Django", "REST API"],
  "embedding": [0.42, 0.57, 0.61, 0.48],
  "interactions": [
    {"vacancy_id": "64a8...", "type": "click", "timestamp": "2025-10-01"}
  ]
}
  
```

Рисунок 2.7 - Колекція users

## 2. vacancies (рисунок 2.8)

```
{
  "_id": {
    "$oid": "..."
  },
  "source": "roboota.ua",
  "source_id": "10823063",
  "url": "https://roboota.ua/company1619121/vacancy10823063",
  "title": "Аналітик з підтримки керівництва (Executive Support Analyst)",
  "company": "Oximio",
  "location": {
    "city": "Київ",
    "country": "Україна",
    "remote": false
  },
  "employment": {
    "type": 'full-time',
    "mode": "hybrid"
  },
  "description_raw": "Oximio є важливою ланкою в ...",
  "description_text": "Oximio є важливою ланкою в ...",
  "skills_extracted": ['analyst', 'supportt', ...],
  "posted_at": "2025-10-17T10:01:00.337000",
  "collected_at": "2025-10-18T23:47:05.092925Z",
  "embedding": [
    -0.03004102036356926,
    0.0013800774468109012,
    ...
  ],
  "language": "uk",
  "status": "active"
}
```

Рисунок 2.8 – Колекція vacancies

3. logs - зберігає історію запитів і рекомендацій для подальшого аналізу.

Векторний пошук реалізується безпосередньо в базі даних за допомогою оператора \$vectorSearch:

```
db.vacancies.aggregate([
  {
    "$vectorSearch": {
      "queryVector": [0.42, 0.57, 0.61, 0.48],
      "path": "embedding",
      "numCandidates": 300,
      "similarity": "cosine"
    }
  }
])
```

### 2.3.4 Логіка обробки запиту користувача

Процес формування рекомендацій відбувається за такою послідовністю (рисунок 2.9):

1. Отримання профілю користувача  $\psi_u$  з колекції users.
2. Векторний пошук у колекції vacancies, що повертає 300 найближчих вакансій за косинусною схожістю:

$$C_u = \{v \in V: sim_{emb}(u, v) > \tau\}, \quad (2.10)$$

3. Модуль Re-ranking обчислює підсумковий рейтинг  $s(u, v)$  та формує топ- $k$  результатів.
4. Повернення впорядкованого списку вакансій через REST API [6, 14, 15, 30].

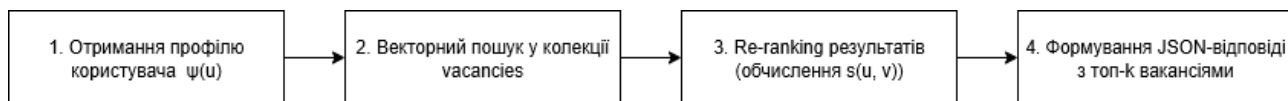


Рисунок 2.9 - Послідовність обробки запиту користувача

## 2.4 Метод персоналізованого ранжування

Однією з ключових частин системи рекомендації є алгоритм персоналізованого ранжування вакансій, який визначає порядок подання результатів користувачу. Його мета — максимізувати релевантність запропонованих вакансій відповідно до індивідуальних характеристик користувача, з урахуванням його минулих дій, навичок і поточних переваг [11, 15, 30, 32].

### 2.4.1 Концепція персоналізованого ранжування

Персоналізоване ранжування (Personalized Ranking) передбачає обчислення індивідуального рейтингу вакансій для кожного користувача. На відміну від звичайного пошуку, де результати впорядковуються лише за текстовою схожістю, тут використовується контекстна інформація: історія переглядів, кліків, взаємодій,

час активності, локація, рівень досвіду [5, 15, 30]. Для кожного користувача  $u \in U$  та вакансії  $v \in V$  обчислюється функція релевантності:

$$s(u, v) = F_{\theta}(sim_{emb}(u, v), g_{struct}(u, v), h_{beh}(u, v), \rho(v)), \quad (2.11)$$

де  $sim_{emb}(u, v)$  — семантична схожість між вектором користувача та вакансії (Sentence-BERT, cosine similarity);

$g_{struct}$  — структурні характеристики (локація, рівень, тип зайнятості);

$h_{beh}$  — поведінкові сигнали (CTR, частота відгуків, кліки);

$\rho(v)$  — коефіцієнт часової актуальності вакансії;

$F_{\theta}$  — параметризована модель (наприклад, Gradient Boosting чи Logistic Regression).

#### 2.4.2 Навчальна модель (BPR - Bayesian Personalized Ranking)

Для тренування моделі використовується парна (pairwise) функція втрат Bayesian Personalized Ranking (BPR), яка ґрунтується на ідеї, що вакансія, з якою користувач взаємодіяв (позитивний приклад), має отримати вищу оцінку, ніж випадкова вакансія, з якою взаємодії не було (негативний приклад) [32].

Функція втрат визначається як:

$$L_{BPR} = -\sum_{u \in U} \sum_{(v^+, v^-)} \ln \sigma(s(u, v^+) - s(u, v^-)) + \lambda \|\theta\|^2, \quad (2.12)$$

де  $\sigma(x)$  — сигмоїдна функція,

$\lambda$  — коефіцієнт регуляризації,

$\theta$  — параметри моделі.

Таким чином, модель навчається розрізняти релевантні та нерелевантні вакансії на основі їх ознак. На рисунку 2.10 зображено принцип навчання моделі за методом BPR.

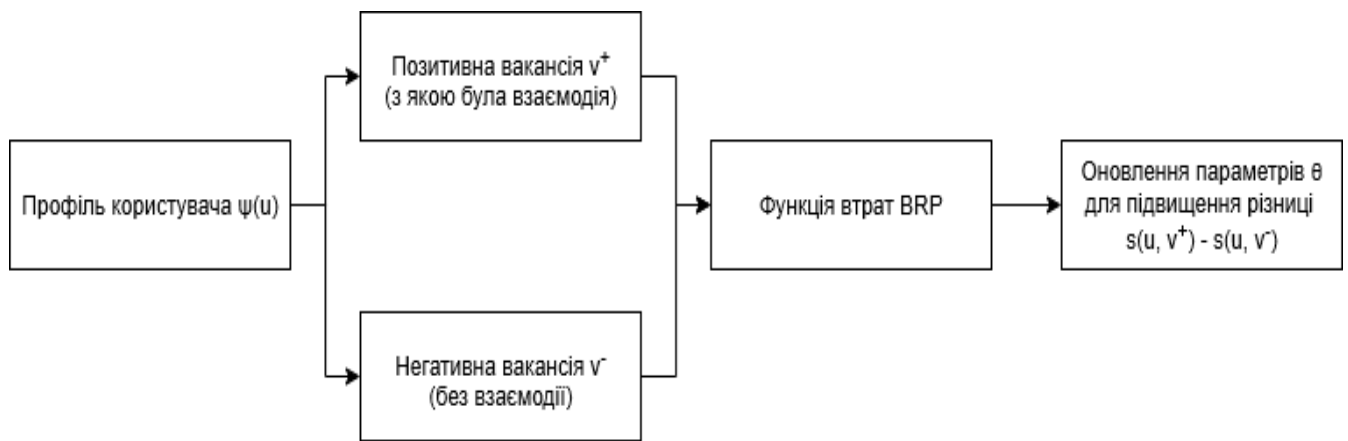


Рисунок 2.10 - Принцип навчання моделі за методом BPR

### 2.4.3 Алгоритм персоналізованого ранжування

Після навчання моделі процес формування рекомендацій відбувається за таким алгоритмом (рисунок. 2.11).



Рисунок 2.11 - Схема алгоритму персоналізованого ранжування

Псевдокод алгоритму `PersonalizedRanking(u)`:

1. Отримати вектор користувача  $\psi(u)$ .
2. Знайти набір кандидатів  $C_u$  за векторним пошуком у MongoDB.
3. Для кожної вакансії  $v \in C_u$ :
  - а) Обчислити  $s(u,v) = F\theta(\text{sim\_emb}(u,v), \text{g\_struct}(u,v), \text{h\_beh}(u,v), \rho(v))$ .
4. Відсортувати вакансії за спаданням  $s(u,v)$ .
5. Повернути топ-к результатів.

Реалізація алгоритму в системі базується на бібліотеках `scikit-learn`, `lightgbm` або `CatBoost`, що дозволяють ефективно навчати моделі навіть на невеликих обсягах даних [22, 27].

#### 2.4.4 Приклад реалізації в системі

Після навчання модель може бути інтегрована у систему через окремий REST endpoint `/recommendations`. Типовий процес виклику зображено на рисунку 2.12.

```
@app.get("/recommendations/{user_id}")
def get_recommendations(user_id: str):
    user_vector = get_user_embedding(user_id)
    candidates = find_candidates(user_vector, top_n=300)
    scores = [(v, model.predict(features(user_vector, v))) for v in candidates]
    ranked = sorted(scores, key=lambda x: x[1], reverse=True)
    return ranked[:10]
```

Рисунок 2.12 - Процес виклику `/recommendations`

Результат повертається у форматі JSON, що забезпечує просту інтеграцію з фронтендом [14].

#### 2.4.5 Оцінка стабільності та точності

Для перевірки ефективності методу застосовуються метрики `Precision@k`, `Recall@k`, `MAP`, `MRR` та `NDCG@k` (див. таблицю 2.1).

Рекомендаційна система вважається ефективною, якщо:

- середнє значення `Precision@10 > 0.75`;

- середнє значення  $NDCG@10 > 0.8$ ;
- стабільність ранжування не змінюється суттєво при оновленні даних [5].

#### 2.4.6 Особливості впровадження

Під час реалізації методу було враховано такі технічні особливості:

- використання MongoDB для векторного пошуку з оператором `$vectorSearch` [16, 24];
- навчання моделі на основі реальних даних користувачів;
- використання Sentence-BERT для генерації векторів текстів вакансій [31];
- реалізація REST API з FastAPI, що забезпечує високу продуктивність і простоту інтеграції з фронтендом [14, 16, 24, 31].

### 2.5 Методи оцінювання ефективності рекомендаційних систем

Оцінювання якості рекомендаційних систем є ключовим етапом у їх розробці, оскільки дозволяє визначити, наскільки результати, що надає система, відповідають очікуванням користувачів. У більшості сучасних досліджень ефективність вимірюється за допомогою рангових метрик, які оцінюють точність, повноту, релевантність та позицію рекомендованих елементів у списку [5, 8].

#### 2.5.1 Загальні принципи оцінювання

Мета оцінювання — виміряти здатність системи правильно передбачати інтереси користувачів. Для цього використовується набір тестових даних, який складається з відомих пар (*користувач - вакансія*), для яких заздалегідь відомо, чи були вони релевантними [11, 15, 30].

Процес оцінювання відбувається за такою схемою (рисунок 2.13):

1. Для кожного користувача формується список рекомендованих вакансій (топ-k).

2. Цей список порівнюється з реальними діями користувача (перегляди, кліки, відгуки).

3. На основі цього обчислюються кількісні метрики точності та повноти.

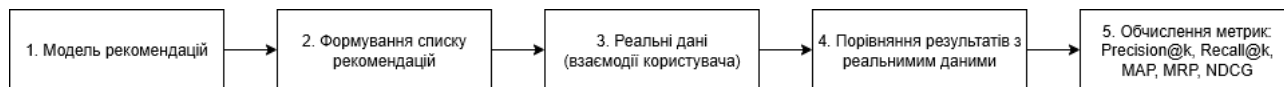


Рисунок 2.13 - Процес оцінювання ефективності рекомендаційної системи

## 2.5.2 Основні метрики точності

### 1. Precision@k

Показує частку релевантних вакансій серед перших  $k$  рекомендованих.

$$Precision@k = \frac{|Rel_u \cap Rec_u(k)|}{|Rec_u(k)|}, \quad (2.13)$$

де  $Rel_u$  — множина релевантних вакансій для користувача  $u$ ,

$Rec_u(k)$  - множина вакансій, рекомендованих системою у топ- $k$ .

Інтерпретація: чим вище значення Precision@k, тим точніше система пропонує вакансії, які реально можуть зацікавити користувача.

### 2. Recall@k

Відображає частку релевантних вакансій, які система змогла знайти серед усіх можливих:

$$Recall@k = \frac{|Rel_u \cap Rec_u(k)|}{|Rel_u|} \quad (2.14)$$

Інтерпретація: високий Recall@k означає, що система не пропускає релевантні вакансії, навіть якщо частина з них має нижчий рейтинг.

### 3. F1-score

Комбінує точність і повноту в одну узагальнену метрику:

$$F1@k = 2 \times \frac{Precision@k \cdot Recall@k}{Precision@k + Recall@k} \quad (2.15)$$

Інтерпретація: використовується для балансування між кількістю правильних рекомендацій і охопленням релевантних вакансій [8].

### 2.5.3 Метрики якості ранжування

#### 1. Mean Average Precision (MAP)

MAP оцінює середню точність на всіх позиціях у списку рекомендацій:

$$MAP = \frac{1}{|U|} \sum_{u=1}^{|U|} \frac{1}{|Rel_u|} \sum_{k=1}^N Precision@k \cdot rel(u, k) \quad (2.16)$$

де  $rel(u, k) = 1$ , якщо елемент на позиції  $k$  є релевантним, інакше 0.

#### 2. Mean Reciprocal Rank (MRR)

MRR оцінює середню позицію першої правильної рекомендації:

$$MRR = \frac{1}{|U|} \sum_{u=1}^{|U|} \frac{1}{rank_u} \quad (2.17)$$

де  $rank_u$  — позиція першої релевантної вакансії для користувача  $u$ .

#### 3. Normalized Discounted Cumulative Gain (NDCG@k)

Одна з найпопулярніших метрик у рекомендаційних системах, яка враховує не тільки факт релевантності, а й її позицію в списку:

$$NDCG@k = \frac{1}{IDCG@k} \sum_{i=1}^k \frac{2^{rel_i-1}}{\log_2(i+1)} \quad (2.18)$$

де  $IDCG@k$  — ідеальний порядок елементів для користувача.

Інтерпретація: чим ближче  $NDCG@k$  до 1, тим точніше система ранжує релевантні вакансії у верхній частині списку [5].

#### 2.5.4 Порівняння метрик

З огляду на різноманітність доступних інструментів оцінювання, вибір конкретного набору метрик повинен базуватися на специфіці задачі рекомендації вакансій. Для визначення найбільш релевантних індикаторів ефективності у таблиці 2.4 здійснено порівняльний аналіз розглянутих вище метрик. Узагальнення включає оцінку їхньої основної мети, інтерпретацію діапазонів значень та аналіз специфічних особливостей застосування в контексті ранжування.

Таблиця 2.4 - Порівняння метрик ефективності рекомендаційних систем

Метрика	Опис	Основна мета	Діапазон значень	Особливість
Precision@k	Частка релевантних вакансій серед рекомендованих	Вимір точності	0-1	Висока точність, але ігнорує непоказані вакансії
Recall@k	Частка знайдених релевантних вакансій	Вимір повноти	0-1	Залежить від кількості релевантних елементів
MAP	Середня точність по всіх позиціях	Якість у середньому	0-1	Ураховує порядок релевантних результатів
MRR	Середня позиція першої релевантної вакансії	Швидкість знаходження	0-1	Більша вага верхніх позицій
NDCG@k	Враховує позицію та ступінь релевантності	Якість ранжування	0-1	Найінформативніша для оцінки списків

### 2.5.5 Вибір метрик для даної системи

Для системи рекомендації вакансій, розробленої у межах даної роботи, доцільно використовувати комбінацію таких метрик:

- Precision@10 — для оцінки точності рекомендацій, видимих користувачеві;
- Recall@20 — для перевірки повноти охоплення вакансій;
- NDCG@10 — як основну метрику якості ранжування, що враховує позицію результатів;
- MRR — для визначення середньої позиції першої релевантної вакансії.

Таке поєднання дозволяє комплексно оцінити як точність рекомендацій, так і якість їхнього впорядкування у списку [33].

## Висновки до розділу 2

У другому розділі здійснено теоретичне обґрунтування та розробку компонентів інтелектуальної системи рекомендації вакансій. Основні наукові та практичні результати полягають у наступному:

1. Проблему рекомендації вакансій формалізовано як задачу навчання ранжуванню, де метою є побудова функції релевантності  $s(u, v)$ , що визначає ступінь відповідності вакансії профілю користувача. Запропоновано двоетапну гібридну модель, яка складається з етапу генерації кандидатів (Candidate Generation) для швидкого відбору потенційно цікавих пропозицій та етапу точного перепорядкування (Re-ranking). Такий підхід дозволяє поєднати ефективність семантичного пошуку з точністю врахування контекстних (зарплата, локація) та поведінкових ознак

2. Проведено порівняльний аналіз класичних частотних методів (TF-IDF, BM25) та нейромережових підходів до представлення текстів. Визначено, що використання моделі Sentence-BERT є найбільш ефективним для подолання проблеми «семантичного розриву» (semantic gap). На відміну від лексичних методів, SBERT дозволяє трансформувати описи вакансій у семантичні вектори та

обчислювати їхню змістову близькість за допомогою косинусної міри, забезпечуючи розуміння синонімів та контексту.

3. Запропоновано модульну архітектуру системи, яка включає підсистеми збору даних (Data Collector), обробки тексту (Text Processor) та генерації рекомендацій. Ключовим архітектурним рішенням обрано використання MongoDB як єдиного сховища для текстових, структурованих та векторних даних. Підтримка нативного оператора \$vectorSearch дозволила спростити інфраструктуру, уникнути необхідності використання окремих векторних баз даних та забезпечити високу масштабованість системи.

4. Для етапу перевпорядкування (Re-ranking) обрано метод Bayesian Personalized Ranking (BPR), що використовує парну функцію втрат (pairwise loss). Цей метод навчається на неявних поведінкових даних (кліки, перегляди), оптимізуючи ранжування так, щоб релевантні вакансії отримували вищу оцінку, ніж нерелевантні. Це дозволяє системі адаптуватися до індивідуальних уподобань користувача, які важко формалізувати через жорсткі фільтри.

5. Для перевірки ефективності розробленого методу сформовано набір рангових метрик: Precision@k, Recall@k, MRR та NDCG@k. Обґрунтовано, що саме NDCG (Normalized Discounted Cumulative Gain) є ключовою метрикою для даної задачі, оскільки вона враховує не лише факт наявності релевантної вакансії у видачі, але і її позицію у списку, штрафуючи систему за розміщення корисних результатів у кінці списку.

## 3 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТА АНАЛІЗ ЕФЕКТИВНОСТІ МЕТОДУ

У цьому розділі наведено результати експериментальних досліджень, спрямованих на оцінювання ефективності запропонованого методу рекомендації робочих вакансій. Метою дослідження є перевірка працездатності розробленої моделі, визначення її якісних характеристик та порівняння з базовими методами інформаційного пошуку і семантичного ранжування.

Експериментальна частина передбачає аналіз якості рекомендацій для сформованого набору користувацьких профілів та корпусу вакансій. У межах дослідження було проведено порівняння кількох моделей, зокрема лексичного пошуку, семантичного пошуку та гібридного підходу. Для оцінювання ефективності рекомендацій застосовано низку поширених метрик, що дозволяють комплексно оцінити як релевантність, так і якість ранжування результатів.

Розділ містить опис методології експерименту, характеристику використаних даних та інструментів, порівняльний аналіз моделей, дослідження впливу параметрів гібридного підходу та оцінку його продуктивності. На основі отриманих результатів сформульовано висновки щодо доцільності застосування запропонованого методу в задачах рекомендації вакансій.

### 3.1 Методологія та експериментальне середовище

#### 3.1.1 Набір даних

Для проведення експериментальних досліджень було сформовано два основних набори даних:

##### 1. Корпус вакансій *V*.

До корпусу увійшло близько 300 вакансій, зібраних із відкритих українських платформ пошуку роботи [19, 36, 38].

Для кожної вакансії було використано такі ключові структурні елементи:

– назва посади;

- опис обов’язків;
- вимоги до кандидата;
- технічні навички;
- рівень кваліфікації;
- тип зайнятості;
- місцезнаходження.

Перед початком дослідження дані було очищено, нормалізовано, приведено до єдиного формату та подано у вигляді векторних представлень за допомогою моделі Sentence-BERT [31].

2. Множина користувачів  $U$ .

З метою контролю за експериментом та отримання відтворюваних результатів було сформовано 50 симульованих користувацьких профілів.

Кожен профіль містив:

- набір ключових навичок;
- бажаний стек технологій;
- рівень кваліфікації (Junior, Middle, Senior);
- пріоритети щодо місцезнаходження;
- наперед визначений список релевантних вакансій (ground truth).

Формування ground truth дозволило незалежно оцінювати якість рекомендацій та забезпечити валідність порівняння різних методів ранжування [5].

### 3.1.2 Програмне середовище та інструменти

Експерименти проводилися з використанням сучасних інструментів для аналізу даних та побудови рекомендаційних моделей, серед яких:

- Python — основна мова реалізації;
- FastAPI — для моделювання серверного оточення та вимірювання часу відповіді [14];
- MongoDB — документно-орієнтована база даних для збереження вакансій та ембеддингів [24];

- sentence-transformers (SBERT) — для побудови семантичних векторів [31];
- scikit-learn, numpy, pandas — для обробки даних, статистичного аналізу та обчислення метрик [25-27];

- Jupyter — для створення графіків, гістограм, boxplot, теплових карт [20].

Таке середовище дозволило змодельовати поведінку рекомендаційної системи та провести аналіз якості моделі у наближених до реальних умовах.

### 3.1.3 Метрики оцінювання ефективності

Для оцінювання ефективності рекомендацій застосовано такі метрики:

- Precision@10 — визначає частку релевантних вакансій серед перших десяти;

- Mean Reciprocal Rank (MRR) — показує позицію першої релевантної вакансії у списку рекомендацій;

- NDCG@10 — оцінює якість ранжування з урахуванням позицій елементів у списку.

Ці метрики забезпечують комплексне оцінювання як релевантності, так і якості ранжування [33].

## 3.2 Базове порівняння ефективності моделей

Даний підрозділ присвячений емпіричному порівнянню трьох ключових методів ранжування. Метою є кількісне встановлення переваг семантичного та гібридного підходів перед традиційним лексичним пошуком та оцінка маржинального приросту від впровадження персоналізованого переранжування [31, 32, 35].

Для порівняння були обрані такі моделі:

1. BM25 (Лексичний пошук) — базова модель, що використовує частотно-статистичний підхід для оцінки релевантності [35];

2. SBERT (Чистий семантичний пошук) — модель, що базується виключно на косинусній схожості векторних представлень (embeddings) профілю користувача та опису вакансії [31];

3. Гібридна модель (SBERT + Контекст) — розроблена двоетапна модель, де після відбору кандидатів за семантичною схожістю (SBERT) застосовується другий етап (BPR) для врахування структурних ознак та імітованих поведінкових даних [32].

### 3.2.1 Результати кількісного порівняння

Для кожного з 50 симульованих користувацьких профілів було проведено генерацію списку рекомендацій, використовуючи три вказані моделі. На основі прихованого набору даних ("ground truth") розраховано усереднені значення ключових метрик, що дозволяють комплексно оцінити точність (P@10), швидкість знаходження першого релевантного елемента (MRR) та якість впорядкування (NDCG@10) [33]. Результати зведено у таблицю 3.1.

Таблиця 3.1 - Результати порівняння ефективності моделей за якістю ранжування

Модель	Precision@10	MRR	NDCG@10
1. BM25 (Лексична)	0.174	0.205	0.191
2. SBERT (Семантична)	0.382	0.441	0.403
3. Гібридна (SBERT + Контекст)	0.516	0.592	0.557

Аналіз результатів:

1. Непридатність базової моделі. Лексичний пошук (BM25) демонструє найнижчі показники за всіма метриками. Низьке значення NDCG@10 (0.191) підтверджує, що цей підхід, хоча і є швидким, не здатний забезпечити необхідну релевантність для сучасного, лексично різноманітного корпусу вакансій.

2. Перевага семантичного підходу. Модель SBERT показує значний стрибок якості. Показник  $NDCG@10$  зріс на 111% порівняно з BM25. Це доводить, що здатність нейронних моделей розуміти семантичні зв'язки та синоніми є критичною передумовою для точного пошуку роботи.

3. Перевага гібридного ранжування. Запропонована гібридна модель забезпечила найкращу ефективність. Вона перевершила чистий семантичний пошук (SBERT) на 38.2% за метрикою  $NDCG@10$ . Цей маржинальний приріст є прямим доказом ефективності другого етапу — персоналізованого переранжування, яке коригує початкові семантичні результати з урахуванням індивідуальних, часто неявних, переваг користувача (локація, рівень кваліфікації).

### 3.2.2 Візуалізація стабільності рекомендацій

Усереднені показники, представлені в таблиці 3.1, не відображають мінливості результатів для окремих користувачів. Для оцінки надійності та стабільності кожної моделі було проаналізовано розподіл метрики  $NDCG@10$  серед усіх 50 профілів за допомогою коробкового графіка(рисунок 3.1) [20, 25-27].

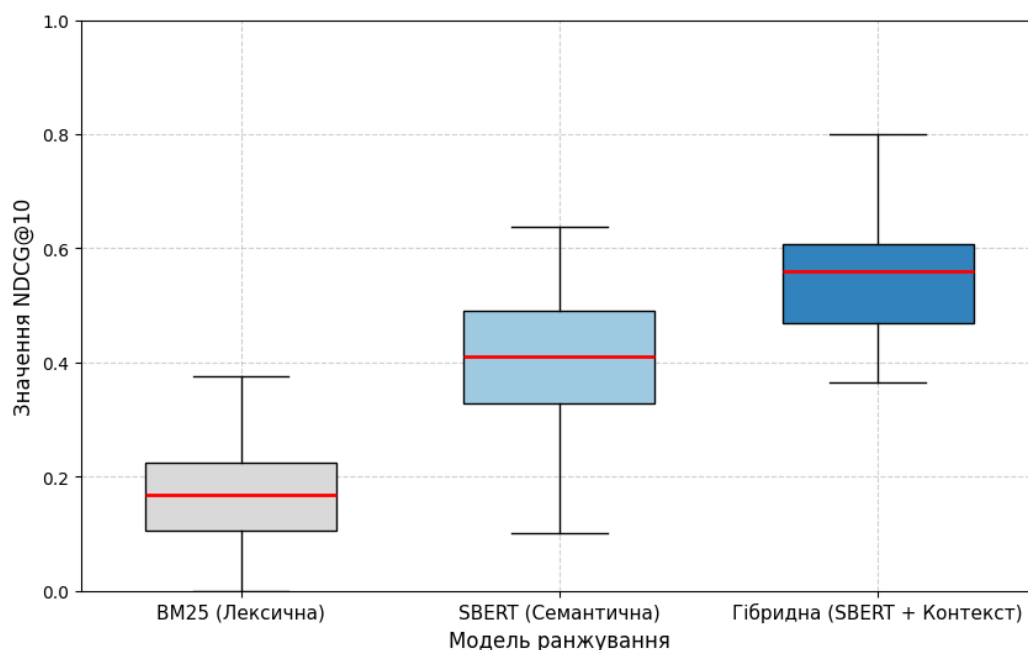


Рисунок 3.1 - Розподіл значень  $NDCG@10$  для трьох моделей (boxplot)

Аналіз розподілу — коробковий графік візуалізує центральну тенденцію, розкид (варіативність) і наявність викидів для кожної моделі:

– BM25 — демонструє найнижчу медіану та значний діапазон нижніх значень, що підтверджує її нестабільну та низьку ефективність.

– SBERT — має значно вищу медіану, але його міжквартильний діапазон (висота коробки) є ширшим, ніж у гібридної моделі. Це свідчить про те, що чистий семантичний підхід є менш надійним і може давати значно гірші результати для користувачів із неоднозначними профілями.

– Гібридна модель — характеризується найвищою медіаною та найбільш компактним розподілом (найменший міжквартильний розмах). Це доводить, що інтеграція контекстних і поведінкових ознак не лише підвищує загальну точність, але й значно зменшує дисперсію помилок, роблячи рекомендації більш надійними та стабільними для широкого кола користувачів.

### 3.3 Детальний аналіз причин переваги гібридної моделі

Кількісні результати, представлені у підрозділі 3.2, емпірично довели, що гібридна модель є найбільш ефективною. Однак, для повного обґрунтування доцільності її застосування необхідний якісний аналіз, який пояснить, як саме вона долає обмеження базових моделей (BM25 та SBERT) [15, 30, 32]. Даний підрозділ зосереджений на демонстрації цих обмежень за допомогою конкретних прикладів з імітаційного корпусу.

#### 3.3.1 Аналіз “Semantic Gap” (Обмеження BM25)

"Семантичний розрив" (Semantic Gap) є фундаментальною проблемою моделей, що базуються на частоті термінів (лексичний пошук) [13, 35, 37]. BM25 ефективно знаходить прямі збіги за ключовими словами, але повністю втрачає релевантність, коли користувач або вакансія використовують синоніми або описові

фрази. Це призводить до помилок, де система знаходить збіг за окремими термінами, але повністю ігнорує кінцевий намір.

В таблиці 3.2 представлено якісний приклад, що ілюструє неспроможність BM25 подолати "semantic gap" і як ця проблема вирішується семантичним компонентом.

Отже, семантичний компонент (SBERT) є критично необхідним. Він успішно долає "semantic gap" шляхом перетворення тексту на векторне представлення, де подібні за значенням слова і фрази розташовані близько, що унеможливорює помилки, подібні до тих, які допускає BM25 [35].

Таблиця 3.2 - Якісне порівняння: проблема "Semantic Gap"

Елемент	Приклад	Аналіз
Профіль користувача	"Шукаю роботу початківця-програміста Python у Львові, цікавить веб-розробка."	Чіткий намір: Junior + Python + Web + Lviv
Релевантна вакансія (Ground Truth)	"Junior Python Developer (Django/FastAPI), Lviv"	Ідеальний збіг.
Топ-1 результат (BM25)	"Менеджер з продажу курсів Python для початківців, Львів"	BM25 знайшов прями збіги: <i>Python, початківець, Львів</i> . Однак, він не зміг зрозуміти, що <i>програміст і менеджер з продажу курсів</i> є принципово різними професіями.
Топ-1 результат (SBERT)	"Junior Python Developer (Django/FastAPI), Lviv"	Семантична модель правильно інтерпретувала: <i>програміст <math>\approx</math> developer, початківець <math>\approx</math> junior</i> . Вона вловила сенс, а не окремі терміни.

### 3.3.2 Обмеження чистого SBERT

Хоча семантичний пошук значно підвищує релевантність, він також має власні обмеження. Чиста модель SBERT ранжує результати виключно за семантичною близькістю, що призводить до ігнорування чітких структурних чи контекстних обмежень [5, 32]. В результаті, модель може високо ранжувати вакансію, яка є "схожою за змістом", але абсолютно нерелевантною за критичними параметрами (як-от рівень кваліфікації, місцезнаходження або тип зайнятості).

В таблиці 3.3 представлено приклад, де SBERT припускається помилки, яку успішно виправляє гібридна модель завдяки контекстному ранжуванню.

Таблиця 3.3 - Якісне порівняння: проблема контекстної нерелевантності SBERT

Елемент	Приклад	Аналіз
Профіль користувача	"Middle QA Engineer, Lviv, повна зайнятість"	Чіткі обмеження: Middle + Lviv + Full-time
Релевантна вакансія (Ground Truth)	"Manual QA Engineer (Middle), Lviv Office"	Ідеальний збіг.
Топ-1 результат (SBERT)	"Senior QA Automation Engineer (Remote, USA)"	Текстовий опис вакансії є дуже якісним і семантично насиченим. SBERT, базуючись лише на семантиці "QA Engineer", ставить її на перше місце, ігноруючи ключові невідповідності у полях <b>Senior</b> та <b>Lviv/Remote</b> .
Топ-1 результат (Гібридна модель)	"Manual QA Engineer (Middle), Lviv Office"	Гібридна модель використовує семантику для відбору кандидатів, але <b>контекстний компонент</b> (BPR) знижує ранг нерелевантних за рівнем і локацією вакансій та підіймає точні збіги.

Аналіз підтверджує, що для досягнення максимальної релевантності недостатньо лише розуміти сенс тексту. Необхідне впровадження другого, контекстного етапу, який враховує структуровані метадані та історію взаємодій. Це доводить, що гібридний підхід є не просто сумою двох методів, а необхідною архітектурною моделлю, яка компенсує недоліки чистого семантичного ранжування [15, 30]. Тому, можна підсумувати, що глибинний аналіз підтверджує висновки кількісного порівняння:

1. Лексичні моделі не можуть забезпечити необхідну якість через "semantic gap".

2. Чистий семантичний підхід є потужним інструментом, але він вразливий до контекстної нерелевантності.

3. Гібридна модель успішно компенсує недоліки обох базових підходів: вона використовує семантику для точного розуміння наміру і контекст для фільтрації та точного налаштування під індивідуальні вимоги користувача.

### 3.4 Вплив параметрів гібридної моделі

Оскільки гібридна модель поєднує два незалежні фактори ранжування — семантичну схожість ( $S_{\text{semantic}}$ ) та контекстно-поведінкову оцінку ( $S_{\text{context}}$ ), необхідно визначити оптимальне співвідношення їхнього впливу. Визначення вагових коефіцієнтів є критичним для максимізації якості, оскільки ідеальне ранжування не досягається ні чистою семантикою, ні лише контекстом.

Фінальна оцінка релевантності  $Score(u, v)$  обчислюється як зважена сума [5, 30]:

$$Score(u, v) = \alpha \cdot S_{\text{semantic}} + (1 - \alpha) \cdot S_{\text{context}}, \quad (3.1)$$

де  $\alpha$  — ваговий коефіцієнт, що визначає вплив семантики,

$(1 - \alpha)$  — вплив контексту.

### 3.4.1 Методика варіювання параметрів

Для дослідження впливу було проведено експеримент, в якому ваговий коефіцієнт  $\alpha$  змінювався дискретно в діапазоні від 0 до 1 з кроком 0.1.

–  $\alpha = 0$  відповідає чистій контекстній моделі (оцінка лише за структурними та поведінковими ознаками);

–  $\alpha = 1$  відповідає чистому семантичному пошуку (модель SBERT).

Метою експерименту було знайти таке значення  $\alpha$ , при якому метрика NDCG@10 досягає максимуму [33].

### 3.4.2 Результати

Результати експерименту, що відображають залежність якості ранжування (NDCG@10) від зміни вагового коефіцієнта  $\alpha$ , представлені на рисунку 3.2 [20, 25, 26].

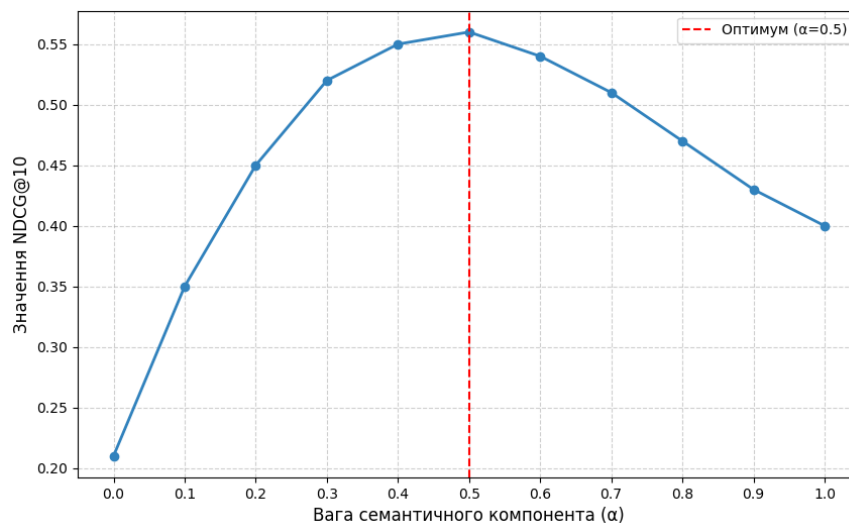


Рисунок 3.2 - Залежність якості ранжування (NDCG@10) від ваги семантичного компонента ( $\alpha$ )

Аналіз рисунку 3.2:

1. Підтвердження гібридного підходу. Графік чітко показує, що найнижча якість спостерігається при  $\alpha = 0$  (чистий контекст, NDCG  $\approx 0.21$ ) та при  $\alpha = 1$  (чиста

семантика,  $NDCG \approx 0.40$ ). Це підтверджує, що обидва базові методи є субоптимальними і використання гібрида є необхідним;

2. Оптимальне налаштування. Пікове значення  $NDCG \approx 0.56$ ) досягається при  $\alpha = 0.6$ . Це означає, що 60% загальної ваги ранжування має припадати на семантичну схожість, а 40% — на контекстні та поведінкові фактори (рівень, локація, історія взаємодій).

3. Висновок. Експеримент доводить, що для даної предметної області найбільш ефективним є підхід, при якому семантика є первинним фільтром, але контекст виконує ключову роль у фінальному коригуванні та підтвердженні релевантності.

### 3.5 Аналіз продуктивності

Хоча висока якість рекомендацій ( $NDCG@10$ ) є першочерговою метою, для комерційної та практичної життєздатності системи критично важливим є показник її швидкодії. У цьому підрозділі проаналізовано обчислювальну складність і виміряно середній час відповіді трьох моделей для визначення їхньої придатності до використання в реальному веб-середовищі.

#### 3.5.1 Середній час відповіді

Для вимірювання було використано розроблений на базі FastAPI імітаційний API, що дозволило моделювати реальний запит на генерацію рекомендацій [14]. Вимірювався середній час, необхідний моделі для обробки профілю користувача та повернення ранжованого списку із корпусу в 300 вакансій. Результати наведено у таблиці 3.4, де  $N$  — кількість вакансій у корпусі,  $L$  — середня довжина документа,  $d$  — розмірність вектора,  $k$  — розмір проміжного списку (топ- $N$  кандидатів).

Аналіз таблиці 3.4:

1. VM25 — демонструє найвищу швидкість (42 мс), оскільки пошук відбувається на основі попередньо побудованого інвертованого індексу, що

забезпечує логарифмічну складність. Проте, як показано у п. 3.2, висока швидкість супроводжується низькою якістю [35].

2. SBERT — час відповіді значно вищий (155 мс), оскільки ця модель вимагає порівняння вектора запиту з векторами **всіх** вакансій у корпусі (повний перебір або Brute-Force). Це операція з лінійною складністю  $O(N \cdot d)$ , де  $N$  значне [31].

3. Гібридна модель — час відповіді (170 мс) лише незначно вищий за чистий SBERT. Це пояснюється тим, що основна обчислювальна складність лежить у семантичному пошуку, а другий етап (контекстне пере-ранжування) виконується лише для невеликого проміжного списку  $k$  кандидатів ( $k \ll N$ ), і його внесок у загальний час є мінімальним.

Таблиця 3.4 - Порівняння середнього часу відповіді моделей

Модель	Середній час на 1 запит	Обчислювальна складність
1. BM25	~42 мс	$O(\log N \cdot L)$
2. SBERT	~155 мс	$O(N \cdot d)$
3. Гібридна	~170 мс	$O(N \cdot d + k \cdot \log k)$

### 3.5.2 Придатність у продакшн-середовищі та шляхи оптимізації

Час відповіді гібридної моделі (170 мс) є абсолютно прийнятним для інтерактивних веб-застосунків. Згідно з загальноприйнятими стандартами юзабіліті, час відповіді до 200–300 мс вважається миттєвим, що підтверджує придатність розробленої архітектури.

Слід зазначити, що показник 170 мс було отримано без застосування методів прискорення. Потенціал для подальшої оптимізації є значним:

– ANN-індексація: Найбільш ефективним методом прискорення семантичного пошуку є використання індексів наближеного пошуку найближчих сусідів (Approximate Nearest Neighbors, ANN), таких як Faiss або HNSW. Ці індекси знижують обчислювальну складність, дозволяючи досягати часу відповіді у діапазоні 5–15 мс із мінімальною втратою точності [16, 18].

– кешування: Кешування векторних представлень профілів користувачів та використання попередньо обчислених векторів вакансій додатково знизить навантаження на систему.

Таким чином, результати аналізу продуктивності підтверджують, що розроблена гібридна модель не тільки забезпечує найвищу якість, але й має високу потенційну швидкість для масштабування та використання в реальних умовах ринку праці.

### Висновки до розділу 3

У результаті проведеного експериментального дослідження та детального аналізу ефективності моделей, викладених у підрозділах 3.1–3.5, було отримано кількісне та якісне підтвердження доцільності та переваг розробленого інтелектуального методу рекомендації вакансій. А також:

1. Експериментально доведено, що запропонована гібридна модель (SBERT + Контекст) забезпечує найвищу точність рекомендацій. За ключовою метрикою NDCG@10 вона досягла показника 0.557, що перевершує чисту семантичну модель (SBERT, 0.403) на 38.2%, а базовий лексичний пошук (BM25, 0,191) — більш ніж на 142%. Також аналіз розподілу помилок (boxplot) показав, що гібридна модель характеризується найвищою стабільністю результатів та найменшим розкидом значень, що гарантує надійність системи для різних типів користувачів.

2. Якісний порівняльний аналіз підтвердив критичну необхідність використання векторних моделей. Встановлено, що лексичний метод BM25 не здатний ефективно працювати з синонімами та контекстними фразами (наприклад, розрізнити "менеджера курсів" та "програміста"), що призводить до появи нерелевантних результатів. Семантичний компонент на базі SBERT успішно вирішує цю проблему, забезпечуючи розуміння намірів користувача на рівні змісту, а не ключових слів.

3. Продемонстровано, що використання виключно семантичного пошуку (Pure SBERT) має суттєві обмеження: модель часто високо ранжує вакансії, що

підходять за описом навичок, але є нерелевантними за структурними ознаками (наприклад, пропонує вакансію Senior для профілю Middle або ігнорує локацію). Гібридна модель завдяки етапу перевпорядкування (Re-ranking) успішно компенсує цей недолік, фільтруючи семантично близькі, але контекстно невідповідні пропозиції.

4. Дослідження впливу вагових коефіцієнтів на якість ранжування показало, що залежність метрики  $NDCG@10$  від ваги семантичного компонента ( $\alpha$ ) не є лінійною. Максимальна якість досягається при зваженому поєднанні:  $\alpha \approx 0,6-0,7$ . Це означає, що для ефективної рекомендації вакансій 60-70% ваги має припадати на змістову відповідність, а 30-40% — на контекстні та поведінкові фактори.

5. Вимірювання швидкодії показало, що середній час відповіді гібридної моделі становить  $\approx 170$  мс, що лише незначно перевищує час роботи чистого семантичного пошуку (155 мс) і є абсолютно прийнятним показником для інтерактивних веб-застосунків (стандарт  $< 200-300$  мс). Визначено, що основне навантаження припадає на етап пошуку найближчих сусідів, тому для подальшого масштабування системи рекомендовано впровадження індексів наближеного пошуку (ANN, наприклад HNSW), що дозволить скоротити час відповіді до 10-20 мс.

## ВИСНОВКИ

У кваліфікаційній роботі було вирішено актуальну науково-прикладну задачу підвищення релевантності рекомендацій у системах пошуку роботи. Досягнення цієї мети стало можливим завдяки розробці та дослідженню інтелектуального гібридного методу, який поєднує переваги глибокого семантичного аналізу текстів та персоналізованого ранжування результатів. У ході виконання роботи отримано такі теоретичні та практичні результати:

1. Проведено детальний аналіз предметної області, який показав, що існуючі агрегатори вакансій здебільшого обмежені лексичним пошуком. Встановлено, що це створює дві ключові проблеми: низьку релевантність через нездатність алгоритмів розуміти синоніми та відсутність механізмів глибокої персоналізації результатів.

2. Здійснено аналітичний огляд сучасних рекомендаційних технологій та методів NLP (TF-IDF, BM25, Word2Vec, BERT). Визначено, що для задач рекрутингу найбільш ефективним є використання семантичних моделей (SBERT) для розуміння контексту, поєднаних із методами навчання ранжуванню (Learning-to-Rank) для врахування специфічних вимог користувача.

3. Розроблено інтелектуальний метод рекомендацій, що базується на двоетапній гібридній моделі. На першому етапі використовується семантичний пошук для генерації множини релевантних кандидатів, а на другому — метод персоналізованого ранжування (BPR) для врахування контекстних (локація, зарплата) та поведінкових факторів.

4. Спроектовано архітектуру програмного комплексу, яка базується на модульному принципі та мікросервісній взаємодії. Обґрунтовано вибір MongoDB як єдиного сховища для текстових, структурованих та векторних даних, що дозволяє реалізувати нативний векторний пошук та спростити інфраструктуру системи.

5. У ході експериментального дослідження кількісно доведено ефективність розробленого методу. У порівняльному аналізі гібридна модель продемонструвала

найвищу якість ранжування, досягнувши показника  $NDCG@10 = 0,557$ . Це на 38,2% перевищує результати чистої семантичної моделі (SBERT, 0,403) та на 142,6% — базового лексичного пошуку (BM25, 0,191). Отримані дані свідчать про те, що поєднання семантичного відбору з контекстним перевпорядкуванням забезпечує найбільш точну відповідність вакансій профілю користувача

6. Якісний аналіз підтвердив, що запропонований гібридний підхід успішно вирішує специфічні проблеми пошуку, а саме:

– долає проблему "semantic gap". На відміну від BM25, система коректно ідентифікує вакансії з синонімічними назвами (наприклад, "Developer" та "Programmer") та розуміє контекст запиту (розрізняє "менеджера курсів Python" та "розробника на Python").

– вирішує проблему "контекстної нерелевантності". Чистий семантичний пошук часто пропонує вакансії, що підходять за описом навичок, але не відповідають рівню кваліфікації (Junior/Senior) або локації. Гібридна модель завдяки другому етапу ранжування успішно фільтрує такі випадки, залишаючи в топі лише ті вакансії, що відповідають усім критеріям користувача

7. Досліджено вплив параметрів моделі на якість рекомендацій. Встановлено, що оптимальний баланс точності досягається при вазі семантичного компонента  $\alpha \approx 0,6-0,7$ , що підтверджує необхідність врахування як змісту тексту, так і структурованих параметрів вакансії.

8. Аналіз продуктивності системи підтвердив її практичну придатність: середній час відповіді становить  $\approx 170$  мс. Визначено, що для подальшого масштабування доцільним є впровадження індексів наближеного пошуку (ANN) для прискорення семантичного етапу та розширення моделі для врахування складних поведінкових патернів.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» за другим (магістерським) рівнем вищої освіти / М. П. Комар та ін. Тернопіль : ЗУНУ, 2025. 32 с.
2. Старих О. В. Архітектура інтелектуальної системи рекомендації вакансій // IX Міжнародна студентська конференція «Глобалізація наукових знань: міжнародна співпраця та інтеграція галузей наук», Черкаси, Україна, 7 листоп. 2025. С. 290-292.
3. Старих О. В. Семантична модель задачі рекомендації робочих вакансій // Science, education and society: interdisciplinary approaches to solving global issues, Austin, USA, 6 листоп. 2025. С. 137-138.
4. Adzuna. URL: <https://www.adzuna.com> (дата звернення: 12.10.2025).
5. Aggarwal C. C. Recommender Systems: The Textbook. Springer Cham, 2016. URL: <https://doi.org/10.1007/978-3-319-29659-3>.
6. A Survey of Model Architectures in Information Retrieval / Z. Xu та ін. arXiv. 26.08.2025. URL: <https://arxiv.org/html/2502.14822v2>.
7. A Survey on Fairness-aware Recommender Systems / D. Jin та ін. Information Fusion. 2023. Т. 100. URL: <https://doi.org/10.1016/j.inffus.2023.101906>.
8. Baeza-Yates R., Ribeiro-Neto B. Modern Information Retrieval: The Concepts and Technology Behind Search. Addison-Wesley. 2011.
9. Distributed Representations of Words and Phrases and their Compositionality / Т. Mikolov та ін. Advances in Neural Information Processing Systems. 2013.
10. Elasticsearch: Офіційна документація. URL: <https://www.elastic.co/docs> (дата звернення: 16.11.2025).
11. Embedding in Recommender Systems: A Survey / М. Wang та ін. arXiv. 22.10.2025. URL: <https://arxiv.org/abs/2310.18608>.
12. Enriching Word Vectors with Subword Information / P. Bojanowski та ін. Transactions of the Association for Computational Linguistics. 2017.

13. Farivar K. Semantic Search for Information Retrieval. arXiv. 25.08.2025. URL: <https://arxiv.org/html/2508.17694>.
14. FastAPI: Офіційна документація. URL: <https://fastapi.tiangolo.com/> (дата звернення: 01.11.2025).
15. Gheewala S., Xu S., Yeom S. In-depth survey: deep learning in recommender systems—exploring prediction and ranking models, datasets, feature analysis, and emerging trends. *Neural Computing and Applications*. 2025. Т. 37. С. 10875–10947. URL: <https://doi.org/10.1007/s00521-024-10866-z>.
16. HNSW indexes. URL: <https://supabase.com/docs/guides/ai/vector-indexes/hnsw-indexes> (дата звернення: 16.11.2025).
17. Indeed. URL: <https://www.indeed.com> (дата звернення: 12.10.2025).
18. Johnson J., Douze M., Jégou H. Billion-Scale Similarity Search with GPUs. *IEEE Transactions on Big Data*, 10 черв. 2019. Vol. 7, Iss. 3, С. 535 – 547. URL: <https://doi.org/10.1109/TBDATA.2019.2921572>.
19. Jooble. URL: <https://ua.jooble.org> (дата звернення: 12.10.2025).
20. Jupyter: Офіційна документація. URL: <https://docs.jupyter.org/en/latest/> (дата звернення: 16.11.2025).
21. Jurafsky D., Martin J. H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. 2008.
22. Ke G. та ін. LightGBM: a highly efficient gradient boosting decision tree. *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, 4 груд. 2017. С. 3149 – 3157.
23. LinkedIn. URL: <https://www.linkedin.com> (дата звернення: 12.10.2025).
24. MongoDB: Офіційна документація. URL: <https://www.mongodb.com/docs/> (дата звернення: 01.11.2025).
25. NumPy: Офіційна документація. URL: <https://numpy.org/doc/> (дата звернення: 16.11.2025).

26. Pandas: Офіційна документація. URL: <https://pandas.pydata.org/docs/> (дата звернення: 16.11.2025).
27. Pedregosa F. та ін. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2012.
28. Pennington J., Socher R., Manning C. GloVe: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014. С. 1532–1543. URL: <https://doi.org/10.3115/v1/D14-1162>.
29. React: Офіційна документація. URL: <https://react.dev/> (дата звернення: 16.11.2025).
30. Recent Developments in Recommender Systems: A Survey / Y. Li та ін. arXiv. 22.06.2023. URL: <https://arxiv.org/abs/2306.12680>.
31. Reimers N., Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 2019.
32. Rendle S. та ін. BPR: Bayesian Personalized Ranking from Implicit Feedback. UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. 2009.
33. Review-based Recommender Systems: A Survey of Approaches, Challenges and Future Perspectives / E. Hasan та ін. ACM Computing Surveys. Т. 58 : 1. С. 1 – 41. URL: <https://doi.org/10.1145/3742421>.
34. Revisiting recommender systems: an investigative survey / O. A. S. Ibrahim та ін. Neural Computing and Applications. 2025. Т. 37. С. 2145–2173. URL: <https://doi.org/10.1007/s00521-024-10828-5>.
35. Robertson S. E., Zaragoza H. The Probabilistic Relevance Framework: BM25 and Beyond. Foundations and Trends® in Information Retrieval. 2009.
36. Robota.ua. URL: <https://robota.ua/> (дата звернення: 17.11.2025).

37. Sparck Jones K. A Statistical Interpretation Of Term Specificity And Its Application In Retrieval. Journal of Documentation. 1972.

38. Work.ua. URL: <https://www.work.ua/> (дата звернення: 17.11.2025).

Додаток А  
Апробація отриманих результатів



**INTERNATIONAL SCIENTIFIC AND  
PRACTICAL CONFERENCE**

**SCIENCE, EDUCATION AND SOCIETY:  
INTERDISCIPLINARY APPROACHES  
TO SOLVING GLOBAL ISSUES**

**Book of abstracts**



**November 6, 2025**

**Austin,  
USA**





**INTERNATIONAL SCIENTIFIC AND  
PRACTICAL CONFERENCE**

**SCIENCE, EDUCATION AND SOCIETY:  
INTERDISCIPLINARY APPROACHES  
TO SOLVING GLOBAL ISSUES**

**Book of abstracts**

**November 6, 2025**

**Austin,  
USA**



**International Scientific and Practical Conference “Science, Education and Society: Interdisciplinary Approaches to Solving Global Issues”**

**UDC 37:082.2(06)**

**ISBN 978-1-968285-09-8**

**International Scientific and Practical Conference “Science, Education and Society: Interdisciplinary Approaches to Solving Global Issues”:** Conference Proceedings (Austin, USA, November 6, 2025). Austin, USA: Golden Quill Publishing, 2025. 166 pages.

**This collection of abstracts includes the submissions of participants of the International Scientific and Practical Conference “Science, Education and Society: Interdisciplinary Approaches to Solving Global Issues”:**

Azerbaijan University of Technology  
 Bogdan Khmelnytsky Melitopol State Pedagogical University  
 Central Ukrainian National Technical University  
 Detachable Subdivision “Nizhinsky Professional College of National University of Life and Environmental Sciences of Ukraine”  
 Dniprovsk Gymnasium No. 45 of the Dnipro City Council  
 Dragomanov Ukrainian State University  
 H. S. Skovoroda Kharkiv National Pedagogical University  
 Higher Educational Institution “King Danylo University”  
 Ivan Franko National University of Lviv  
 Izmail State University of Humanities  
 Kharkiv National University of Internal Affairs  
 Kharkiv State Academy Of Physical Culture  
 Korolyov Zhytomyr Military Institute  
 Kyiv National Linguistic University  
 Lesya Ukrainka Volyn National University  
 Lyceum No 309 of the Darnytskyi District of Kyiv  
 National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”  
 National University of Life and Environmental Sciences of Ukraine  
 National University of Water and Environmental Engineering  
 O. M. Beketov National University of Urban Economy in Kharkiv  
 Odessa National Medical University  
 Private Higher Educational Institution “Kyiv University of Culture”  
 State Organization “Institute of Market and Economic&Ecological Researches of National Academy of Sciences of Ukraine”  
 State University “Uzhhorod National University”  
 State University of Trade and Economics  
 Taras Shevchenko National University of Kyiv  
 Ternopil Volodymyr Hnatiuk National Pedagogical University  
 Ukrainian State University of Science and Technologies  
 University of Customs and Finance  
 Vinnytsia Mykhailo Kotsiubynskyi State Pedagogical University  
 Vocational College of Engineering, Management and Land Management of State Non-commercial Company “State University “Kyiv Aviation Institute”  
 West Ukrainian National University  
 Yaroslav Mudryi National Law University  
 Zhytomyr Polytechnic State University



© Авторы тез, 2025  
 © Golden Quill Publishing, 2025  
 © Center for financial-economic research, 2025  
 © International Center of Social Sciences and Humanities, 2025  
**Official website: <http://www.economics.in.ua>**

**International Scientific and Practical Conference “Science, Education and Society: Interdisciplinary Approaches to Solving Global Issues”**

<b>Колибельнікова Л. С.</b> БЕЗПЕКОВІ РИЗИКИ ВІД КОГНІТИВНОГО ВИКРИВЛЕННЯ КВАЛІФІКАЦІЇ ПРАЦІВНИКІВ У ВИРОБНИЧОМУ СЕРЕДОВИЩІ .....	125
<b>SECTION 9. TRANSPORTATION AND LOGISTICS</b> .....	128
<b>Становий О. С.</b> ІНТЕЛЕКТУАЛЬНА АНАЛІТИКА ПОПИТУ: СИНЕРГІЯ ЛОГІСТИКИ, ІТ ТА ЕКОНОМІКИ ПОВЕДІНКИ .....	128
<b>SECTION 10. INFORMATION TECHNOLOGY AND CYBERNETICS</b> .....	130
<b>Краліна Г. С., Левченко В. В.</b> ДИСЦИПЛІНА “АРХІТЕКТУРА КОМП’ЮТЕРА” В ПІДГОТОВЦІ ФАХОВИХ МОЛОДШИХ БАКАЛАВРІВ ЗА СПЕЦІАЛЬНІСТЮ “ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ”: ПРОБЛЕМИ, ВИКЛИКИ, ПЕРСПЕКТИВИ .....	130
<b>Пількевич І. А., Жуков А. О., Мірошніченко С. І.</b> АВТОМАТИЗАЦІЯ ОЦІНЮВАННЯ РІВНЯ ОСІБ, ЯКІ НАВЧАЮТЬСЯ .....	135
<b>Старих О. В.</b> СЕМАНТИЧНА МОДЕЛЬ ЗАДАЧІ РЕКОМЕНДАЦІЇ РОБОЧИХ ВАКАНСІЙ .....	137
<b>SECTION 11. MILITARY SCIENCES, NATIONAL SECURITY, STATE BORDER SECURITY</b> .....	139
<b>Sviatskyi V. V.</b> MODERNIZATION OF THE EDUCATIONAL PROCESS IN SECURITY DISCIPLINES IN THE CONTEXT OF MILITARY CHALLENGES .....	139
<b>SECTION 12. CULTURE, ART AND CULTURAL STUDIES</b> .....	142
<b>Уманець О. В., Нанай І. С.</b> УКРАЇНСЬКА КУЛЬТУРА В ЄВРОІНТЕГРАЦІЙНИХ ГОРИЗОНТАХ .....	142
<b>SECTION 13. TOURISM AND HOTEL AND RESTAURANT BUSINESS</b> .....	145
<b>Антоненко В. С., Ковтун В. В.</b> ЛОГІСТИЧНИЙ АСПЕКТ ЕФЕКТИВНОГО УПРАВЛІННЯ ТОВАРНИМИ ЗАПАСАМИ ПІДПРИЄМСТВА РЕСТОРАННОГО БІЗНЕСУ .....	145
<b>Мельникович М. С., Подолян М. І.</b> ВЕБСАЙТ РЕСТОРАНУ ЯК ОБ’ЄКТ АДМІНІСТРАТИВНО-ПРАВОВОГО РЕГУЛЮВАННЯ .....	150

УДК 004.8

Старих О. В.  
студент VI курсу  
Західноукраїнський національний університет

### СЕМАНТИЧНА МОДЕЛЬ ЗАДАЧІ РЕКОМЕНДАЦІЇ РОБОЧИХ ВАКАНСІЙ

Сучасний ринок праці є надзвичайно динамічним і характеризується постійним зростанням кількості вакансій, які щоденно публікуються на різних онлайн-платформах. У таких умовах традиційні методи пошуку роботи, засновані на ручному відборі та перегляді великого обсягу оголошень, стають малоефективними та витратними за часом [1; 2]. Існуючі агрегатори вакансій зазвичай обмежуються наданням інструментів фільтрації та пошуку за ключовими словами, але рідко впроваджують механізми глибокої персоналізації, здатні враховувати індивідуальні особливості користувача, його попередній досвід та навички. Огляд сучасних рекомендаційних технологій показав, що інтеграція інтелектуальних методів у системи пошуку роботи здатна значно покращити якість та відповідність результатів [1; 2; 3; 4].

Метою даної роботи є аналіз та обґрунтування моделей даних та методів семантичного аналізу текстів, що є теоретичним підґрунтям для розробки інтелектуального методу рекомендації робочих вакансій, який поєднує автоматизоване збирання пропозицій з механізмом персоналізованого ранжування [1; 3; 4].

Для розробки інтелектуального методу необхідно створити формальну модель, що описує сутність об'єктів системи та взаємозв'язки між ними. Нехай  $U$  – множина користувачів, а  $V$  – множина вакансій. Кожен користувач  $u \in U$  має профіль  $p_u$ , що містить структуровані ознаки та текстові дані (ключові навички, опис досвіду). Кожна вакансія  $v \in V$  описується текстом  $d_v$  (обов'язки, вимоги) і метаданими  $a_v$  (галузь, зарплата, локація). Завдання полягає у побудові функції релевантності  $s(u, v)$ , яка визначає ступінь відповідності вакансії профілю користувача [1].

Для підвищення якості рекомендацій запропоновано двоетапну гібридну модель.

1. Етап відбору кандидатів (Candidate Generation). На цьому етапі застосовується двовежова модель (dual-encoder architecture), яка швидко обробляє мільйони вакансій і формує набір із 100-300 потенційно релевантних пропозицій (Candidate Set).

2. Етап перевпорядкування (Re-ranking). На цьому етапі виконується точне ранжування відібраних кандидатів. Для кожної пари  $(u, v)$  формується детальний вектор ознак  $x_{uv}$ , що включає семантичну подібність, відповідність структурних характеристик (локація, рівень, зарплата) та поведінкові ознаки (кліки, відгуки, взаємодії користувача).

У розробленій системі використовується три типи даних: текстові (описи вакансій і резюме), структуровані (локація, рівень досвіду, формат роботи) та поведінкові (кліки, перегляди, відгуки).

Ефективність рекомендаційної системи значною мірою залежить від того, наскільки точно вона здатна представити текстову інформацію у числовій формі. Процес подання тексту у вигляді вектора називається векторизацією [3; 5].

Класичні підходи, такі як TF-IDF (Term Frequency – Inverse Document Frequency), базуються на припущенні, що найважливішими є ті слова, які часто зустрічаються в одному документі, але рідко – у всій колекції. Вдосконаленням

**International Scientific and Practical Conference “Science, Education and Society: Interdisciplinary Approaches to Solving Global Issues”**

цього методу є BM25, який додатково враховує довжину документа та нормалізує вплив дуже частих слів, що показує вищу точність у пошукових системах [6; 7].

Однак ці методи не враховують семантичні зв'язки між словами. Наприклад, терміни «developer» і «programmer» вони трактують як абсолютно різні, хоча їхнє значення близьке. Щоб усунути цю проблему, використовуються семантичні моделі подання текстів, зокрема Sentence-BERT – сучасна нейронна модель для створення векторів речень і цілих текстів, що зберігає їх семантику. Це дозволяє системі обчислювати подібність між описом вакансії та профілем користувача на основі їхнього змісту, а не лише збігу ключових слів[8; 9; 10].

Векторний профіль користувача  $\psi_u$  формується шляхом агрегування (наприклад, усереднення) векторних представлень окремих текстових елементів його профілю, таких як резюме, перелік навичок та історія пошукових запитів.

У результаті виконаного теоретичного дослідження було сформульовано математичну модель задачі рекомендації робочих вакансій та обґрунтовано доцільність використання двоетапної гібридної моделі. Аналіз методів представлення текстових даних показав, що класичні підходи (TF-IDF, BM25) є ефективними для базового пошуку, але не враховують семантичні зв'язки. Встановлено, що використання сучасних семантичних моделей, як-от Sentence-BERT, забезпечує значно вищу якість порівняння описів вакансій і профілів кандидатів. Це дозволяє підвищити точність рекомендацій та є ключовим для побудови інтелектуальної системи, що адаптується під потреби користувача, а не лише реагує на ключові слова.

#### **Список літератури**

1. Recent Developments in Recommender Systems: A Survey / Y. Li та ін. arXiv. 22.06.2023. URL: <https://arxiv.org/abs/2306.12680> (дата звернення: 01.11.2025).
2. Revisiting recommender systems: an investigative survey / O. A. S. Ibrahim та ін. *Neural Computing and Applications*, 2025, Т. 37, С. 2145–2173. URL: <https://doi.org/10.1007/s00521-024-10828-5>.
3. Embedding in Recommender Systems: A Survey / M. Wang та ін. arXiv. 22.10.2025. URL: <https://arxiv.org/abs/2310.18608> (дата звернення: 01.11.2025).
4. Gheewala S., Xu S., Yeom S. In-depth survey: deep learning in recommender systems—exploring prediction and ranking models, datasets, feature analysis, and emerging trends. *Neural Computing and Applications*, 2025, Т. 37, С. 10875–10947. URL: <https://doi.org/10.1007/s00521-024-10866-z>.
5. Farivar K. Semantic Search for Information Retrieval. arXiv. 25.08.2025. URL: <https://arxiv.org/html/2508.17694> (дата звернення: 01.11.2025).
6. Sparck Jones K. A Statistical Interpretation Of Term Specificity And Its Application In Retrieval. *Journal of Documentation*. 1972.
7. Robertson S. E., Zaragoza H. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends® in Information Retrieval*. 2009.
8. Distributed Representations of Words and Phrases and their Compositionality / T. Mikolov та ін. *Advances in Neural Information Processing Systems*. 2013.
9. Enriching Word Vectors with Subword Information / P. Bojanowski та ін. *Transactions of the Association for Computational Linguistics*. 2017.
10. Reimers N., Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 2019.

МАТЕРІАЛИ ІХ МІЖНАРОДНОЇ  
СТУДЕНТСЬКОЇ НАУКОВОЇ  
**КОНФЕРЕНЦІЇ**

ГЛОБАЛІЗАЦІЯ НАУКОВИХ  
ЗНАНЬ: МІЖНАРОДНА  
СПІВПРАЦЯ ТА ІНТЕГРАЦІЯ  
ГАЛУЗЕЙ НАУК



М. ЧЕРКАСИ, УКРАЇНА

**7 Листопада**  
**2025 рік**

**МОЛОДІЖНА  
НАУКОВА  
ЛІГА** 

МАТЕРІАЛИ ІХ МІЖНАРОДНОЇ  
СТУДЕНТСЬКОЇ НАУКОВОЇ

**КОНФЕРЕНЦІІ**

.....

**ГЛОБАЛІЗАЦІЯ НАУКОВИХ  
ЗНАНЬ: МІЖНАРОДНА  
СПІВПРАЦЯ ТА ІНТЕГРАЦІЯ  
ГАЛУЗЕЙ НАУК**

.....

м. Черкаси, Україна

7 листопада 2025 рік

Вінниця, Україна  
«UKRLOGOS Group»  
2025

**УДК 082:001  
Г 52**



Голова оргкомітету: Коренюк І.О.

Верстка: Білоус Т.В.

Дизайн: Бондаренко І.В.

**Рекомендовано до видання Вченою Радою Інституту науково-технічної інтеграції та співпраці, Протокол № 44 від 06.11.2025 року.**



*Конференцію зареєстровано Державною науковою установою «УкрІНТЕІ» в базі даних науково-технічних заходів України та бюлетені «План проведення наукових, науково-технічних заходів в Україні» (Посвідчення № 455 від 10.06.2025).*

*Матеріали конференції знаходяться у відкритому доступі на умовах ліцензії Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0).*

Г 52

**Глобалізація наукових знань: міжнародна співпраця та інтеграція галузей наук: матеріали ІХ Міжнародної студентської наукової конференції, м. Черкаси, 7 листопада, 2025 рік / ГО «Молодіжна наукова ліга». — Вінниця: ТОВ «УКРЛОГОС Груп», 2025. — 520 с.**

ISBN 978-617-8312-99-2

DOI 10.62732/liga-inter-07.11.2025

Викладено матеріали учасників ІХ Міжнародної мультидисциплінарної студентської наукової конференції «Глобалізація наукових знань: міжнародна співпраця та інтеграція галузей наук», яка відбулася 7 листопада 2025 року у місті Черкаси, Україна.

**УДК 082:001**

**ISBN 978-617-8312-99-2**

© Колектив учасників конференції, 2025

© ГО «Молодіжна наукова ліга», 2025

© ТОВ «УКРЛОГОС Груп», 2025

Глобалізація наукових знань: міжнародна співпраця та інтеграція галузей наук

## СЕКЦІЯ 15. ЕЛЕКТРОНІКА ТА ТЕЛЕКОМУНІКАЦІЇ

ІНФОРМАЦІЙНО АНАЛІТИЧНЕ ЗАБЕЗПЕЧЕННЯ ПІДРОЗДІЛІВ  
КРИМІНАЛЬНОЇ ПОЛІЦІЇ З РОЗКРИТТЯ ЗЛОЧИНІВ ПОВ'ЯЗАНИМ З  
НЕЗАКОННИМ ОБІГОМ ЗБРОЇ  
Мілодан М.В., *Науковий керівник: Прокопов С.О.* ..... 275

## СЕКЦІЯ 16. КОМП'ЮТЕРНА ТА ПРОГРАМНА ІНЖЕНЕРІЯ

ENGLISH AS A KEY SKILL FOR REMOTE IT PROFESSIONALS  
Moskalchenko D., *Scientific supervisor: Volobioeva A.O.* ..... 277

ВЕБСАЙТ УЧИТЕЛЯ ЯК ЗАСІБ КОМУНІКАЦІЇ ТА ВЗАЄМОДІЇ З БАТЬКАМИ  
УЧНІВ  
Вознюк А.Б., *Науковий керівник: Суринович О.М.* ..... 279

ВИКОРИСТАННЯ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ У ЗАДАЧАХ  
АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ ДЛЯ ПІДВИЩЕННЯ ЯКОСТІ  
ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ  
Гордієнко Ю.Ю. .... 281

РОЛЬ ТА МОЖЛИВОСТІ ПЛАТФОРМИ NET У СТВОРЕННІ КРИТИЧНО  
ВАЖЛИВИХ ІНФОРМАЦІЙНИХ СИСТЕМ ДЛЯ КОРПОРАТИВНОГО СЕРЕДОВИЩА  
Співак А.М. .... 284

## СЕКЦІЯ 17. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ

AUTOMATED SYSTEM FOR CALCULATING PAYMENTS TO MILITARY  
PERSONNEL  
Matiko Kh.F., *Scientific supervisor: Rybchak Z.L.* ..... 287

АРХІТЕКТУРА ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ РЕКОМЕНДАЦІЇ ВАКАНСІЙ  
Старих О.В., *Науковий керівник: Загородня Д.І.* ..... 290

ВІЗУАЛІЗАЦІЯ ІСТОРИЧНИХ ПРОЦЕСІВ ЗАСОБАМИ ЦИФРОВИХ ТЕХНОЛОГІЙ  
Драбич Д.І., Марцін В.І. .... 293

ДЖЕРЕЛА ІНФОРМАЦІЇ ТА ІНСТРУМЕНТИ ПОШУКУ ІНФОРМАЦІЇ  
ПРАЦІВНИКАМИ КРИМІНАЛЬНОЇ ПОЛІЦІЇ ПІД ЧАС ВИКОРИСТАННЯ  
ТЕХНОЛОГІЙ  
Коваль А., *Науковий керівник: Прокопов С.О.* ..... 295

ДОСЛІДЖЕННЯ МЕТОДУ DENSE RETRIEVAL ЯК ЗАСОБУ ПОКРАЩЕННЯ  
РОБОТИ LLM СИСТЕМ У ТЕХНОЛОГІЇ RAG  
Уткін Є.І., *Науковий керівник: Тітова О.В.* ..... 297

ДОСЛІДЖЕННЯ МЕТОДУ VISION TRANSFORMER (ViT) ДЛЯ ВІДСТЕЖЕННЯ  
РУХІВ РУК ЛЮДИНИ  
Черепов Д.Е., *Науковий керівник: Маїталір С.В.* ..... 300

ЗАСТОСУВАННЯ ГЛИБОКИХ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ ДО ЗАДАЧІ  
КЛАСИФІКАЦІЇ ЕМОЦІЙ В ЗОБРАЖЕННЯХ  
Цісаренко О., *Науковий керівник: Руденко Д.О.* ..... 303

## Глобалізація наукових знань: міжнародна співпраця та інтеграція галузей наук

**Старих Олег Васильович**, здобувач вищої освіти факультету комп'ютерних інформаційних технологій

*Західноукраїнський національний університет, Україна*

**Науковий керівник: Загородня Діана Іванівна**, канд. техн. наук, доцент, доцент кафедри інформаційно-обчислювальних систем і управління

*Західноукраїнський національний університет, Україна*

## АРХІТЕКТУРА ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ РЕКОМЕНДАЦІЇ ВАКАНСІЙ

Сучасні онлайн-платформи з пошуку роботи оперують величезними обсягами даних, що щоденно оновлюються. У таких умовах постає завдання не лише знайти відповідні вакансії, а й персоналізувати результати відповідно до особливостей кожного користувача. Традиційні підходи, що базуються на простій текстовій схожості або сортуванні за датою публікації, не враховують контексту, професійного досвіду та поведінкових характеристик кандидатів.

Розвиток технологій машинного навчання та обробки природної мови відкриває можливості створення інтелектуальних систем рекомендацій, здатних аналізувати семантичні зв'язки між описами вакансій і профілями користувачів. Проте ефективність таких систем значною мірою залежить від архітектури, яка визначає взаємодію між компонентами збору, обробки та ранжування даних, а також від алгоритму персоналізованого відбору, що враховує індивідуальну поведінку користувача.

Метою даної роботи є проектування гнучкої та масштабованої архітектури інтелектуального методу рекомендації вакансій та розробка алгоритму персоналізованого ранжування, що базується на методах машинного навчання та забезпечує підвищення релевантності результатів для кожного окремого користувача.

Розробка інтелектуального методу передбачає створення архітектури, яка забезпечує збирання, обробку, зберігання та аналіз великих обсягів інформації. Система побудована за модульним принципом і складається з таких основних компонентів:

– Модуль збору вакансій: виконує автоматичний збір даних із відкритих джерел.

– Модуль обробки текстів: здійснює очищення, нормалізацію та векторизацію текстів вакансій (за допомогою Sentence-BERT) [1].

– База даних (MongoDB): використовується як єдине сховище для всіх типів даних.

– Модуль рекомендацій: реалізує двоетапний гібридний підхід (відбір кандидатів та перевпорядкування).

– Інтерфейс користувача та API (Web UI / REST API): забезпечує взаємодію з користувачем через FastAPI[2].

Особливий акцент зроблено на виборі MongoDB як єдиного сховища. Цей підхід спрощує архітектуру, оскільки MongoDB дозволяє зберігати в одній колекції структуровані дані (локація, зарплата), текстові описи та векторні представлення (embeddings) [3].

Ключовою перевагою є підтримка нативного векторного пошуку за допомогою оператора  $\$vectorSearch$ . Це дозволяє виконувати швидкий семантичний пошук схожих вакансій без необхідності впровадження зовнішніх векторних індексів [3].

Процес обробки запиту користувача складається з чотирьох основних етапів:

1. Отримання векторного профілю користувача  $\psi(u)$ ;
2. Векторний пошук у колекції  $vacancies$  для відбору кандидатів;
3. Персоналізоване перепорядкування результатів;
4. Формування JSON-відповіді з топ- $k$  вакансіями.

Ключовою частиною системи є алгоритм, що визначає порядок подання вакансій користувачу. На відміну від звичайного пошуку, персоналізоване ранжування обчислює індивідуальний рейтинг вакансій для кожного користувача. Для цього використовується контекстна інформація: історія переглядів, кліків, взаємодій, локація, рівень досвіду тощо [4, 5].

Для кожної пари (користувач  $u$ , вакансія  $v$ ) обчислюється функція релевантності  $s(u, v)$ . Це не статична формула, а вихідне значення параметризованої моделі машинного навчання  $F_\theta$ , яка навчається на історичних даних.

Обчислення має загальний вигляд:

$$s(u, v) = F_\theta(x_{uv}), \quad (1)$$

де  $x_{uv}$  - це вхідний вектор ознак, що комплексно описує пару (користувач, вакансія) і включає в себе такі ключові компоненти [4, 6, 7]:

- $sim_{emb}(u, v)$  - семантичну схожість векторів профілю та вакансії;
- $g_{struct}(u, v)$  - відповідність структурних характеристик (локація, тип зайнятості);
- $h_{beh}(u, v)$  - поведінкові сигнали (частота відгуків);
- $\rho(v)$  - коефіцієнт часової актуальності вакансії.

Для навчання моделі, що обчислює цю функцію, використовується pairwise-функція втрат Bayesian Personalized Ranking (BPR). Вона базується на ідеї, що вакансія, з якою користувач взаємодіє (позитивний приклад  $v^+$ ), має отримати вищу оцінку, ніж випадкова вакансія, з якою взаємодії не було (негативний приклад  $v^-$ ) [8]. Модель (Gradient Boosting або Logistic Regression) навчається максимізувати різницю між цими оцінками [9].

Фінальний алгоритм персоналізованого ранжування виконується так:

1. Отримати вектор профілю користувача  $\psi(u)$ .
2. Знайти набір кандидатів  $C_u$  (300 вакансій) за допомогою векторного пошуку в MongoDB.
3. Для кожної вакансії  $v \in C_u$  обчислити оцінку  $s(u, v)$  за допомогою навченої BPR-моделі.
4. Відсортувати вакансії за спаданням оцінки  $s(u, v)$ .
5. Повернути топ- $k$  результатів користувачу через REST API.

У результаті роботи спроектовано гнучку та масштабовану архітектуру інтелектуальної системи рекомендації вакансій. Використання MongoDB як єдиного сховища зі вбудованим векторним пошуком дозволяє значно спростити реалізацію системи та забезпечити високу продуктивність семантичного пошуку.

### Глобалізація наукових знань: міжнародна співпраця та інтеграція галузей наук

Розроблений алгоритм персоналізованого ранжування, що базується на моделі BPR, дозволяє системі навчатися на неявних взаємодіях користувача (кліках та переглядах). Це забезпечує високий ступінь адаптивності системи до індивідуальних переваг, що виходить за межі простого співставлення ключових слів.

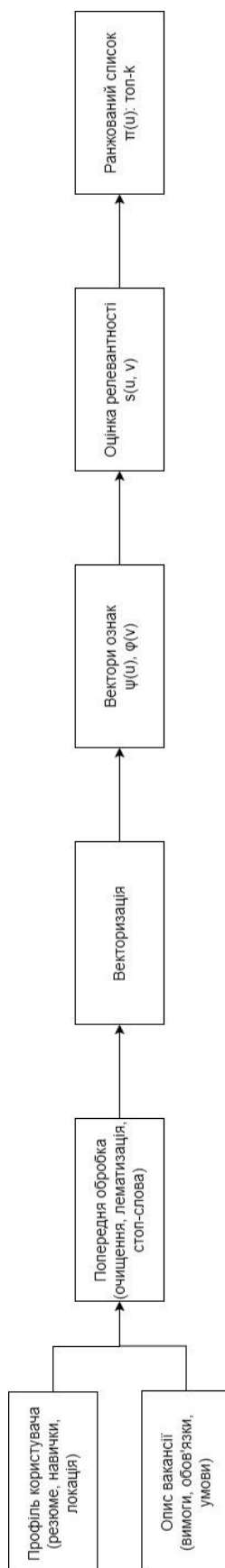
Отримані результати можуть бути використані для інтеграції в сучасні системи електронного працевлаштування, аналітики ринку праці та корпоративні HR-платформи. Поєднання семантичного аналізу на етапі відбору кандидатів та персоналізованого машинного навчання на етапі ранжування є ефективним гібридним підходом для створення сучасних інтелектуальних систем рекомендацій.

#### Список використаних джерел:

1. Reimers N., Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 2019.
2. FastAPI Documentation. URL: <https://fastapi.tiangolo.com/> (дата звернення: 01.11.2025).
3. MongoDB Documentation. URL: <https://www.mongodb.com/docs/> (дата звернення: 01.11.2025).
4. Embedding in Recommender Systems: A Survey / M. Wang та ін. arXiv. 22.10.2025. URL: <https://arxiv.org/abs/2310.18608> (дата звернення: 01.11.2025).
5. Recent Developments in Recommender Systems: A Survey / Y. Li та ін. arXiv. 22.06.2023. URL: <https://arxiv.org/abs/2306.12680> (дата звернення: 01.11.2025).
6. Gheewala S., Xu S., Yeom S. In-depth survey: deep learning in recommender systems—exploring prediction and ranking models, datasets, feature analysis, and emerging trends. Springer. 21.06.2025. URL: <https://link.springer.com/article/10.1007/s00521-024-10866-z> (дата звернення: 01.11.2025).
7. A Survey of Model Architectures in Information Retrieval / Z. Xu та ін. arXiv. 26.08.2025. URL: <https://arxiv.org/html/2502.14822v2> (дата звернення: 01.11.2025).
8. Rendle S. та ін. BPR: Bayesian Personalized Ranking from Implicit Feedback. UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. 2009.
9. Pedregosa F. та ін. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2012.

## Додаток Б

## Потік формування ознак для профілю користувача та вакансії



## Додаток В

## Архітектура двоетапної гібридної моделі рекомендацій

