

Вінницький національний технічний університет

(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматики

(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних наук

(повна назва кафедри (предметної, циклової комісії))

Бакалаврська дипломна робота на тему:

**ІНТЕЛЕКТУАЛЬНИЙ МОДУЛЬ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ
В БАЗАХ ДАНИХ**

Виконав: студент 4 курсу, групи 1КН-206
спеціальності 122 – Комп'ютерні науки

(шифр і назва напрямку підготовки, спеціальності)



Коханівський А.П.

(прізвище та ініціали)

Керівник: к.т.н., професор каф. КН



Савчук Т.О.

(прізвище та ініціали)

« 07 » 06 2024 р.

Рецензент: к.т.н., професор кафедри АІТ

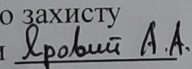


Паламарчук Є. А.

(прізвище та ініціали)

« 07 » 06 2024 р.

Допущено до захисту

Зав. кафедри  Яровий А.А.

« 07 » 06 2024 р.

Вінниця ВНТУ – 2024 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних наук
Рівень вищої освіти перший (бакалаврський)
Галузь знань – 12 “Інформаційні технології”
Спеціальність – 122 “Комп'ютерні науки”
Освітньо-професійна програма – Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри КН
проф., д.т.н. Яровий А.А.

“ 04 ” 03 2024 р

**ЗАВДАННЯ
НА БАКАЛАВРСЬКУ ДИПЛОМНУ РОБОТУ СТУДЕНТУ**

Коханівському Антону Павловичу

1. Тема роботи: Інтелектуальний модуль пошуку різноформатних матеріалів в базах даних

Керівник роботи: професор Савчук Тамара Олександрівна.

Затверджені наказом вищого навчального закладу “ 11 ” 03 року № 80

2. Термін подання студентом роботи 24.05.2024

3. Вихідні дані до роботи :

- Текстова строка розміром не більше як 2000 символів.
- Кількість ваг характеристик – 4.
- Кількість наборів даних не менше 40 наборів даних, що містять назви матеріалів.
- Тип формату файлів – аудіо або текстовий.
- Мова програмування – об'єктно орієнтована.

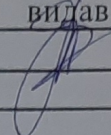
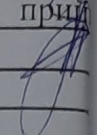
4. Зміст текстової частини (перелік питань, які потрібно розробити):

Вступ; Аналіз існуючих засобів пошуку різноформатних матеріалів в базах даних; Розробка удосконаленого алгоритму пошуку різноформатних матеріалів в базах даних; Розробка структури інтелектуального модуля пошуку різноформатних матеріалів в базах даних; Розробка складових інтелектуального модуля пошуку різноформатних матеріалів в базах даних; Аналіз функціонування інтелектуального модуля пошуку різноформатних матеріалів в базах даних; Висновки; Список використаної літератури.

5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень):

UML-діаграма взаємодії алгоритмів обробки даних з користувачем, Взаємодія алгоритмів пошуку даних з користувачем, Взаємодія між блоками інтелектуального модуля пошуку різноформатних матеріалів в базах даних, Дизайн вебсайту (користувацького інтерфейсу), Приклад фрагмента бази даних реляційної моделі

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-5	Савчук Т.О., професор		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва та зміст етапу	Термін виконання		Прийняв
		початок	закінчення	
1	Аналіз існуючих засобів пошуку різноформатних матеріалів в базах даних	06.05.24	12.05.24	
2	Розробка удосконаленого алгоритму пошуку різноформатних матеріалів в базах даних	12.05.24	14.05.24	
3	Розробка структури інтелектуального модуля пошуку різноформатних матеріалів в базах даних	14.05.24	15.05.24	
4	Розробка складових інтелектуального модуля пошуку різноформатних матеріалів в базах даних	15.05.24	16.05.24	
5	Аналіз функціонування інтелектуального модуля пошуку різноформатних матеріалів в базах даних	16.05.24	14.05.24	
6	Оформлення додатків	14.05.24	21.05.24	
7	Оформлення матеріалів до захисту БДР	21.05.24	06.06.24	

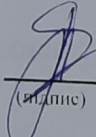
7. Дата видачі завдання 04.03.2024

Студент


(підпис)

Коханівський А. П.
(прізвище та ініціали)

Керівник роботи


(підпис)

Савчук Т. О.
(прізвище та ініціали)

АНОТАЦІЯ

Коханівський А.П. Інтелектуальний модуль пошуку різноформатних матеріалів в базах даних. Бакалаврська дипломна робота складається з 81 сторінок формату А4, на яких є 15 рисунків, список використаних джерел містить 20 найменувань.

В даній бакалаврській дипломній роботі розроблено інтелектуальний модуль пошуку різноформатних матеріалів в базах даних. Визначено об'єкт, предмет та завдання дослідження. Сформульовано мету дослідження. В роботі проаналізовано сучасні системи-аналоги, наведено їх переваги та недоліки, розроблено удосконалений алгоритм процесу пошуку різноформатних матеріалів в базах даних та структурну схему інтелектуального модуля. Запропоновано відповідно рецензію інтелектуального модуля пошуку різноформатних матеріалів в базах даних на рівні його складових.

Результатом дипломного дослідження є модуль у вигляді веб додатка, що реалізована мовою програмування Python у програмному середовищі PyCharm.

Ключові слова: пошук, різноформатні матеріали, алгоритми, нейронні мережі, машинне навчання, семантика, удосконалення.

ABSTRACT

Kokhanivskiy A.P. Intellectual module for searching various format materials in databases. The bachelor's thesis consists of 81 A4 pages, including 15 figure, and the list of used sources contains 20 denominations.

This bachelor's thesis presents the development of an intellectual module for searching various format materials in databases. The object, subject, and research tasks are defined. The research goal is formulated. The thesis analyzes modern analog systems, outlines their advantages and disadvantages, develops an improved algorithm for searching various format materials in databases, and provides a structural diagram of the intellectual module. A corresponding review of the intellectual search module for multi-format materials in databases is proposed at its components level.

The result of the diploma research is a system in the form of a web application implemented in the Python programming language using the PyCharm software environment.

Keywords: search, various format materials, algorithms, neural networks, machine learning, semantics, improvement.

ЗМІСТ

1 АНАЛІЗ ІСНУЮЧИХ ЗАСОБІВ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ	7
1.1 Аналіз сучасного стану пошуку різноформатних матеріалів в базах даних	87
1.2 Аналіз сучасних алгоритмів існуючих пошукових систем	13
1.3 Застосування сучасних підходів підвищення точності шуканих результатів.	17
1.4 Постановка задачі.	19
1.5 Висновки.	20
2 РОЗРОБКА УДОСКОНАЛЕНОГО АЛГОРИТМУ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ	21
3 РОЗРОБКА СТРУКТУРИ МОДУЛЯ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ	26
3.1 Структура серверної частини модуля.	26
3.2 Структура вебінтерфейсу користувача.	30
4 РОЗРОБКА СКЛАДОВИХ ІНТЕЛЕКТУАЛЬНОГО МОДУЛЯ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ	33
4.1 Вибір мови програмування	33
4.2 Вибір середовища програмування та проєктування класів	34
4.3 Розробка модуля бази даних	37
4.4 Розробка інтерфейсу web-сайту	39
4.5 Ініціалізація роботи модулів пошуку	41
5 АНАЛІЗ ФУНКЦІОНУВАННЯ ІНТЕЛЕКТУАЛЬНОГО МОДУЛЯ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ	46
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	55
ДОДАТКИ	57
ДОДАТОК А. ТЕХНІЧНЕ ЗАВДАННЯ	58
ДОДАТОК Б. ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ	

НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ	61
ДОДАТОК В. ІНСТРУКЦІЯ КОРИСТУВАЧА	62
ДОДАТОК Г. ЛІСТИНГ ПРОГРАМИ	63
ДОДАТОК Д. ГРАФІЧНА ЧАСТИНА	68

ВСТУП

Обґрунтування вибору теми дослідження

Розробка інтелектуального модуля пошуку різноформатних матеріалів стає дедалі актуальнішою у зв'язку з тим, що обсяги інформації постійно зростають, а точність пошуку стає все гіршим, що не дозволяє користувачам отримувати бажані результати. Одним із перспективних підходів до вирішення цієї проблеми є використання семантичного аналізу для створення інтелектуальних модулів пошуку різноформатних матеріалів в базах даних. За допомогою семантичного аналізу, можна ефективно опрацьовувати різноманітні типи даних, виявляти приховані закономірності та надавати точні результати.

Наразі існують певні інструменти та сервіси, спрямовані на полегшення пошуку різноформатних матеріалів у базах даних. Однак, вони мають низку обмежень та недоліків. Такі платформи, як WorldCat, BookFinder, SlideShare та Archive.org, хоча й пропонують можливість пошуку книг, презентацій та архівних матеріалів відповідно, проте їхні пошукові алгоритми часто є недосконалими. Вони покладаються лише на співставлення ключових слів та метаданих, не враховуючи глибинний зміст та контекст матеріалів. Крім того, ці платформи зазвичай спеціалізуються на певному типі контенту, що обмежує їхню універсальність.

Зв'язок роботи з науковими програмами, планами та темами

Проведена робота відповідає плану виконання наукових досліджень, розробленому на кафедрі програмного забезпечення.

Мета та завдання дослідження

Метою даного дослідження є підвищення точності пошуку різноформатних матеріалів в базах даних.

Основними завданнями дослідження є:

- Аналіз сучасного стану пошуку різноформатних матеріалів в базах даних;

- Постановка конкретних завдань для реалізації інтелектуального модуля пошуку різноформатних матеріалів в базах даних;
- Розробка удосконаленого алгоритму пошуку різноформатних матеріалів в базах даних;
- Розробка структури модуля пошуку різноформатних матеріалів в базах даних;
- Розробка складових інтелектуального модуля пошуку різноформатних матеріалів в базах даних;
- Аналіз функціонування інтелектуального модуля пошуку різноформатних матеріалів в базах даних;

Ці завдання спрямовані на досягнення поставленої мети та розвиток інноваційного програмного продукту, який відповідає сучасним вимогам у сфері комп'ютерних технологій.

Об'єкт дослідження – процес пошуку різноформатних матеріалів в базах даних.

Предмет дослідження – програмні засоби пошуку різноформатних матеріалів в базах даних.

Методи дослідження

У цій роботі пропонується розглянути створення інтелектуального модуля пошуку різноформатних матеріалів у базах даних на прикладі пошуку книг та фільмів. Сучасні бази даних часто містять величезні обсяги різноманітної інформації, яку важко систематизувати та ефективно опрацьовувати традиційними методами. Застосування нейронних мереж дозволить створити інтелектуальний модуль пошуку, здатну аналізувати зміст, метадані та інші характеристики матеріалів, а також враховувати контекст запиту та персональні уподобання користувачів.

Новизна отриманих результатів

Запропоновано інтелектуальний модуль пошуку різноформатних матеріалів в базах даних, який відрізняється підвищеною точністю за рахунок

застосування технологій аналізу семантичного змісту запиту та матеріалів, що зберігаються в базі даних з подальшим пошуком за семантичним змістом.

Практична цінність отриманих результатів

1. Розроблено удосконалений алгоритм пошуку, що використовує векторні репрезентації запиту та файлів для отримання шуканого результату;
2. Розроблено алгоритм запису різноформатних матеріалів в базу даних;
3. Розроблено вебзастосунок інтелектуального модуля.

Розроблено інтелектуальний модуль пошуку різноформатних матеріалів в базах даних, який відрізняється підвищеною точністю за рахунок застосування технологій аналізу семантичного змісту запиту та матеріалів, що зберігаються в базі даних з подальшим пошуком за семантичним змістом.

Особистий внесок здобувача

Усі наукові та прикладні результати, викладені у бакалаврській дипломній роботі, отримані автором особисто.

Технічне завдання наведено в додатку А.

Достовірність отриманих результатів засвідчується узгодженістю кінцевого практичного завдання із початковими теоретичними дослідженнями.

Апробація результатів роботи

Результати дослідження були апробовані на міжнародній науково-практичній інтернет-конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2024)», на X Міжнародній науково-практичній конференції «TOPICAL ASPECTS OF MODERN SCIENTIFIC RESEARCH» та на VII Міжнародній науково-практичній конференції «SCIENCE AND SOCIETY: MODERN TRENDS IN A CHANGING WORLD».

Публікації

За результатами бакалаврської дипломної роботи опубліковано 3 тез доповідей на конференціях: «Молодь в науці: дослідження, проблеми, перспективи (МН-2024)» (Вінниця, Україна) [1], «X Міжнародна науково-практична конференція «TOPICAL ASPECTS OF MODERN SCIENTIFIC RESEARCH»» (Токіо, Японія) [2], «VII Міжнародна науково-практична

конференція «SCIENCE AND SOCIETY: MODERN TRENDS IN A CHANGING WORLD»» (Відень, Австрія) [3].

1 АНАЛІЗ ІСНУЮЧИХ ЗАСОБІВ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ

1.1 Аналіз сучасного стану пошуку різноформатних матеріалів в базах даних

Пошук даних є невід'ємною частиною нашого повсякденного життя в сучасному світі, хоча ми часто не усвідомлюємо його вплив. Кожен раз, коли ми шукаємо інформацію в Інтернеті, використовуємо навігаційні додатки чи здійснюємо покупки в Інтернеті, ми покладаємось на системи пошуку даних [4].

Ці системи дозволяють нам швидко знаходити релевантну інформацію серед величезних масивів даних, що значно полегшує наше життя. Уявіть собі, як би ви шукали потрібну книгу в бібліотеці без каталогу чи як би ви знайшли потрібну адресу без карт. Системи пошуку даних виконують цю роль, допомагаючи нам ефективно знаходити те, що нам потрібно.

Крім того, пошук даних відіграє важливу роль у бізнесі та науці. Компанії використовують його для аналізу ринкових тенденцій, вивчення поведінки клієнтів та оптимізації своїх продуктів і послуг. Науковці застосовують пошук даних для виявлення закономірностей, тестування гіпотез та здійснення нових відкриттів.

Однак його вплив не обмежується лише практичною стороною. Він також має соціальні та культурні наслідки. Завдяки легкому доступу до інформації, ми можемо розширювати свої знання, краще розуміти світ навколо нас та спілкуватися з людьми з різних куточків планети.

У сучасному світі важко уявити наше життя без Інтернету та можливості миттєвого пошуку необхідної інформації. Проте ми рідко замислюємось над величезними зусиллями, вкладеними в розробку цих систем пошуку даних.

Саме тому важливо не лише усвідомлювати значущість пошуку інформації, але й розуміти, на якій стадії розвитку перебувають сучасні системи пошуку даних.

Серед поширених підходів, що застосовуються в цій сфері сьогодні слід відзначити такі:

1. Пошук за ключовими словами:

Найпоширенішим і базовим методом пошуку є введення користувачем одного або кількох ключових слів чи фраз для знаходження інформації. Даний метод простий у використанні для користувачів, оскільки не потребує попередніх знань та навичок. Він дозволяє швидкий пошук великих обсягів даних та підходить для знаходження чітко визначеної інформації. З іншого боку, цей метод може бути неточним та не враховувати контекст запиту, а також може повертати нерелевантні результати через свою простоту. Також з цим методом доволі складно здійснювати пошук неструктурованих або різноформатних даних. Основною ідеєю його роботи є те, що пошукова система повертає результати, які містять зазначені слова у своєму вмісті або метаданих [5].

Даний підхід широко використовується як в простих, так і в складних пошукових системах. Його простота у реалізації, ефективність пошуку та гнучкість у використанні роблять його чудовим інструментом. Залежно від потреб та вимог пошукової системи, пошук за ключовими словами може виступати як основним компонентом, так і допоміжним інструментом. У простих системах він може бути єдиним методом пошуку, тоді як у більш складних системах його доцільно комбінувати з іншими підходами.

Загалом, пошук за ключовими словами залишається важливим та широко використовуваним підходом завдяки своїй простоті, ефективності та гнучкості. Однак, для забезпечення високої релевантності результатів пошуку, його часто потрібно доповнювати більш просунутими методами.

2. Використання алгоритмів ранжування:

Сучасні пошукові системи використовують складні алгоритми ранжування, щоб визначити, які сторінки або документи найбільш відповідають запиту користувача, та відобразити їх у порядку релевантності. Даний метод дозволяє покращити точність результатів пошуку та враховує різноманітні фактори, такі як авторитетність джерела та структура вебсторінки. Він також

дозволяє користувачам отримувати найбільш корисну інформацію на початку списку результатів, що робить процес пошуку зручнішим та легшим для них. До недоліків можна віднести складність алгоритмів, яка може знизити швидкодію пошуку, а також ризик упередженості або маніпуляцій ранжуванням, що призводить до зниження точності. Даний метод працює за рахунок розрахунку та співставлення факторів, таких як відповідність ключовим словам, авторитетність джерела, структура вебсторінки тощо, з подальшим ранжуванням результатів та їх відображенням користувачеві [5].

Цей алгоритм є доцільним для використання в інформаційних системах із великими обсягами даних з метою досягнення оптимальних результатів пошуку. Саме в таких умовах можливе максимальне розкриття його потенціалу. Також, використання згаданого алгоритму як допоміжного інструменту дозволить підвищити точність отриманих результатів завдяки ширшому охопленню пошукового простору. Проте, слід зважати на те, що подібні системи висувають високі вимоги до обчислювальних ресурсів, що може спричинити перевантаження та виникнення потенційних проблем на рівні серверної або клієнтської частини.

3. Персоналізований пошук:

Головною метою персоналізованого пошуку є адаптування результатів пошуку під конкретного користувача. Даний метод характеризується адаптивністю результатів пошуку до індивідуальних потреб користувача та можливістю покращити релевантність результатів конкретної особи. Він надає дещо різні результати при повторюваних запитах, що урізноманітнює процес пошуку та дозволяє, з відносною легкістю, знаходити потенційно цікаву інформацію для людини. Проте існує проблема з питанням конфіденційності даних користувача та конфіденційності загалом, оскільки програма будує портрет людини, який може в майбутньому бути використаний з корисливими цілями проти самого користувача. Цей метод дозволяє адаптувати результати пошуку під конкретного користувача, за рахунок аналізу його історії пошуку, уподобань, місцезнаходження та інших персоналізованих даних. Як правило, це

покращує релевантність результатів, але також може призвести до ефекту "фільтр-бульбашки", коли користувачі бачать лише інформацію, яка відповідає їхнім уподобанням [6].

Описуваний підхід набув популярності у зв'язку з постійним зростанням обсягів інформації в мережі Інтернет та необхідністю її фільтрації, з чим він успішно справляється. Завдяки використанню методів штучного інтелекту та статистичних алгоритмів, ми отримуємо можливість відфільтровувати гігантські масиви наявної інформації, відокремлюючи ту частину, що є релевантною для користувача. Існує низка методів імплементації згаданого підходу, проте кожен з них має свої особливості. Рекомендується застосовувати даний підхід у поєднанні з одним або декількома іншими алгоритмами для досягнення оптимальних результатів та мінімізації ризику виникнення ефекту "фільтр-бульбашки" [7].

4. Семантичний пошук:

Сучасні пошукові системи все частіше використовують технології обробки природної мови (NLP), зображень (CV) та семантичного аналізу інших даних для кращого розуміння змісту та контексту запитів користувачів.

Перевагами подібних систем є покращення розуміння контексту та наміру запиту користувача, що в свою чергу допомагає знаходити більш релевантні результати. Також подібний метод доцільно використовувати при складних або багатозначних запитах, оскільки у випадку аналізу його семантики, не зростає складність самого запиту порівняно з коротшим запитом, а навіть навпаки, вона зменшується, оскільки в такому випадку полегшується пошук схожої інформації. Це відбувається через високу кількість інформації й відповідно високий рівень розуміння шуканої інформації.

Через складність алгоритмів та моделей, здатних на подібний аналіз, потреби в ресурсах для даного методу також доволі високі. Додатково, існує можливість помилки через неоднозначність або неправильне трактування запиту. Різниця між іншими традиційними та цим методами пошуку полягає в тому, що замість простого пошуку за ключовими словами, системи намагаються

зрозуміти справжній намір користувача та представити більш релевантні результати на основі семантичного аналізу [6].

Сьогодні семантичний пошук реалізується здебільшого на метаданих, які не завжди можуть точно репрезентувати семантику матеріалу, внаслідок чого користувач може не отримати семантично релевантну інформацію щодо сформульованого запиту. Через складність імплементації та потребу у значних обчислювальних ресурсах, згаданий підхід не знаходить широкого застосування або ж використовуються спрощені алгоритми з метою збереження прийнятної швидкодії та відносної точності пошукових результатів [8].

Вибір відповідного підходу до пошуку інформації має визначатися специфікою самої системи, оскільки для різних цілей доцільним є використання різних алгоритмів. Наприклад, застосування персоналізованого пошуку у додатку для пошуку музичного контенту забезпечує вищу релевантність результатів порівняно з використанням пошуку за ключовими словами.

Вибір конкретного підходу залежить від певних факторів, таких як об'єм даних, характеристики даних та вимоги до продуктивності. В таблиці 1.1 наведено переваги та недоліки кожного з розглянутих класичних підходів пошуку даних в базах даних.

Таблиця 1.1 – Порівняльна таблиця підходів пошуку даних в базах даних

Підходи	Характеристики
1	2
Пошук за ключовими словами	+ Простота використання для користувачів; + Швидкий пошук великих обсягів даних; + Підходить для знаходження чітко визначеної інформації;

Продовження таблиці 1.1

1	2
	– Не враховує контекст або семантику запиту;

	– Може повертати нерелевантні результати; – Складнощі при пошуку неструктурованих або різноформатних даних;
Алгоритми ранжування	+ Покращують релевантність результатів пошуку; + Враховують різноманітні фактори, такі як авторитетність джерела та структуру вебсторінки; + Дозволяють користувачам отримувати найбільш корисну інформацію на початку списку результатів; – Складність алгоритмів може знизити швидкодію пошуку; – Існує ризик упередженості або маніпуляцій ранжируванням;
Персоналізований пошук	+ Адаптує результати пошуку до індивідуальних потреб користувача; + Може покращити релевантність результатів конкретного користувача; + Корисний для часто повторюваних запитів; – Ризик ефекту "фільтр-бульбашки", коли користувачі бачать лише певний тип інформації; – Конфіденційність даних та питання конфіденційності;
Семантичний пошук	+ Покращує розуміння контексту та наміру запиту користувача; + Допомогає знаходити більш релевантні результати; + Корисний для складних або багатозначних запитів; – Складність семантичного аналізу природної мови; – Можливі помилки через неоднозначність або неправильне трактування;

В межах даного дослідження буде доцільно використати саме семантичний пошук через його універсальність та можливість отримання векторної

репрезентації тексту. Що також, теоретично, можна порівняти з векторними репрезентаціями інших форматів.

1.2 Аналіз сучасних алгоритмів існуючих пошукових систем

Існує велика кількість алгоритмів пошуку інформації, починаючи від пошуку числа в масиві та закінчуючи складним переплетенням набору алгоритмів, що доповнюють, покращують або покривають свою сферу пошуку [4].

Розглянемо алгоритми для семантичного аналізу, що використовуються в пошуку інформації сьогодні:

1. Word2Vec:

Алгоритм, що перетворює слова у вектори у просторі високої розмірності таким чином, що схожі слова мають схожі вектори. Word2Vec може бути навчений за допомогою нейронних мереж, таких як Continuous Bag of Words (CBOW) або Skip-gram.

Основною перевагою даного алгоритму є здатність виявляти семантичні зв'язки між словами та представляти їх у векторному просторі. Це дозволяє ефективно виконувати різноманітні завдання обробки природної мови, такі як аналіз тональності, класифікація текстів, машинний переклад тощо. Крім того, Word2Vec може бути легко інтегрований з іншими алгоритмами машинного навчання для покращення їх ефективності.

Недоліками Word2Vec можна вважати необхідність великих обсягів текстових даних для навчання моделі, а також те, що алгоритм не враховує контекст речення та порядок слів, що може призводити до неточностей у деяких випадках. Також, Word2Vec не здатний представляти рідкісні або невідомі слова, які не зустрічалися під час навчання.

Word2Vec працює шляхом навчання нейронної мережі передбачати слова на основі їх контексту (CBOW) або передбачати контекст на основі слова (Skip-gram). Під час навчання, алгоритм ітеративно налаштовує ваги нейронної мережі,

щоб мінімізувати помилку передбачення. Результатом є векторні представлення слів, де семантично схожі слова мають близькі вектори у просторі високої розмірності.

2. Nearest Neighbor Algorithms:

Nearest Neighbor Algorithms – алгоритм, що використовує алгоритми кластеризації, що являються алгоритмами типу навчання без вчителя та використовують відстань або схожість між користувачами або об'єктами для рекомендації нових елементів.

Перевагою Nearest Neighbor Algorithms є їх відносна простота та інтерпретовність. Ці алгоритми можуть бути легко реалізовані та налаштовані для різних доменів та типів даних. Крім того, вони можуть ефективно працювати з нелінійними та складними залежностями між об'єктами або користувачами.

Недоліками Nearest Neighbor Algorithms можна вважати їх відносно високу обчислювальну складність, особливо при роботі з великими наборами даних, оскільки потрібно обчислювати відстані між усіма парами об'єктів або користувачів. Крім того, ці алгоритми можуть бути чутливими до вибору метрики відстані та до наявності викидів (outliers) у даних.

Nearest Neighbor Algorithms працюють шляхом обчислення відстані або схожості між об'єктами або користувачами в багатовимірному просторі ознак. Найбільш поширеними метриками відстані є Евклідова відстань, відстань Манхеттена та косинусна схожість.

3. Sent2Vec:

Sent2Vec - це узагальнення алгоритму Word2Vec, який генерує векторні репрезентації для речень замість окремих слів. Він враховує контекст речення і відношення між словами у ньому.

Основною перевагою Sent2Vec є здатність представляти семантику цілих речень у векторному просторі, що дозволяє ефективно виконувати різноманітні завдання обробки природної мови на рівні речень, такі як аналіз тональності, класифікація текстів, кластеризація документів тощо. Sent2Vec також може

враховувати порядок слів та синтаксичні зв'язки в реченні, що дозволяє отримувати більш точні та контекстуально-залежні представлення.

Недоліками Sent2Vec можна вважати необхідність великих обсягів розмічених даних на рівні речень для навчання моделі, а також відносно високу обчислювальну складність у порівнянні з Word2Vec. Крім того, Sent2Vec може бути менш ефективним для дуже коротких або дуже довгих речень, де контекстуальна інформація може бути обмеженою або надлишковою.

Sent2Vec працює шляхом навчання нейронної мережі передбачати слова в реченні на основі їх контексту, подібно до Word2Vec. Однак, замість передбачення окремих слів, Sent2Vec навчається передбачати ціле речення на основі його контексту в документі. Після навчання, векторні представлення речень можуть бути отримані шляхом усереднення або об'єднання векторів окремих слів у реченні з урахуванням їх ваг та порядку.

4. LASER Embeddings (Language-Agnostic SEntence Representations):

LASER Embeddings (Language-Agnostic SEntence Representations) - це набір моделей глибокого навчання, розроблених у Facebook AI, які генерують контекстуальні репрезентації речень, враховуючи їх семантику та трансферність між різними мовами.

Основною перевагою LASER є їх здатність генерувати універсальні представлення речень, які є спільними для різних мов. Це дозволяє ефективно передавати знання та моделі між мовами, що особливо корисно для низькоресурсних мов або для завдань міжмовного аналізу тексту. LASER також можуть враховувати контекст та семантику речень, що дозволяє отримувати більш точні та змістовні представлення.

Недоліками LASER можна вважати їх відносно високу обчислювальну складність та потребу у великих обсягах паралельних даних для навчання багатомовних моделей. Крім того, якість представлень LASER може залежати від якості вирівнювання речень між мовами та від наявності специфічних для певної мови явищ, таких як ідіоми або культурні відмінності.

LASER працюють шляхом навчання багатомовної seq2seq моделі на паралельних корпусах текстів. Модель складається з кодувальника (encoder), який перетворює речення з різних мов у спільний векторний простір, та декодувальника (decoder), який генерує переклади речень на цільову мову. Після навчання, кодувальник може бути використаний окремо для отримання векторних представлень речень, які є семантично значущими та трансферними між мовами. Ці представлення можуть бути далі використані для різноманітних завдань обробки природної мови, таких як класифікація текстів, пошук схожих речень, міжмовне порівняння документів тощо.

Дані алгоритми можуть істотно підвищити якість пошуку та користувацький досвід, але вони також вимагають значних обчислювальних ресурсів, ретельного налаштування та навчання на великих наборах даних. Тому їх впровадження слід ретельно планувати та зважувати з урахуванням потреб системи, доступних ресурсів та вимог користувачів.

Таблиця 1.2 – Порівняльна таблиця алгоритмів для семантичного аналізу

Алгоритм	Характеристики
1	2
Word2Vec	+ Здатність виявляти семантичні зв'язки між словами; + Легка інтеграція з іншими алгоритмами ML; + Ефективність у базових NLP завданнях; – Не враховує контекст речення та порядок слів; – Не представляє рідкісні або невідомі слова;
Nearest Neighbor Algorithms	+ Відносна простота; + Легкість реалізації та налаштування; – Висока обчислювальна складність для великих даних;

Продовження таблиці 1.2

1	2
---	---

Nearest Neighbor Algorithms	– Чутливість до вибору метрики відстані; – Чутливість до викидів (outliers) у даних;
Sent2Vec	+ Представлення семантики цілих речень; + Врахування порядку слів і синтаксису; – Вища обчислювальна складність порівняно з Word2Vec; – Менша ефективність для дуже коротких або довгих речень;
LASER Embeddings	+ Генерація універсальних представлень речень; + Трансферність знань між мовами; + Врахування контексту та семантики речень; – Висока обчислювальна складність; – Залежність від якості вирівнювання речень і мовних особливостей; – Потреба у великих обсягах паралельних даних;

Серед наведених алгоритмів існуючих пошукових систем можна виділити алгоритми векторизації Word2Vec та LASER Embeddings, для подальшого використання в розробці інтелектуального модуля, оскільки саме дані алгоритми дозволяють отримати якісні векторні репрезентації слів та тексту відповідно.

1.3 Застосування сучасних підходів підвищення точності шуканих результатів.

Розробка удосконаленого алгоритму пошуку різноформатних матеріалів в базах даних передбачає використання підходів з підвищення точності шуканих результатів.

Розглянемо існуючі підходи підвищення точності:

1. Налаштування параметрів алгоритмів:

Цей метод підвищення точності та інших параметрів системи є доволі простим за рахунок легкості в програмній реалізації. Цей підхід не є універсальним, оскільки не всі алгоритми мають параметри для налаштування. Проте, даний метод дозволяє підвищити релевантність результатів шляхом зміни деяких значень, що значно спрощує та пришвидшує процес покращення системи пошуку. Водночас розробник має бути знайомим з алгоритмом, адже кожен з них має власний набір параметрів, які, хоча і можуть бути схожими, але зазвичай відрізняються через особливості архітектури [8].

2. Модифікація існуючого алгоритму:

Цей метод передбачає внесення змін до вже існуючого алгоритму пошуку з метою покращення його ефективності та точності. Модифікації можуть включати оптимізацію певних кроків алгоритму, додавання нових функцій або зміну методів обробки даних. Цей підхід вимагає глибокого розуміння алгоритму та його внутрішніх механізмів, а також навичок програмування для реалізації необхідних змін. Перевагою цього методу є можливість адаптувати алгоритм до специфічних потреб системи та типів даних. Однак, процес модифікації може бути трудомістким і вимагати ретельного тестування та налагодження для забезпечення стабільності та надійності роботи системи пошуку [9].

3. Об'єднання декількох алгоритмів:

Даний метод передбачає використання кількох алгоритмів пошуку одночасно для отримання більш релевантних результатів. Замість покладання лише на один алгоритм, система комбінує переваги різних методів, щоб компенсувати їхні індивідуальні недоліки та максимізувати точність пошуку [5]. Цей метод може бути доволі складним у реалізації, оскільки потребує ретельної інтеграції та налаштування різних алгоритмів для забезпечення їх злагодженої роботи. Перевагами даного підходу є підвищення якості пошуку за рахунок поєднання сильних сторін декількох алгоритмів, покриття різних типів даних та різноманітних потреб користувачів. Проте, він також має певні недоліки, такі як збільшення обчислювального навантаження та складність налаштування та балансування внесків різних алгоритмів.

В межах даного дослідження буде використано саме метод об'єднання декількох алгоритмів, для об'єднання алгоритмів семантичного аналізу з алгоритмом пошуку за ключовими словами, що дозволить отримувати точні результати й мінімізації можливу похибку.

1.4 Постановка задачі.

Застосування інтелектуального модулю пошуку різноформатних матеріалів в базах даних підвищує точність отримуваних результатів та надає зручний для користувача інтерфейсу, завдяки якому користувач має змогу взаємодіяти з інтелектуальним модулем.

Пошук різноформатних матеріалів в базах даних базується на отриманні єдиної, векторної репрезентації матеріалів та запиту користувача з подальшим порівнянням отриманих векторів шляхом отримання дистанції між векторами. Це дозволяє здійснити пошук за контекстом та знайти найбільш контекстуально схожі матеріали із запитом користувача.

Нехай дано базу даних у якій зберігається інформація про матеріали, їхні векторні репрезентації, ключові слова матеріалів та векторні репрезентації ключових слів матеріалів.

Нехай задано вхідний вектор $X(x_1, x_2, x_3, x_4, x_5)$, де:

x_1 – текстовий запит користувача, що описує бажаний матеріал, який хоче знайти користувач. Запит може містити ключові слова, фрази або речення, що характеризують шуканий контент.

x_2 – додатне число, що характеризує значення ваг характеристики пошуку семантичних залежностей між запитом та назвами шуканих матеріалів.

x_3 – додатне число, що характеризує значення ваг характеристики пошуку семантичних залежностей між запитом та описами шуканих матеріалів.

x_4 – додатне число, що характеризує значення ваг характеристики пошуку залежностей між ключовими словами з запитом та ключовими словами з назв шуканих матеріалів.

x_5 – додатне число, що характеризує значення ваг характеристики пошуку залежностей між ключовими словами з запитом та ключовими словами з опису шуканих матеріалів.

Тоді, задачу пошуку різноформатних матеріалів в базах даних можна подати у вигляді:

$$F(X) = Y,$$

де Y – вихідний вектор, знайдених файлів, що відсортовані у відповідності до схожості з запитом користувача, де перші значення будуть мати найвищий коефіцієнт схожості.

1.5 Висновки.

У межах даного розділу було проаналізовано сучасний стан систем пошуку різноформатних матеріалів, розглянуто вже відомі підходи до пошуку, їх використання та характеристики. Проведено детальний аналіз сучасних алгоритмів пошуку та проаналізовано сфери їхнього застосування й обмеження. Виявлено сучасні способи покращення систем пошуку та проведено їх аналіз. Актуальність розвитку пошукових систем підтверджено.

У наслідок проведеного в даному розділі аналізу, було прийнято рішення розробити алгоритм, що дозволить проводити пошук різноформатних матеріалів в базах даних за рахунку аналізу семантики самих матеріалів та запиту користувача.

2 РОЗРОБКА УДОСКОНАЛЕНОГО АЛГОРИТМУ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ

Удосконалений алгоритм пошуку різноформатних матеріалів в базах даних передбачає розробку двох основних його складових, а саме алгоритм обробки даних та алгоритм пошуку даних. Особливістю удосконаленого алгоритму є підвищена точність та можливість пошуку файлів різних форматів за рахунок використання єдиного стандарту запису інформації у вигляді вектора та порівнянню цих векторів для здійснення пошуку.

Алгоритми обробки даних та пошуку даних мають такий вигляд:

1. Обробка даних:

Для обробки даних потрібно використовувати алгоритми, які можуть призвести різні формати до одного формату. У випадку обробки аудіо файлів та текстових файлів найдоцільніше буде використовувати алгоритми, що перетворюють дані на векторні репрезентації. Для обробки текстових файлів може бути використано алгоритм, побудований на основі алгоритму LASER Embeddings. Для аудіо файлів можна спочатку здобути текст з аудіо за допомогою використання алгоритмів типу Speech2Text, а потім провести обробки тексту з використанням алгоритму LASER Embeddings.

Щоб отримати контекстуальну репрезентацію великого об'єму тексту в межах одного файлу, необхідно застосувати алгоритми, здатні враховувати контекст на рівні речень чи навіть цілих текстів, а не лише окремих слів. Саме тому для вирішення цієї проблеми доцільно використати алгоритм типу Sent2Vec або Doc2Vec.

У межах даної роботи доцільно застосувати саме LASER Embeddings, оскільки цей алгоритм поєднує високу якість контекстуальних репрезентацій речень з легкістю налаштування та можливістю роботи з різними мовами. Хоча він і не враховує контекст на рівні документа, проте для багатьох завдань обробки природної мови репрезентації речень є достатніми.

Розглянемо алгоритм типу Speech2Text, який дозволяє перетворювати вміст аудіофайлів у текстовий формат для подальшої обробки й аналізу. Подібні алгоритми відіграють ключову роль у розробці удосконаленої системи пошуку різноформатних матеріалів у базах даних. Такий алгоритм забезпечить ефективну взаємодію між алгоритмами та сервісами, а також використання інструментів для аналізу аудіо з метою кодування інформації, її зберігання та подальшого використання. Розглянемо алгоритм Conformer Transducer від Nvidia.

Conformer Transducer від Nvidia – архітектура Conformer Transducer заснована на поєднанні переваг двох сучасних архітектур, transformer та conformer. Поєднуючи переваги цих двох підходів, Conformer Transducer забезпечує високу точність розпізнавання мовлення, водночас зберігаючи швидкість його роботи.

Алгоритм Conformer Transducer є найбільш доцільним рішенням завдяки поєднанню високої точності розпізнавання мовлення, ефективного використання обчислювальних ресурсів та швидкої роботи. Ця перевага в точності та швидкості робить Conformer Transducer ідеальним вибором для отримання якісних текстових даних з великих обсягів аудіо, попри потенційну складність налаштування та високі вимоги до обчислювальних ресурсів під час навчання.

Даний набір алгоритмів дозволяє побудувати удосконалений алгоритм обробки різноформатних матеріалів та їх запису в базу даних для подальшого здійснення пошуку за отриманими векторними репрезентаціями матеріалів, який являється одним з основних принципів роботи даного модуля. UML-діаграма взаємодії алгоритмів обробки даних з користувачем зображено на рисунку 2.1.

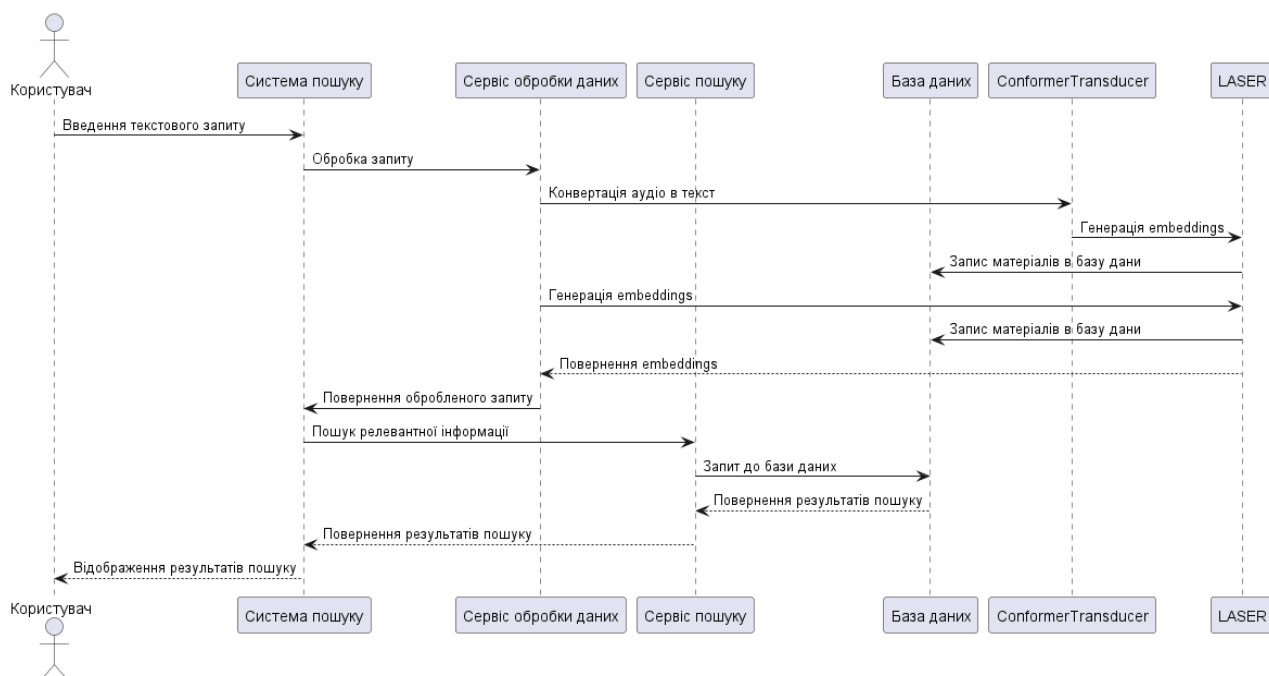


Рисунок 2.1 – UML-діаграма взаємодії алгоритмів обробки даних з користувачем

Як видно з діаграми, дані алгоритми являються одними з основних компонентів інтелектуального модуля, оскільки саме завдяки ним процес пошуку можливий.

Окрім алгоритмів обробки інформації, також важливі алгоритми пошук інформації, тому наступним кроком буде розробка алгоритму пошуку інформації в базі даних, якщо бути точнішим, то потрібно визначити алгоритм, що надаватиме можливість знаходити найбільш схожий вектор до шуканого.

Існує декілька основних алгоритмів для порівняння векторів, але в межах даного дослідження буде використано саме Cosine Similarity.

Cosine Similarity – один з найпростіших алгоритмів для вимірювання схожості між двома векторами. Він обчислює косинус кута між двома векторами у просторі ознак. Чим ближчий кут між векторами до нуля, тим більша їхня схожість.

Оскільки в контексті даної роботи, не потребується складних алгоритмів для пошуку даних, репрезентованих векторами, то найбільш доречним вибором буде Cosine Similarity, на відміну від Euclidean Distance його легше налаштувати

та він обмежується значеннями від -1 до 1, де -1 це повна протилежність й 1 – стовідсоткова схожість, а також Cosine Similarity легше використовувати в контексті великих (1024 значень) векторах ніж K-Nearest Neighbors (KNN).

Для пошуку за ключовими словами буде використано фільтрацію слів по частинах мови у парі з алгоритмом Word2Vec з метою отримання семантичної репрезентації слів, що являється допоміжним інструментом, що дозволить фільтрувати нерелевантні результати, які семантичний пошук оцінив як подібні із запитом через обмежену кількість інформації або типовість запиту даний алгоритм також можна використовувати як основний алгоритм пошуку інформації, оскільки в такій імплементації ми отримуємо семантичний зміст слова, але через обмеженість потенціалу пошуку з використанням лише семантики слова, використання даного підходу буде обмеженим.

Для сортування та відображення результатів буде використано сортування Quick Sort з метою відображення найбільш релевантної інформації на початку списку результатів пошуку.

Даний набір алгоритмів дозволяє побудувати удосконалений алгоритм пошуку різноформатних матеріалів в базах даних, що попередньо пройшли обробку даних та були записані в базу даних з отриманими їхніми векторними репрезентаціями. Даний удосконалений алгоритм являється ще одним з основних принципів роботи даного модуля. UML-діаграма взаємодії алгоритмів пошуку даних з користувачем зображено на рисунку 2.2.

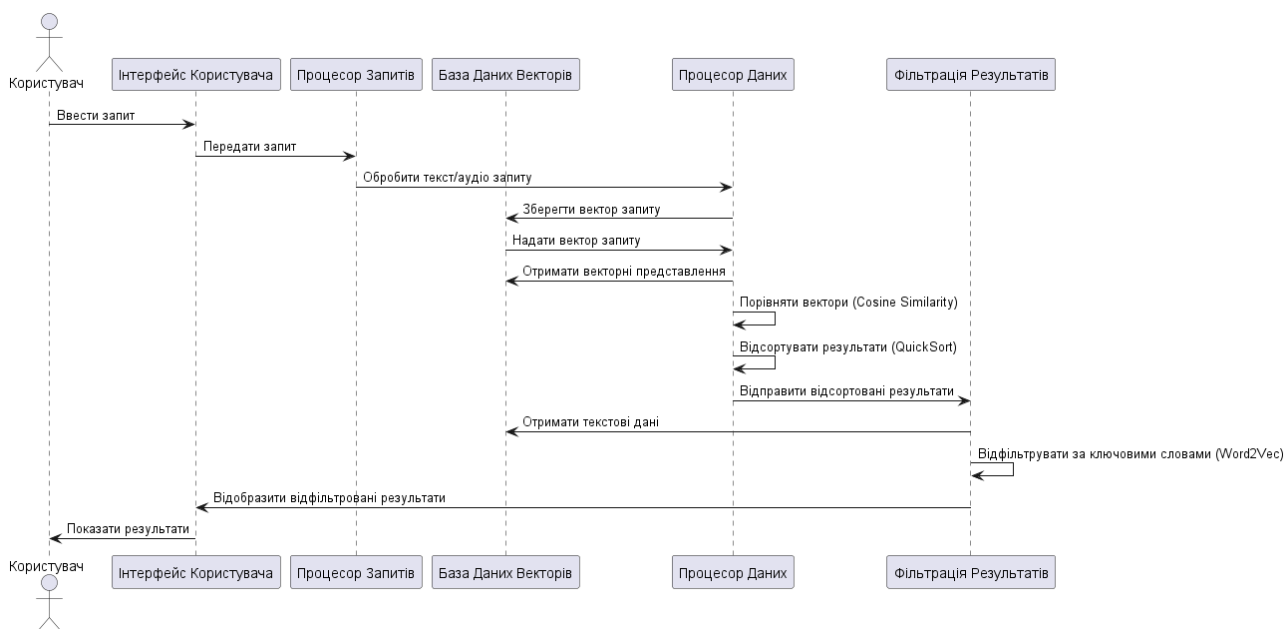


Рисунок 2.2 – UML-діаграма взаємодії алгоритмів пошуку даних з користувачем

Діаграма на рисунку 2.2 ілюструє взаємодію між різними компонентами модуля під час виконання процесу пошуку різноформатних матеріалів у базах даних. Діаграма демонструє потік даних та операцій, що відбуваються в системі від моменту введення запиту користувачем до відображення відфільтрованих результатів.

У цьому розділі було розроблено удосконалений алгоритм пошуку різноформатних матеріалів у базах даних. Також розглянуто взаємодію користувача з різними компонентами інтелектуального модуля та розроблено відповідні UML-діаграми.

3 РОЗРОБКА СТРУКТУРИ МОДУЛЯ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ

3.1 Структура серверної частини модуля.

Структура серверної частини інтелектуального модуля пошуку різноформатних матеріалів в базах даних складається з двох основних сервісів:

- Сервіс обробки даних;
- Сервіс пошуку різноформатних матеріалів у базі даних;

Ці сервіси виконуватимуть такі завдання:

1. Сервіс обробки даних:

- Використання алгоритму Conformer Transducer від Nvidia для виконання Speech-to-Text конвертації для аудіофайлів;
- Для текстових даних використовується алгоритм LASER Embeddings для отримання векторних репрезентацій речень/документів;
- Векторні репрезентації зберігаються в базі даних разом із метаданими файлів;
- Для отримання ключових слів з тексту використано бібліотеку spacy з підключенням моделі “en_core_web_sm”, для отримання ключових слів за алгоритмом NER.
- Для отримання векторних репрезентацій ключових слів використано word2vec з бібліотеки gensim.

2. Сервіс пошуку різноформатних матеріалів:

- Отримання текстового запиту;
- Текстовий запит перетворюється на векторну репрезентацію за допомогою LASER Embeddings;
- Додатково із запиту отримуються ключові слова та переводяться у вектор за допомогою word2vec.

- Векторна репрезентація запиту порівнюється з векторними репрезентаціями в базі даних за допомогою алгоритму косинусної подібності (Cosine Similarity);
- Результати сортуються за значенням косинусної подібності та повертаються користувачеві;

Щоб забезпечити ефективну роботу інтелектуального модуля, необхідно розробити чітку схему передачі даних та взаємодії між блоками. Нижче наведено детальний опис взаємодії між програмними блоками модуля:

1. Блок зміни формату вхідної інформації:

- Отримує вхідні дані від користувача (текст або аудіо).
- Якщо вхідні дані у текстовому форматі, передає їх до блоку обробки ключових слів з тексту та блоку обробки тексту.
- Якщо вхідні дані в аудіоформаті, використовує Conformer Transducer для перетворення аудіо на текст, а потім передає текст до блоку обробки ключових слів та блоку обробки тексту.

2. Блок обробки ключових слів з тексту:

- Отримує текст від блоку зміни формату вхідної інформації.
- Використовує бібліотеку `spacy` з моделлю "en_core_web_sm" для виділення ключових слів з тексту за алгоритмом NER.
- Отримує векторні репрезентації ключових слів за допомогою Word2Vec (бібліотека `gensim`).
- Передає ключові слова та їх векторні репрезентації до блоку запису інформації в базу даних (для нових матеріалів) або до блоку пошуку даних (для запитів користувача).

3. Блок обробки тексту:

- Отримує текст від блоку зміни формату вхідної інформації.
- Використовує алгоритм LASER Embeddings для отримання векторної репрезентації тексту.

- Передає текст та його векторну репрезентацію до блоку запису інформації в базу даних (для нових матеріалів) або до блоку пошуку даних (для запитів користувача).

4. Блок запису інформації в базу даних:

- Отримує оригінальні дані, ключові слова, векторні репрезентації тексту та ключових слів від блоків обробки тексту та обробки ключових слів.
- Зберігає отримані дані в базі даних у відповідній структурі (оригінальні дані, векторна репрезентація, ключові слова, векторні репрезентації ключових слів).

5. Блок пошуку даних:

- Отримує запит користувача у вигляді ключових слів та векторних репрезентацій від блоків обробки тексту та обробки ключових слів.
- Порівнює векторні репрезентації запиту з векторними репрезентаціями матеріалів у базі даних за допомогою алгоритму косинусної подібності (Cosine Similarity).
- Обчислює значення схожості між запитом та кожним матеріалом.
- Передає отримані результати з оцінками схожості до блоку обробки результатів.

6. Блок обробки результатів:

- Отримує результати пошуку з оцінками схожості від блоку пошуку даних.
- Обчислює фінальний коефіцієнт схожості для кожного результату на основі значень схожості для тексту та ключових слів.
- Сортує результати за фінальним коефіцієнтом схожості.
- Передає відсортовані результати до блоку відображення результатів.

7. Блок відображення результатів:

- Отримує відсортовані результати від блоку обробки результатів.
- Відображає результати пошуку через вебінтерфейс для користувача.

Блок зміни формату вхідної інформації є початковою точкою, яка обробляє вхідні дані та передає текст до блоків обробки ключових слів та обробки тексту. Ці два блоки працюють паралельно: один виділяє та векторизує ключові слова, інший векторизує весь текст. Обидва блоки передають свої результати або до блоку запису в БД (для нових матеріалів), або до блоку пошуку (для запитів користувача).

Блок пошуку отримує векторизовані запити від блоків обробки ключових слів та тексту, порівнює їх з даними в БД, та передає оцінки схожості до блоку обробки результатів. Останній обчислює фінальні коефіцієнти, сортує результати та надсилає їх до блоку відображення, який презентує їх користувачеві через вебінтерфейс. Блок запису в БД працює автономно, не передаючи дані іншим блокам.

Така взаємодія між блоками забезпечує ефективний потік даних та обробку інформації на різних етапах, починаючи від отримання вхідних даних від користувача та закінчуючи відображенням релевантних результатів пошуку.

Взаємодія блоків інтелектуального модуля пошуку різноформатних матеріалів в базах даних представлено на рисунку 3.2.

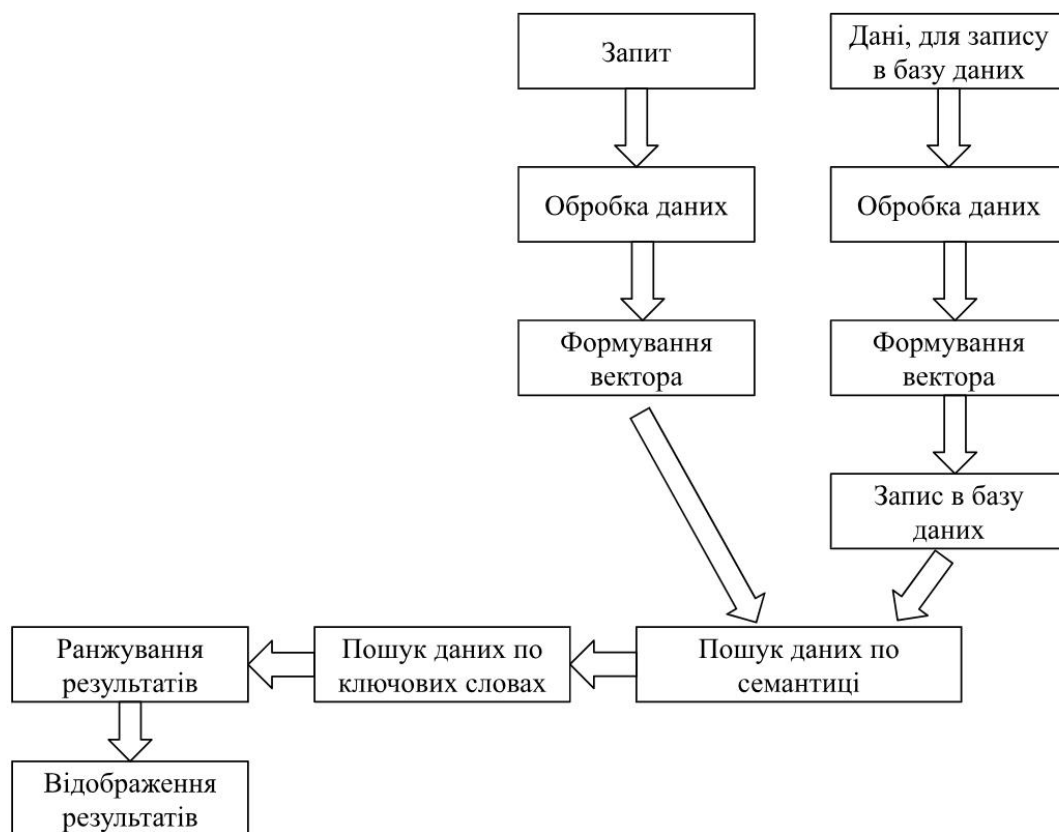


Рисунок 3.2 – UML-діаграма взаємодії між блоками інтелектуального модуля пошуку різноформатних матеріалів в базах даних.

Розроблено структуру серверної частини інтелектуального модуля пошуку різноформатних матеріалів в базах даних. В процесі розробки структури серверної частини модуля було сформовано основні блоки програмного забезпечення та взаємодії між ними в межах модуля.

3.2 Структура вебінтерфейсу користувача.

Для тестування та взаємодії користувача з інтелектуальним модулем розроблено вебінтерфейс користувача. Першим кроком до розробки структури вебінтерфейсу користувача є розробка дизайну користувацького інтерфейсу в програмному забезпеченні Figma.

На рисунку 3.1 зображено дизайн вебсайту (користувацького інтерфейсу), для інтелектуального модуля пошуку різноформатних матеріалів в базах даних.

ІНТЕЛЕКТУАЛЬНИЙ МОДУЛЬ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ

Підготував студент 1КН-20Б
Коханівський Антон Павлович

Введіть запит для пошуку...

Пошук

Налаштування

Скинути

Результати пошуку

#1 Аудіо: Test 1	Опис: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo...	Відповідність: 96.32%
#2 Аудіо: Test 2	Опис: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo...	Відповідність: 94.02%
#3 Текст: Test 3	Опис: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo...	Відповідність: 91.02%

Рисунок 3.1 – Дизайн вебсайту (користувачького інтерфейсу)

Представлений дизайн (рисунок 3.1) візуально має декілька основних компонентів:

1. Поле для запиту:

— Звичайне текстове поле, куди користувач вводить запит;

2. Кнопки керування:

— Пошук – даний елемент запускає основний алгоритм, що дозволяє взаємодію між сервером та клієнтом;

— Налаштування – при натисканні на дану кнопку з правої частини екрана виїжджатиме меню, де можна буде зазначити конфігурації пошуку. Дані елементи потрібні для легкого налаштування та тестування додатка;

— Скинути – дозволяє стерти всі результати пошуку й привести налаштування до тих, що були за замовчуванням;

3. Результати пошуку:

- Результати пошуку представлені у вигляді списку, де можна побачити базову інформацію шуканого результату, а саме: номер, назву, опис, схожість з запитом, а також додаткові деталі при натисканні на елемент;

Подібна структура користувацького інтерфейсу дозволяє не лише прискорити розробку, оскільки інтерфейс є доволі простим і зрозумілим, але й спростити його використання для користувачів. Чіткий і лаконічний дизайн з мінімальною кількістю елементів керування зменшує когнітивне навантаження на користувача і полегшує навігацію та взаємодію з системою [10].

Крім того, такий підхід до побудови сайту забезпечує гнучкість і масштабованість, дозволяючи легко додавати або змінювати функціональність у майбутньому без значних змін у структурі інтерфейсу. Це також полегшує тестування роботи інтелектуального модуля пошуку, оскільки користувацький інтерфейс є відносно простим і зосередженим на основній функції – пошуку інформації.

Загалом, запропонована структура користувацького інтерфейсу є ефективним рішенням, що поєднує простоту, зрозумілість і функціональність, забезпечуючи зручний і ефективний доступ до інтелектуального модуля пошуку різноформатних матеріалів у базах даних.

Розроблена структура інтелектуального модуля пошуку різноформатних матеріалів в базах даних. Було розроблено структуру серверної частини інтелектуального модуля пошуку різноформатних матеріалів в базі даних, визначено основні блоки програмного засобу та їх взаємодію. Також, було розроблено структуру вебінтерфейсу користувача.

4 РОЗРОБКА СКЛАДОВИХ ІНТЕЛЕКТУАЛЬНОГО МОДУЛЯ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ

4.1 Вибір мови програмування

Для розробки інтелектуального модуля пошуку різноформатних матеріалів у базах даних буде використано високорівневу інтерпретовану мову програмування Python з огляду на такі технічні переваги:

1. Великий набір бібліотек для обробки природної мови (NLTK, Gensim, spaCy) та машинного навчання (PyTorch, TensorFlow), що забезпечують можливості для роботи з текстовими даними, векторизації, тренування та застосування нейронних мереж [9, 10].
2. Простий та зрозумілий синтаксис, що спрощує розробку та підвищує читабельність коду, а також полегшує підтримку та розширення проєкту. Крос-платформеність Python забезпечує сумісність з різними операційними системами (Windows, macOS, Linux) [10].
3. Активна спільнота розробників, що забезпечує значну кількість ресурсів, документації, бібліотек та підтримку при вирішенні проблем. Підтримка вебфреймворків (Flask, Django) робить Python зручним для розробки вебдодатків [11].

У рамках дослідження буде використано фреймворк Flask завдяки його легковісності та відсутності жорстких архітектурних обмежень, що ідеально підходить для створення невеликих вебдодатків. Гнучкість Flask дозволяє вибирати бібліотеки та інструменти відповідно до потреб проєкту. Завдяки простоті та легковажності Flask є ідеальним рішенням для невеликих проєктів, на відміну від комплексних фреймворків, як Django. Активна спільнота Flask забезпечує доступ до додаткових бібліотек та ресурсів для швидкої розробки функцій, таких як пошук, кешування та робота з базами даних [12].

Для забезпечення швидкої розробки функціонального вебінтерфейсу буде використано Jijnja2, що дозволить поєднати бекенд-логіку обробки природної

мови та машинного навчання (реалізовану за допомогою бібліотек Python) з фронтенд-частиною вебдодатку.

Для зберігання даних буде використана об'єктно-реляційна система управління базами даних PostgreSQL, яка забезпечує масштабованість, ефективну обробку великих обсягів даних та підтримку SQL. Інтеграція з Python здійснюється через бібліотеку `psycopg2` [12]. PostgreSQL підтримує широкий спектр типів даних (текстові, числові, бінарні, мультимедійні), що дозволяє ефективно зберігати різноформатні матеріали, а також забезпечує потужні механізми безпеки, цілісності даних та високу продуктивність. Основною причиною обрання саме цієї бази даних являється наявність пакета `pgvector`, що дозволяє значно пришвидшити швидкість роботи з векторними масивами, що будуть широко використовуватись в межах даного інтелектуального модуля.

Таким чином, архітектура проєкту складатиметься з Python, фреймворк Flask для веброзробки та PostgreSQL для зберігання даних, що забезпечить гнучку, ефективну та масштабовану реалізацію інтелектуального модуля пошуку різноформатних матеріалів зі зручним вебінтерфейсом.

4.2 Вибір середовища програмування та проєктування класів

Для розробки інтелектуального модуля пошуку різноформатних матеріалів у базах даних буде використано інтегроване середовище розробки (IDE) PyCharm. PyCharm є потужним і зручним середовищем, спеціально розробленим для Python. Воно забезпечує широкий спектр функціональних можливостей, таких як:

- Автоматичне доповнення коду та перевірка синтаксису на льоту, що підвищує продуктивність розробки та зменшує кількість помилок.
- Інтелектуальний рефакторинг коду, що дозволяє легко реструктурувати існуючий код без порушення його функціональності.

- Інтеграція з системами контролю версій, такими як Git, для ефективного управління кодом.
- Налагодження та профілювання коду, що допомагає виявляти та усувати помилки й оптимізувати продуктивність.
- Підтримка різноманітних фреймворків та бібліотек Python, включно з Flask та бібліотеками для обробки природної мови та машинного навчання.

Для проведення аналітики та експериментів з даними, буде використано Google Collaboratory. Це безкоштовне хмарне середовище, яке надає доступ до потужних обчислювальних ресурсів, включаючи графічні процесори (GPU) для прискорення роботи нейронних мереж. Google Collaboratory інтегрується з Google Drive, що спрощує завантаження та зберігання даних, а також дозволяє легко поділитися результатами з іншими учасниками проєкту [13].

Для візуалізації та управління базою даних PostgreSQL буде використано pgAdmin – кросплатформовий графічний інструмент для адміністрування PostgreSQL. pgAdmin забезпечує зручний інтерфейс для створення, модифікації та перегляду баз даних, таблиць, індексів та інших об'єктів бази даних. Він також дозволяє виконувати SQL-запити та переглядати їх результати.

Щодо проєктування класів та архітектури програмного забезпечення, буде використано об'єктно-орієнтований підхід, який дозволяє структурувати код у логічні модулі та забезпечує легкість у підтримці та розширенні проєкту. Класи будуть спроектовані відповідно до принципів SOLID (Single Responsibility, Open/Closed, Liskov Substitution, Interface Segregation, Dependency Inversion), що є загальновизнаними практиками об'єктно орієнтованого програмування [11].

Інтелектуальний модуль буде програмно розділений на 4 сервіси:

1. Користувацький інтерфейс – візуалізація результатів та взаємодія користувача з програмним модулем завдяки використанню графічного вебінтерфейсу, що буде розроблено за допомогою Flask, Jinja2, HTML, CSS та JavaScript.

2. Взаємодія з базою даних – підключення та взаємодія з базою даних на рівні отримання та додавання даних таблиць. Дозволяє отримати дані для здійснення пошуку, тестування та обробки нових даних. Першим кроком до запису нових даних в базу даних є запис сирих даних про об'єкт, такі як назва, опис, дата публікації тощо, щоб надалі здійснити обробку даної інформації та привести її до векторного вигляду й записати векторну репрезентацію об'єктів у відповідну таблицю.
3. Обробка даних – попередня обробка даних для подальшої роботи з ними. Охоплює зміну формату, приведення до вектора, отримання ключових слів.
4. Пошук даних – проведення пошуку даних за запитом користувача шляхом порівняння векторів та зіставлення ключових слів для підвищення точності. При отриманні результатів схожості проводиться обчислення фінального коефіцієнта схожості шляхом обчислення зваженого середнього, за яким буде відбуватись сортування результатів.
Основні класи, що будуть розроблені, включатимуть:
 1. FormatPreprocessing – приведення набору форматів до єдиного формату. В межах даного дослідження буде виконано приведення аудіо в текстовий формат.
 2. DatabaseConnection – клас підключення до бази даних для взаємодії з нею.
 3. DatabaseQuery – логіка здійснення запитів до бази даних для отримання та додавання даних.
 4. TextPreprocessor – обробка тексту для видалення дефектів, підготовки тексту до подальшої роботи та отримання ключових слів з тексту.
 5. TextVectorizer – приведення тексту до векторного вигляду для подальшого здійснення пошуку з використанням отриманих векторних репрезентацій семантичного змісту слова чи речення.
 6. SearchController – клас для пошуку даних, що використовує алгоритми типу косинусної подібності для отримання відстані між векторами, що

репрезентує загальну схожість між двома записами або між записом та запитом користувача.

7. FlaskApp – сервер та графічна частина інтелектуального модуля, що надає змогу користувачам взаємодіяти з програмним забезпеченням.

Ця модульна структура забезпечує гнучкість, масштабованість та легкість у підтримці проєкту, дозволяючи працювати над окремими компонентами паралельно та легко інтегрувати їх у єдиний модуль.

4.3 Розробка модуля бази даних

Першим кроком в програмній розробці інтелектуального модуля пошуку різноформатних матеріалів є розробка модуля бази даних, що передбачає створення структури бази даних та підключення інтелектуального модуля до неї. Буде обрано реляційну базу даних, оскільки структура даних повністю відповідає реляційним представленням [14].

У межах даного дослідження потрібно розробити базу даних, що містить тексти, аудіо, векторні репрезентації текстових описів матеріалів та векторні репрезентації ключових слів [15].

Беручи це до уваги, формується база даних з такими сутностями: Текст, Аудіо, Векторна репрезентація опису, Векторна репрезентація ключових слів [16].

Ці сутності матимуть такі атрибути:

1. Текст:

- materialid – ідентифікатор матеріалу. Тип даних: text.
- title – назва текстового матеріалу. Тип даних: text.
- description – опис текстового матеріалу. Тип даних: text.
- title_keywords – ключові слова з назви текстового матеріалу. Тип даних: text[].

- `description_keywords` – ключові слова з опису текстового матеріалу. Тип даних: `text[]`.
- `title_embedding` – векторна репрезентація назви текстового матеріалу. Тип даних: `float[]`.

2. Аудіо:

- `materialid` – ідентифікатор матеріалу. Тип даних: `text`.
- `title` – назва аудіо матеріалу. Тип даних: `text`.
- `audio_description` – опис текстового матеріалу. Тип даних: `text`.
- `audio_url` – посилання на аудіо файл матеріалу. Тип даних: `text`.
- `title_keywords` – ключові слова з назви аудіо матеріалу. Тип даних `text[]`,
- `audio_text_keywords` – ключові слова з опису аудіо матеріалу. Тип даних: `text[]`
- `title_embedding` – векторна репрезентація назви аудіо матеріалу. Тип даних: `float[]`.

3. Векторна репрезентація опису:

- `materialid` – ідентифікатор матеріалу для отримання посилання на матеріал, з якого було отримано векторні репрезентації. Тип даних: `text`.
- `text_embedding` – векторна репрезентація опису матеріалу. Тип даних: `float[]`,
- `sentence_number` – номер речення векторної репрезентації. Тип даних: `int`.
- `source_type` – тип матеріалу: аудіо, текст. Тип даних: `text`.

4. Векторна репрезентація ключових слів:

- `keyword text` – ключове слово. Тип даних: `text`.
- `keyword_embedding` – векторна репрезентація ключового слова. Тип даних: `float[]`.
- `keyword_type` – тип ключового слова: організація, особа, назва локації й т.д.. Тип даних: `text`.

На рисунку 4.1 зображено приклад фрагмента реляційної моделі таблиці TextMaterials

	materialid text	title text	description text	title_keywords text[]	description_keywords text[]	title_embeddings double precision[]
1	2314769.Mind_Blowing_Mu...	Mind Blowing Music	Tune in to all sorts of cool musical happenings, from the birth of t...	["Mind Blowing Music"]	(Elvis,Mozart)	(-0.0011316685,0.0011510523,0.016226145,0.009237174,0.014929732,0
2	1229867-metal-buttons	Metal Buttons: C.900 BC - C.1700 ...	Among everyday dress-accessories, buttons are probably the mo...	[null]	(second)	(0.010319652,-9.774539e-05,0.018394,0.007743987,0.01386023,0.00122
3	2598001-iron-angel	Iron Angel	In this stunning follow-up to his epic fantasy debut, Alan Campbel...	["Iron Angel"]	("Alan Campbell",Deepgate,Presbyte...	(0.0077385847,-0.00012555871,-0.0001418419,0.009144619,0.01060302
4	15803037-red-sparrow	Red Sparrow	In the grand spy-tale tradition of John le Carré comes this shocki...	["Red Sparrow"]	("John le Carré",CIA,present-day,Rus...	(0.0029003178,-0.0014593872,-0.0026858563,0.007883897,0.014376797
5	13496_A_Game_of_Thrones	A Game of Thrones	Here is the first volume in George R. R. Martin's magnificent cycle...	(Thrones)	(first,"George R. R. Martin's","A Stor...	(0.0015818024,-0.00032430963,0.0012684186,0.0019396789,0.0283233
6	2956.The_Adventures_of_H...	The Adventures of Huckleberry Finn	A nineteenth-century boy from a Mississippi River town recounts ...	(Adventures,"Huckleberry Finn")	(nineteenth-century,"Mississippi Riv...	(0.0066853915,-2.2750113e-05,-0.0008191449,-0.0023576936,0.0313383
7	4981.Slaughterhouse_Five	Slaughterhouse-Five	Selected by the Modern Library as one of the 100 best novels of a...	(Slaughterhouse-Five)	("the Modern Library","one,100,Slaug...	(0.0032120224,0.0039121606,0.0002384712,7.249988e-05,0.01902684,0
8	52036.Siddhartha	Siddhartha	Herman Hesses classic novel has delighted, inspired, and influen...	[null]	("Herman Hesses","Indian Brahmin",...	(-0.0028970144,-0.00021558414,0.0008217392,0.0006281583,0.0040741
9	99107.Winnie_the_Pooh	Winnie-the-Pooh	The adventures of Christopher Robin and his friends in which Poo...	[null]	("Christopher Robin","Pooh Bear","Pig...	(0.006509795,0.00086491054,-0.0021717092,0.009048663,0.044928245
10	17848055-kurathi-mudukku	Kurathi Mudukku		[null]	[null]	(0.0062128897,5.9492554e-06,0.0016610442,0.010274799,-0.000874112
11	12565115-troidmanden-fra...	Troidmanden fra Waterloo	Dansk thriller af internationalt format om genforskningens etiske ...	(Troidmanden,Waterloo)	(De,"uskyldens forfald")	(0.0039789514,-0.0001878849,0.00021503541,0.005480673,0.01198637
12	12557776-frantic	Frantic	Can a deranged serial killer be stopped before it's too late?For ga...	[null]	("Marny Toogood","Marny,Marny")	(-0.0011081946,6.998152e-05,0.000201704,0.005867194,0.002178646,0.1
13	15714701-a-cry-from-egypt	A Cry From Egypt (Promised Land,...	Adventure, excitement, love, and faith come together when Jara...	(Egypt,"Promised Land",1)	(Jara,"Egypt","four hundred years")	(0.008655858,1.03258035e-05,0.002852602,0.0061961478,0.03436801,0
14	14663202-enterrer-la-haine...	Enterrer La Haine Et La Vengeance		("La Haine")	[null]	(0.0028704877,-6.343056e-05,0.0012603059,0.0076038614,0.035010416
15	634636.The_Seventh_Secret	The Seventh Secret	Emily Ashcroft goes to Berlin where her father died under mysteri...	(Seventh)	("Emily Ashcroft","Berlin,Hittler")	(0.019539997,-0.0009380312,0.0018228994,0.014008054,0.011513418,0
16	16052055-return-to-shadow...	Return to Shadow Lake		("Shadow Lake")	[null]	(0.014760548,0.0002025309,-0.00055318745,0.001564304,0.03815561,-0
17	6295.Howl_and_Other_Poe...	Howl and Other Poems		[null]	[null]	(0.04413573,0.0005546847,-0.0010518543,0.0072764093,0.02615409,9.4
18	17826082-ruthy-s-new-texa...	Ruthys New Texas Lawyer Friend	young adult love story—a western	("Lawyer Friend")	[null]	(0.008397739,-7.937122e-06,-0.0017354678,0.0037399076,0.025077531,1
19	385228.On_Liberty	On Liberty	Alternate cover edition of ISBN 9780140432077Published in 185...	[null]	(9780140432077Published,1859,"Jo...	(-0.0006648501,0.00010479229,0.000946152,-4.174588e-05,0.002996882
20	21348.Aesop_s_Fables	Aesops Fables		[null]	[null]	(0.02342464,0.00015830244,-0.005790694,0.003756255,0.03323583,-8.4
21	17184.The_Invisible_Man	The Invisible Man	This masterpiece of science fiction is the fascinating story of Grif...	[null]	(Griffin)	(-0.0005562299,-0.0021373916,-0.00050916575,0.014916419,-0.0045539
22	233649.The_Great_Hunt	The Great Hunt	The Wheel of Time turns and Ages come and pass. What was, wh...	[null]	("The Wheel of Time",Shadow)	(0.0073358985,-0.0006136626,-0.0015283822,0.056796554,0.00185146
23	2498952_	كلية ودعنة	مجموعة حكايات من الأساطير على لسان الحيوانات والطيور	[null]	[null]	(0.00042751356,5.85089e-05,-0.00011082149,0.009182666,0.02644666
24	295621.Light_in_my_Darkin...	Light in my Darkness	One of Times women of the century, Helen Keller, reveals her mys...	(Darkness)	(One,Times,"the century","Helen Kel...	(0.0027517912,0.00031742646,-0.0017961345,0.004915164,0.020114522
25	6905134-eldens-hemlighet	Eldens Hemlighet	När Sofia var tolv år trampade hon på en landmina och miste båd...	[null]	("När Sofia","sina ben","Minan","lyckas ...	(0.021204337,-0.00069105654,-0.0015426305,0.012171982,0.043001894
26	15995904-sweepstakes	Sweepstakes	Henry discovers he has purchased the winning candy bar to the H...	[null]	("the Hungry Huggles Sweepstakes",...	(0.001696429,-0.00057838584,-8.44294e-05,0.0060255136,0.009922656,
27	316943.Selected_Poems	Selected Poems		[null]	[null]	(0.030634357,-0.0005405362,-0.0024448582,0.02164744,0.0031791497,0
28	1711487.Un_certain_sourire	Un certain sourire	Sagans second novel tells the story of Dominique, a bored twenty...	(Un)	(Sagans,second,Dominique,"twenty ...	(0.008994042,-0.0017274839,0.0070413356,0.017892309,-0.016787631,-
29	982250.Blood_Magic	Blood Magic	Her sister was murdered, and Kirin hungers for revenge. Using the...	[null]	(Kirin,Kirin,"Lia Cho",Kirin)	(-0.0013849972,-0.00088120333,-0.0022229005,0.011904814,0.01055281

Рисунок 4.1 – Приклад фрагмента реляційної моделі таблиці TextMaterials

Для підключення до бази даних та отримання доступу до даних на сервері буде розроблено сервіси DatabaseConnection та DatabaseQuery за допомогою rsusorg2, де буде відбуватись підключення до бази даних та відправлення запитів до бази даних відповідно.

Така структура бази даних забезпечує ефективне зберігання різноформатних матеріалів, їх векторних репрезентацій та ключових слів, що є необхідним для реалізації функціональності інтелектуального модуля пошуку.

4.4 Розробка інтерфейсу web-сайту

У розділі 3.1 було описано структуру та дизайн вебінтерфейсу для інтелектуального модуля пошуку різноформатних матеріалів у базах даних. На основі цієї інформації буде розроблено інтерфейс вебсайту з використанням HTML, CSS та JavaScript.

Структура вебсайту складається з таких основних компонентів:

1. Заголовок сторінки з назвою модуля та інформацією про автора.
2. Поле для введення запиту користувачем.
3. Панель з кнопками керування: "Пошук", "Налаштування" та "Скинути".
4. Область для відображення результатів пошуку.

Розглянемо реалізацію кожного компонента більш детально.

Заголовок сторінки та поле для введення запиту реалізовані за допомогою HTML-розмітки та CSS-стилів. Використано семантичні теги, такі як `<header>`, `<h1>`, `<h2>` та `<form>` для забезпечення структурованості та доступності вебсторінки [17].

Панель з кнопками керування створена з використанням HTML-елементів `<button>` та оформлена за допомогою CSS-стилів. Для кожної кнопки визначено окремий клас, що дозволяє застосувати унікальні стилі та обробники подій у JavaScript [18].

Область для відображення результатів пошуку реалізована як HTML-елемент `<div>` з класом `results`. Кожен результат пошуку відображається як окремий блок `<div>` з класом `result-obj`. Ці блоки містять інформацію про назву, опис та коефіцієнт схожості результату з запитом користувача. Для відображення коефіцієнта схожості використано додаткові вкладені блоки з CSS-стилями для забезпечення відповідного візуального представлення [19].

Стилізація вебінтерфейсу здійснена за допомогою CSS-файлу `style.css`. В ньому визначено стилі для всіх компонентів сторінки, включаючи шрифти, кольори, розміри, відступи та інші візуальні властивості [20].

Для забезпечення взаємодії користувача з вебінтерфейсом та обробки подій, таких як натискання кнопок, буде використано JavaScript. Обробники подій будуть додані до відповідних кнопок та полів введення за допомогою JavaScript, що дозволяє виконувати відповідні дії, такі як відправлення запиту на сервер, очищення результатів пошуку та відкриття вікна налаштувань.

Отже, розробка інтерфейсу вебсайту для інтелектуального модуля пошуку різноформатних матеріалів у базах даних здійснюється з використанням HTML, CSS та JavaScript. Структура та дизайн вебінтерфейсу відповідають

вимогам простоти, зрозумілості та функціональності, забезпечуючи зручну взаємодію користувача з модулем пошуку.

4.5 Ініціалізація роботи модулів пошуку

Перед початком використання потрібно провести векторизацію текстів, як зображено на рисунку 4.2.

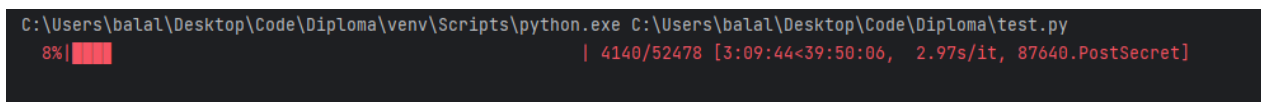


Рисунок 4.2 – Процес векторизації тексту

Як видно на зображенні, це доволі довгий та ресурсомісткий процес, який потрібно проводити під час внесення додаткових матеріалів до бази даних та під час обробки запиту користувача. Приклад векторної репрезентації назв матеріалів зображено на рисунку 4.3.

```
>>> from laserembeddings.laser import Laser
>>> laser = Laser()
>>> laser.embed_sentences("Leaves of Grass", lang="en").reshape(1024)
array([1.1975600e-02, 3.5477642e-06, 1.7415744e-03, ..., 6.2727188e-03,
       1.3800581e-03, 1.9773073e-02], dtype=float32)
>>> vector = laser.embed_sentences("Leaves of Grass", lang="en").reshape(1024)
>>> vector.shape
(1024,)
>>>
```

Рисунок 4.3 – Приклад роботи LASER Embeddings

Наступним кроком буде отримання ключових слів з тексту. На рисунку 4.4 можна побачити приклад отримання ключових слів з тексту. Даний процес є набагато швидшим та дозволяє ефективно витягувати ключову інформацію з тексту.

```
>>> doc
There is a door at the end of a silent corridor. And it's haunting H
arry Pottter's dreams. Why else would he be waking in the middle of
the night, screaming in terror?Harry has a lot on his mind for this,
his fifth year at Hogwarts: a Defense Against the Dark Arts teacher
with a personality like poisoned honey; a big surprise on the Gryff
indor Quidditch team; and the looming terror of the Ordinary Wizardi
ng Level exams. But all these things pale next to the growing threat
of He-Who-Must-Not-Be-Named - a threat that neither the magical gov
ernment nor the authorities at Hogwarts can stop.As the grasp of dar
kness tightens, Harry must discover the true depth and strength of h
is friends, the importance of boundless loyalty, and the shocking pr
ice of unbearable sacrifice.His fate depends on them all.
>>> get_entities(doc)
Harry Pottter PERSON
the middle of the night DATE
his fifth year DATE
the Dark Arts FAC
Gryffindor Quidditch ORG
Harry PERSON
>>>
```

Рисунок 4.4 – Приклад роботи алгоритму отримання ключових слів з тексту.

Після проведення векторизації текстів та отримання ключових слів, наступним кроком є реалізація алгоритму пошуку за допомогою векторних репрезентацій та ключових слів. Для цього буде використано клас `SearchController`, який міститиме логіку порівняння векторів та фільтрації результатів за допомогою ключових слів.

Основні функції класу `SearchController` включатимуть:

1. Отримання векторної репрезентації запиту користувача за допомогою алгоритму `LASER Embeddings`.
2. Отримання ключових слів із запиту користувача з використанням бібліотеки `sprasy`.
3. Порівняння векторної репрезентації запиту з векторними репрезентаціями матеріалів у базі даних за допомогою алгоритму косинусної подібності (`Cosine Similarity`).

4. Фільтрація результатів пошуку за наявністю ключових слів із запиту в назвах, описах та ключових словах матеріалів.
5. Обчислення загального коефіцієнта релевантності для кожного результату шляхом зваженого підрахунку косинусної подібності векторів та відповідності ключових слів.
6. Сортування результатів за коефіцієнтом релевантності та повернення відсортованого списку результатів.

Процес починається з отримання векторної репрезентації запиту користувача за допомогою алгоритму LASER Embeddings. Паралельно з цим, з тексту запиту витягуються ключові слова з використанням бібліотеки spaCy. Ці два етапи є основними для подальшої обробки запиту.

Далі починається цикл обробки кожного матеріалу в базі даних. Для поточного матеріалу проводиться порівняння його векторної репрезентації з векторною репрезентацією запиту за допомогою алгоритму обчислення косинусної подібності. Це дозволяє виміряти семантичну близькість між запитом та матеріалом.

Наступним кроком є порівняння ключових слів запиту з ключовими словами матеріалу. Ця операція виконується за допомогою функції `check_keyword_match`, яка перевіряє наявність ключових слів із запиту в назвах, описах та ключових словах матеріалу.

Після отримання результатів порівняння векторів та ключових слів, обчислюється загальний коефіцієнт релевантності матеріалу відносно запиту. Це робиться шляхом комбінування косинусної подібності векторів та показника відповідності ключових слів за допомогою функції `compute_relevance_score`.

Поточний матеріал та його коефіцієнт релевантності додаються до списку результатів за допомогою операції `results.append((material, relevance_score))`. Цикл продовжується для всіх матеріалів у базі даних.

Після завершення циклу, список результатів сортується в порядку спадання значень коефіцієнта релевантності за допомогою функції `sorted` з

використанням ключа сортування `lambda x: x[4]`, де `x[4]` - це значення релевантності для кожного елемента списку.

Нарешті, відсортований список результатів повертається у вигляді списку матеріалів без коефіцієнтів релевантності за допомогою генератора списків `[result[0] for result in sorted_results]`.

На наведеній діаграмі взаємодії відповідного інтелектуального модуля з користувачем (рисунок 4.5) візуалізовано основні етапи алгоритму пошуку матеріалів в базі даних з використанням векторних репрезентацій та ключових слів, що було описано вище.

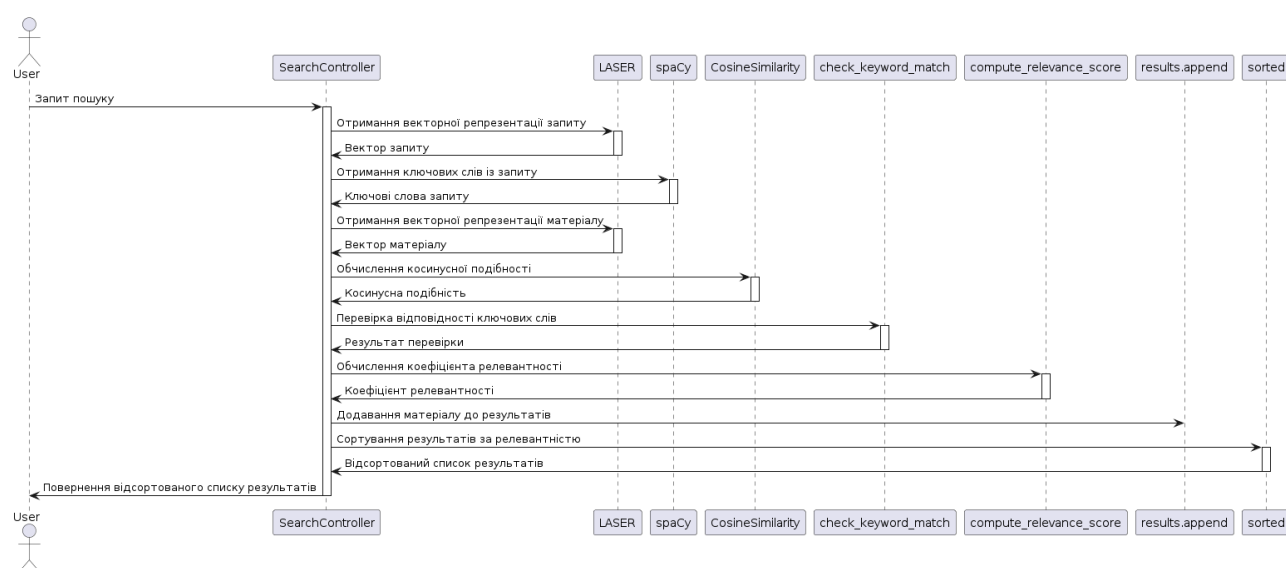


Рисунок 4.5 – UML-діаграма взаємодії інтелектуального модуля пошуку різноформатних матеріалів в базах даних з користувачем

Таким чином, ініціалізація роботи модулів пошуку охоплює весь процес від векторизації текстів до реалізації алгоритму пошуку, забезпечуючи ефективний пошук різноформатних матеріалів у базі даних з урахуванням як семантичної подібності векторних репрезентацій, так і відповідності ключових слів.

В даному розділі було розроблено складові інтелектуального модуля пошуку різноформатних матеріалів в базах даних. Проведено аналіз мов та середовищ програмування для розробки інтелектуального модуля пошуку

різноформатних матеріалів в базах даних. В результаті аналізу вибрано мову програмування Python та середовище програмування PyCharm, як оптимальні. Розроблено засоби обробки та пошуку даних, базу даних, інтерфейс web-сайту та ініціалізовано роботу модулів пошуку.

5 АНАЛІЗ ФУНКЦІОНУВАННЯ ІНТЕЛЕКТУАЛЬНОГО МОДУЛЯ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ

Для проведення аналізу функціонування інтелектуального модуля пошуку різноформатних матеріалів у базах даних було обрано два датасети: LibriSpeech ASR corpus та Best Books Ever Dataset. Вибір даних датасетів дозволить ґрунтовно оцінити роботу модуля з різними типами даних, а саме аудіо та текстовими матеріалами.

1. LibriSpeech ASR corpus – це датасет, що містить аудіозаписи прочитаних книг. Його буде використано для отримання текстової репрезентації аудіо за допомогою класу FormatPreprocessing. Отриманий текст далі буде піддано препроцесингу та векторизації для подальшого використання в пошуку матеріалів. Таким чином, цей датасет дозволить перевірити ефективність модуля в обробці аудіоданих та їх інтеграції в процес пошуку.
2. Best Books Ever Dataset – це датасет, що містить більше ніж 50 тисяч книг у форматі CSV. Представлення даних у табличному вигляді полегшує зчитування інформації про книги та їх подальший препроцесинг за допомогою класу TextPreprocessor. Цей датасет забезпечить перевірку роботи модуля з великими обсягами текстових даних, а також дозволить оцінити ефективність векторизації та пошуку текстових матеріалів.

Для порівняння точності розробленого інтелектуального модуля пошуку різноформатних матеріалів у базах даних було обрано такі пошукові системи: Google, WorldCat та ChatGPT. Ці системи представляють різні підходи до пошуку та обробки інформації, що дозволить комплексно оцінити переваги та недоліки розробленого модуля.

1. Google є однією з найпопулярніших та потужних пошукових систем в Інтернеті. Порівняння з Google дозволить оцінити ефективність модуля в пошуку різноформатних матеріалів відносно загальнодоступних інструментів пошуку в мережі.

2. WorldCat – це всесвітній каталог бібліотечних ресурсів, що охоплює величезну кількість книг, аудіокниг, відео та інших матеріалів. Порівняння з WorldCat дасть змогу оцінити якість пошуку модуля в контексті спеціалізованих бібліотечних каталогів та систем.
3. ChatGPT – це сучасна система штучного інтелекту, здатна розуміти та генерувати природну мову. Порівняння з ChatGPT дозволить оцінити ефективність модуля відносно передових технологій обробки природної мови та інформаційного пошуку на основі великих мовних моделей.

Вибір різноманітних систем для порівняння забезпечить всебічну оцінку переваг та недоліків розробленого модуля, виявлення його сильних та слабких сторін, а також можливостей для подальшого вдосконалення. Такий комплексний підхід до тестування дозволить ретельно перевірити функціональність модуля та його здатність ефективно здійснювати пошук різноформатних матеріалів у базах даних.

Розпочнемо тестування на пошукові твору “В морі є крокодили” та аудіо книги “451 градус за Фаренгейтом”. Результати тестування зображено на рисунку 5.1-5.3.

ІНТЕЛЕКТУАЛЬНИЙ МОДУЛЬ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ
Підготував студент 1КН-20Б
Коханівський Антон Павлович

Введіть запит для пошуку...

Пошук

Налаштування

Скинути

Результати пошуку - "When a village succumbs to Taliban rule, a boy is taken to Pakistan by his mother. Sparking a five-year journey through multiple countries before seeking asylum in Italy."

№1 Текст: In the Sea There are Crocodiles: Based on the True Story of Enaiatollah Akbari	Опис: When a ten-year-old boys village in Afghanistan falls prey to Taliban rule, his mother shepherds the boy across the border into Pakistan but has to leave him there all alone to fend for himself. Thus begins Enaiats remarkable and often punishing five...	Відповідність: 54.05%
№2 Текст: Notes from My	Опис: From the ever-intriguing and appealing actress Angelina Jolie comes the	

Рисунок 5.1 – Результат пошуку твору “В морі є крокодили” в інтелектуальному модулі

Результати пошуку - "When a village succumbs to Taliban rule, a boy is taken to Pakistan by his mother. Sparking a five-year journey through multiple countries before seeking asylum in Italy."

№1 Текст: In the Sea There are Crocodiles: Based on the True Story of Enaiatollah Akbari	Опис: When a ten-year-old boy's village in Afghanistan falls prey to Taliban rule, his mother shepherds the boy across the border into Pakistan but has to leave him there all alone to fend for himself. Thus begins Enaiat's remarkable and often punishing five...	Відповідність:	54.05%
№2 Текст: Notes from My Travels: Visits with Refugees in Africa, Cambodia, Pakistan and Ecuador	Опис: From the ever-intriguing and appealing actress Angelina Jolie comes the personal journals she compiled while performing humanitarian relief efforts in such countries as Sierra Leone and Tanzania, Pakistan and Cambodia. Three years ago, award-winning a...	Відповідність:	41.45%
№3 Текст: For the Love of a Son: One Afghan Woman's Quest for Her Stolen Child	Опис: From the time she was a little girl, Maryam rebelled against the terrible second-class existence that was her destiny as an Afghan woman. Maryam had witnessed the miserable fate of her grandmother and three aunts, and wished that she had been born a b...	Відповідність:	39.92%
№4 Текст: A Map Is Only One Story: Twenty Writers on Immigration, Family, and the Meaning of Home	Опис: From rediscovering an ancestral village in China to experiencing the realities of American life as a Nigerian, the search for belonging crosses borders and generations. Selected from the archives of Catapult magazine, the essays in A Map Is Only One ...	Відповідність:	36.01%
№5 Текст: The Seven	Опис: One of the most dramatic stories of genetic discovery since James Watson's The		

Рисунок 5.2 – Результати пошуку твору “В морі є крокодили” в інтелектуальному модулі

Результати пошуку - "In a society dominated by television and where literature is fading, certain individuals perform unconventional duties. Among them is a man tasked with a peculiar responsibility—to eradicate forbidden objects, without question. However, an encounter with an intriguing acquaintance sparks a subtle shift in his perception, leading him to reconsider the truths he once took for granted. As unexpected events unfold, he finds himself questioning the very essence of his existence."

№1 Аудіо: Fahrenheit 451	Опис: Guy Montag is a fireman. In his world, where television rules and literature is on the brink of extinction, firemen start fires rather than put them out. His job is to destroy the most illegal of commodities, the printed book, along with the houses i...	Відповідність:	45.45%
№2 Текст: On The Intuitive Spectrum: A Deeper Look into the Amazing Value of Intuition	Опис: The sun god Apollo sentenced Cassandra to be able to prophesy her intuitions but to never be believed. Since ancient times, the gift of prophecy and intuition was a present that was bestowed upon Cassandra by Apollo. Apollo loved Cassandra, and she r...	Відповідність:	32.98%
№3 Текст: When the Heart Waits: Spiritual Direction for Life's Sacred Questions	Опис: I stood at the window watching the cocoon, which hung in the winter air like an upside-down question mark. That was the moment... I understood. Really understood. Crisis, change, all the myriad upheavals that blister the spirit and leave us groping— ...	Відповідність:	32.89%
№4 Аудіо: Chicken Soup for the Soul: Miracles Happen: 101 Inspirational Stories about Hope, Answered...	Опис: Miracles happen every day—we just have to look to see them. These powerful stories will give you hope and deepen your faith. These 101 amazing stories prove that miracles happen, every day, to people from all walks of life. You will find hope and faith...	Відповідність:	32.81%

Рисунок 5.3 – Результати пошуку аудіо книги “451 градус за Фаренгейтом” в інтелектуальному модулі

Як видно з рисунків 5.1–5.3, інтелектуальний модуль успішно виконує поставлену задачу та надає точні результати пошуку. Проведемо аналіз результатів пошуку за допомогою систем WorldCat, ChatGPT та Google.

- WorldCat – при введенні заданого пошукового запиту не було знайдено жодної літератури та аудіо книги.
- ChatGPT – успішно виконав пошукову задачу.
- Google – серед перших 30 результатів, наданих Google, не було знайдено очікуваного твору та аудіо книги.

Отримані результати дозволяють зробити висновок, що розроблений інтелектуальний модуль успішно виконує поставлене завдання пошуку різноформатних матеріалів у базах даних. Завдяки застосуванню методів глибокого аналізу змісту запиту та матеріалів, модуль здатний надавати точні та релевантні результати пошуку, що відповідають запиту.

Крім того, даний інтелектуальний модуль надає можливість більш гнучкого налаштування, що дозволяє персоналізувати критерії пошуку та сфокусуватись саме на тих аспектах, які становлять інтерес для користувача. Наприклад, у випадку, коли користувач надає лише семантичний зміст опису матеріалу, не обов'язково приділяти значну увагу семантичному змісту назви матеріалу або ключовим словам, що містяться у заголовку шуканої інформації, оскільки це може призвести до отримання небажаних результатів. Меню, що використовується для налаштування, представлено на рисунку 5.4.



Рисунок 5.4 – Меню налаштування ваг інтелектуального модуля пошуку різноформатних матеріалів у базах даних

За замовчуванням ваги, встановлені в модулі, мають такі значення:

1. Ваги семантики назви – 1;
2. Ваги семантики опису – 1;
3. Ваги ключових слів назви – 1;
4. Ваги ключових слів опису – 1;

При зміні ваг змінюється й кінцевий результат, оскільки в такому випадку більша увага надається певним вагам, що мають вищу значущість для пошуку.

Для тестування змінимо ваги на такі значення:

1. Ваги семантики назви – 1;
2. Ваги семантики опису – 17;
3. Ваги ключових слів назви – 1;
4. Ваги ключових слів опису – 1;

Це дозволить підвищити максимально можливу схожість на стовідсоткову відповідність семантики опису з 25% до 85%. Результати тестування зображено на рисунку 5.5.

Результати пошуку - "In a society dominated by television and where literature is fading, certain individuals perform unconventional duties. Among them is a man tasked with a peculiar responsibility—to eradicate forbidden objects, without question. However, an encounter with an intriguing acquaintance sparks a subtle shift in his perception, leading him to reconsider the truths he once took for granted. As unexpected events unfold, he finds himself questioning the very essence of his existence."		
№1 Текст: The Unlikely Ones	Опис: A band of outcasts begins an arduous journey through a world of evil witches, walking trees, and miraculous gems along a path that will reunite them with their true destinies....	Відповідність: 63.78%
№2 Текст: Pictures of Hollis Woods	Опис: In this Newbery Honor Book, a troublesome 12-year-old orphan, staying with an elderly artist who needs her, remembers the only other time she was happy in a foster home, with a family that truly seemed to care about her....	Відповідність: 63.53%
№3 Текст: Tegami Bachi: Letter Bee, Volume 1	Опис: In Amberground, a dangerous terrain where a man-made star casts a permanent twilight, young Lag Seeing aspires to become a Letter Bee: a postman entrusted to deliver the hearts of people separated from the ones they love....	Відповідність: 62.98%
№4 Текст: Un mago de Terramar / Las Tumbas de Atuan	Опис: En estas dos primeras novelas de la saga, las aventuras vividas por el joven mago Ged y la sacerdotisa Tenar los transformarán profundamente, les harán crecer y liberarse de sus miedos, y se convertirán en auténticos héroes que ayudarán a restablecer...	Відповідність: 62.31%

Рисунок 5.5 – Результати пошуку аудіо книги “451 градус за Фаренгейтом” в інтелектуальному модулі зі зміненими початковими вагами

Як видно з рисунка 5.5, результати суттєво змінилися, і ми отримали зовсім інші дані порівняно з початковим результатом, хоча було змінено лише одне значення в налаштуваннях. Це свідчить про високий потенціал для вдосконалення в напрямку автоматизації визначення оптимальних ваг, підвищення швидкодії, оптимізації використання пам'яті тощо.

Перейдемо до тестування інтелектуального модуля та порівняння його з існуючими системами пошуку. Для отримання детальної статистики було проведено експеримент, в ході якого було здійснено 500 запитів для пошуку різноформатних матеріалів. Результати тестування інтелектуального модуля пошуку різноформатних матеріалів в базах даних та порівняння його точності з іншими системами пошуку, як WorldCat, ChatGPT та Google представлено в таблиці 5.1.

Таблиця 5.1 – Порівняння результатів пошуку систем пошуку WorldCat, ChatGPT, Google та інтелектуального модуля пошуку різноформатних матеріалів в базах даних.

Пошуковий засіб	Позитивний результат пошуку текстових матеріалів	Позитивний результат пошуку аудіо матеріалів	Швидкодія (середній час отримання результатів)
WorldCat	9.6%	3.2%	4,5 секунди
ChatGPT	90.8%	14.8%	20 секунд
Google	78.2%	30.2%	2 секунди
Інтелектуальний модуль	96.4%	56.8%	300 секунд

Беручи до уваги результати, що зображені в таблиці 5.1, можна дійти до висновку, що розроблений інтелектуальний модуль виконав своє завдання підвищення точності пошукових систем та отримав високі результати при тестуванні програмного забезпечення порівняно з іншими системами пошуку.

Отже, згідно з наведеною інформацією, розроблений інтелектуальний модуль пошуку різноформатних матеріалів у базах даних продемонстрував кращу точність порівняно з такими системами, як WorldCat, ChatGPT та Google, особливо в пошуку аудіоматеріалів. Хоча час обробки запитів був дещо довшим, ніж у конкурентів, модуль забезпечив найвищий відсоток позитивних результатів пошуку як для текстових, так і для аудіоматеріалів.

Ключові результати тестування:

1. Інтелектуальний модуль показав найвищу точність пошуку текстових матеріалів (96,4%) та аудіо матеріалів (56,8%).
2. Інтелектуальний модуль потребував більше часу на обробку запитів (300 секунд у середньому), ніж інші системи.

Загалом, попри нижчу швидкодію, інтелектуальний модуль пошуку продемонстрував перевагу у точності та здатності ефективно шукати різноформатні матеріали, особливо аудіодані, що є його ключовою перевагою. Також, базуючись на результатах експерименту, можна стверджувати, що використання даного модуля може підвищити точність процесу пошуку текстових файлів на 5.6% для текстових файлів та на 26.6% для аудіо файлів.

В даному розділі було проаналізовано функціонування інтелектуального модуля пошуку різноформатних матеріалів пошуку в базах даних. Протестовано роботоздатність розробленого інтелектуального модуля та порівняно точність пошуку з існуючими засобами пошуку даних. Результати тестування продемонстровані інтелектуальним модулем підвищення точності процесу пошуку текстових файлів на 5.6% для текстових файлів та на 26.6% для аудіо файлів.

ВИСНОВКИ

Поставлені перед бакалаврською дипломною роботою задачі виконано в повному обсязі, а саме:

- Проаналізовано сучасного стану пошуку різноформатних матеріалів в базах даних;
- Здійснено постановку конкретних завдань для реалізації інтелектуального модуля пошуку різноформатних матеріалів в базах даних;
- Розроблено удосконалений алгоритм пошуку різноформатних матеріалів в базах даних;
- Розроблено структуру модуля пошуку різноформатних матеріалів в базах даних;
- Розроблено складові інтелектуального модуля пошуку різноформатних матеріалів в базах даних;
- Проаналізовано функціонування інтелектуального модуля пошуку різноформатних матеріалів в базах даних;

У ході виконання бакалаврської роботи були проаналізовані сучасні засоби пошуку різноформатних матеріалів у базах даних, було вказано їх переваги, недоліки та обрано й обґрунтовано алгоритми для розробки інтелектуального модуля.

Було розроблено удосконалений алгоритм, що виконує дві основні функції, а саме обробку даних та пошук даних. Обробка даних виконується за рахунок приведення форматів до текстового вигляду, отримання ключових слів з тексту та семантичного аналізу текстової інформації з метою отримання його векторної репрезентації. Пошук відбувається за рахунок порівняння векторних репрезентацій запиту та файлів у базі даних за допомогою обчислення косинусної відстані та ранжування отриманих результатів для подальшого відображення на користувацькому інтерфейсі, що представлений у вигляді вебсайту.

Було розроблено структуру модуля пошуку різноформатних матеріалів в базах даних, що включає розробку структури серверної частини та структуру вебінтерфейсу користувача. В межах розробки структури серверної частини було проаналізовано основні завдання складових серверної частини модуля та взаємодію між програмними блоками відповідного інтелектуального модуля. В межах розробки структури вебінтерфейсу користувача було розроблено його дизайн та проаналізовано компоненти дизайну з визначенням їхнього функціонала та застосування.

Відповідно до завдання бакалаврської роботи було розроблено удосконалений алгоритм пошуку різноформатних матеріалів у базах даних та користувацький інтерфейс у вигляді вебсайту. Також за допомогою ручного тестування було проведено тестування інтелектуального модуля. В межах тестування було виконано 500 запитів та здійснено порівняння результатів пошуку між сучасними системами пошуку WorldCat, ChatGPT, Google та інтелектуальним модулем пошуку різноформатних матеріалів в базах даних. Результати тестування показали підвищення точності процесу пошуку текстових файлів на 5.6% для текстових файлів та на 26.6% для аудіо файлів.

Мета роботи – підвищення точності процесу пошуку різноформатних матеріалів у базах даних була досягнута за рахунок використання семантичного аналізу матеріалів та запиту користувача й приведення різних форматів до одного – векторного, що дозволяє здійснювати пошук різних форматів, використовуючи алгоритми пошуку лише для одного формату, що в свою чергу забезпечує приріст точності на 5.6% для текстових файлів та на 26.6% для аудіо файлів в порівнянні з сучасними системами пошуку.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Савчук Т. О., Коханівський А. П. Структура інтелектуального модуля пошуку різноформатних даних в базах даних – Тези міжнародної науково-практичної інтернет-конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2024)».
2. Савчук Т. О., Коханівський А. П. Обґрунтування вибору алгоритму пошуку різноформатних матеріалів в базах даних. X Міжнародна науково-практична конференція «TOPICAL ASPECTS OF MODERN SCIENTIFIC RESEARCH», Токіо, Японія, 2024.
3. Савчук Т. О., Коханівський А. П. Удосконалений алгоритм пошуку різноформатних матеріалів в базах даних. VII Міжнародна науково-практична конференція «SCIENCE AND SOCIETY: MODERN TRENDS IN A CHANGING WORLD», Відень, Австрія, 2024.
4. Introduction to Information Retrieval. Cambridge University Press, 2008.
5. Croft B., Metzler D., Strohman T. Search Engines: Information Retrieval in Practice. Pearson Education, Limited, 2011. 552 p.
6. Klein E., Bird S., Loper E. Natural Language Processing with Python. O'Reilly Media, Incorporated, 2009.
7. Berryman J., Turnbull D. Relevant Search: With applications for Solr and Elasticsearch. Manning Publications, 2016. 360 p.
8. Chollet F. Deep Learning with Python. Manning Publications, 2017. 384 p.
9. Sarkar D. Text Analytics with Python. Berkeley, CA : Apress, 2016. URL: <https://doi.org/10.1007/978-1-4842-2388-8> (date of access: 22.05.2024).
10. Lutz M. Learning Python. O'Reilly Media, Incorporated, 2013.
11. Ramalho L. Fluent Python: Clear, Concise, and Effective Programming / ed. by M. Blanchette, R. Roumeliotis. O'Reilly Media, 2015. 792 p.
12. Phillips D. Python 3 Object Oriented Programming. Packt Publishing, Limited, 2010.

- 13.Géron A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media, 2019. 856 p.
- 14.Navathe S., Elmasri R. Fundamentals of Database Systems. Pearson Education, Limited, 2013. 1088 p.
- 15.Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
- 16.Schwarz, M., Artetxe, M., Wenzek, G., Grave, É., & Bojanowski, P. (2019). Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. arXiv preprint arXiv:1812.10464.
- 17.Duckett J. HTML & CSS: Design and build websites. 2014. 490 p.
- 18.Haverbeke M. Eloquent JavaScript: A Modern Introduction to Programming. 3rd ed. San Francisco, California, United States of America : No Starch Press, 2018. 472 p.
- 19.Meyer E. A., Weyl E. CSS : The Definitive Guide: Web Layout and Presentation. O'Reilly Media, Incorporated, 2023.
- 20.McFarland D. S. CSS : the Missing Manual: The Missing Manual. O'Reilly Media, Incorporated, 2006.

ДОДАТКИ

ДОДАТОК А

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Інтелектуальний модуль пошуку різноформатних матеріалів
в базах даних

Тип роботи: бакалаврська дипломна робота
(БДР, МКР)

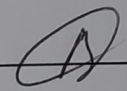
Підрозділ кафедра комп'ютерних наук, ФІТА
(кафедра, факультет)

Показники звіту подібності Unichesk

Оригінальність 96,82% Схожість 3,18%

Аналіз звіту подібності (відмітити потрібне):

- ☒ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- ☐ Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
- ☐ Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку  Озеранський В.С.

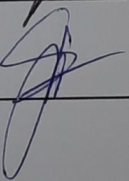
Ознайомлені з повним звітом подібності, який був згенерований системою Unichesk щодо роботи.

Автор роботи



Коханівський А.П.

Керівник роботи



Савчук Т.О.

ДОДАТОК В. ІНСТРУКЦІЯ КОРИСТУВАЧА

Для запуску даного інтелектуального модуля потрібно здійснити такі дії:

1. Завантажити та встановити Python;
2. Завантажити інтелектуальний модуль та перейти в директорію модуля в консолі;
3. Встановити всі залежності за допомогою команди `python requirements.txt`;
4. Після завантаження всіх бібліотек, прописати команду `python read_data.py path_to_file_with_data_to_search/`, де `path_to_file_with_data_to_search` – файл з набором даних, по якому буде відбуватись пошук.
5. Після завершення обробки всіх матеріалів та запису їх в базу даних, прописати команду `python run.py`;
6. Перейти по посиланню: <http://127.0.0.1:8080/>;
7. Розпочати роботу.

ДОДАТОК Г. ЛІСТИНГ ПРОГРАМИ

format_preprocessing.py

```
import speech_recognition as sr

class FormatPreprocessing:
    def audio_to_text(self, audio_file):
        recognizer = sr.Recognizer()
        with sr.AudioFile(audio_file) as source:
            audio = recognizer.record(source)
            text = recognizer.recognize_google(audio)
        return text
```

database_connection.py

```
import psycopg2

class DatabaseConnection:
    def __init__(self, dbname, user, password, host, port):
        self.conn = psycopg2.connect(database=dbname, user=user, password=password, host=host,
port=port)

    def get_connection(self):
        return self.conn

    def close_connection(self):
        self.conn.close()
```

database_query.py

```
class DatabaseQuery:
    def __init__(self, connection):
        self.conn = connection
        self.cursor = self.conn.cursor()

    def execute_query(self, query, params=None):
        self.cursor.execute(query, params)
        return self.cursor.fetchall()
```

```
def commit(self):
    self.conn.commit()
```

```
def close(self):
    self.cursor.close()
```

text_preprocessor.py

```
import spacy
```

```
import string
```

```
class TextPreprocessor:
```

```
    def __init__(self):
        self.nlp = spacy.load('en_core_web_sm')
```

```
    def clean_text(self, text):
        text = text.lower()
        text = text.translate(str.maketrans("", "", string.punctuation))
        return text
```

```
    def get_keywords(self, text):
        doc = self.nlp(text)
        return [entity.text.lower() for entity in doc.ents]
```

search_controller.py

```
from sklearn.metrics.pairwise import cosine_similarity
```

```
import numpy as np
```

```
class SearchController:
```

```
    def __init__(self, database_query, text_vectorizer, text_preprocessor):
        self.db_query = database_query
        self.vectorizer = text_vectorizer
        self.preprocessor = text_preprocessor
```

```
    def compute_similarity(self, query_vectors, embedding):
        embedding = np.array(embedding).reshape(1, -1)
        return max([cosine_similarity(query_vector, embedding) for query_vector in query_vectors])
```

```

def compute_keyword_similarity(self, query_keywords, current_keywords):
    similarity = 0
    for query_keyword in query_keywords:
        if query_keyword in current_keywords:
            similarity += 1
    if len(query_keywords) > 0:
        return similarity / len(query_keywords)
    else:
        return 0

def search(self, query, weights=(1, 1, 1, 1), batch_size=1000):
    query_vectors = self.vectorizer.text_to_vectors(query)
    query_keywords = self.preprocessor.get_keywords(query)

    search_query = """
SELECT
    TM.materialid,
    ARRAY_AGG(BDE.descriptionembedding) AS descriptionembeddings,
    TM.title_embeddings AS title_embeddings,
    TM.title_keywords AS title_keywords,
    TM.description_keywords AS description_keywords
FROM
    TextMaterials TM
JOIN
    BookDescriptionEmbedding BDE ON TM.materialid = BDE.bookid
JOIN
    Books B ON TM.materialid = B.bookid
WHERE
    B.rating > 3.95 AND B.rating < 4.05
GROUP BY
    TM.materialid, TM.title_embeddings, TM.title_keywords, TM.description_keywords
ORDER BY
    TM.materialid;
"""

```

```

data = self.db_query.execute_query(search_query)

materialids = []
similarity_scores = np.zeros((len(data), 4))

index = 0
for row in data:
    materialid, bookDescriptionEmbedding, title_embedding, title_keywords,
description_keywords = row

    title_keywords = self.preprocessor.clean_text(title_keywords)
    description_keywords = self.preprocessor.clean_text(description_keywords)
    materialids.append(materialid)

    similarity_scores[index][0] = self.compute_similarity(query_vectors, title_embedding)[0][0]

    description_similarities = [
        self.compute_similarity(query_vectors, sentence)
        for sentence in bookDescriptionEmbedding
    ]
    similarity_scores[index][1] = np.mean(np.sort(description_similarities)[:len(query_vectors)-1])

    similarity_scores[index][2] = self.compute_keyword_similarity(query_keywords,
title_keywords)
    similarity_scores[index][3] = self.compute_keyword_similarity(query_keywords,
description_keywords)
    index += 1

df = pd.DataFrame(columns=['materialid', 'similarity_score'])
df['materialid'] = materialids
df['similarity_score'] = similarity_scores[:, 0] * weights[0]
df['similarity_score'] += similarity_scores[:, 1] * weights[1]
df['similarity_score'] += similarity_scores[:, 2] * weights[2]
df['similarity_score'] += similarity_scores[:, 3] * weights[3]

```

```
df['similarity_score'] /= sum(weights)
```

```
df.sort_values(by='similarity_score', ascending=False, inplace=True)
```

```
return df
```

run.py

```
from flask_app import app
```

```
if __name__ == '__main__':
```

```
    app.run(host='0.0.0.0', port=8080, debug=True)
```

page.py

```
from flask import Blueprint, render_template, request
```

```
from flask_app import search_controller
```

```
page = Blueprint('page', __name__)
```

```
@page.route('/', methods=["GET"])
```

```
def home():
```

```
    if request.method == "GET":
```

```
        if 'search' in request.args:
```

```
            query = request.args.get("query")
```

```
            weights = [request.args.get("weights-sem-title"),
```

```
                        request.args.get("weights-sem-desc"),
```

```
                        request.args.get("weights-key-title"),
```

```
                        request.args.get("weights-key-desc")]
```

```
            weights = [int(i) for i in weights]
```

```
            if len(query) > 0:
```

```
                df = search_controller.search(query, weights)
```

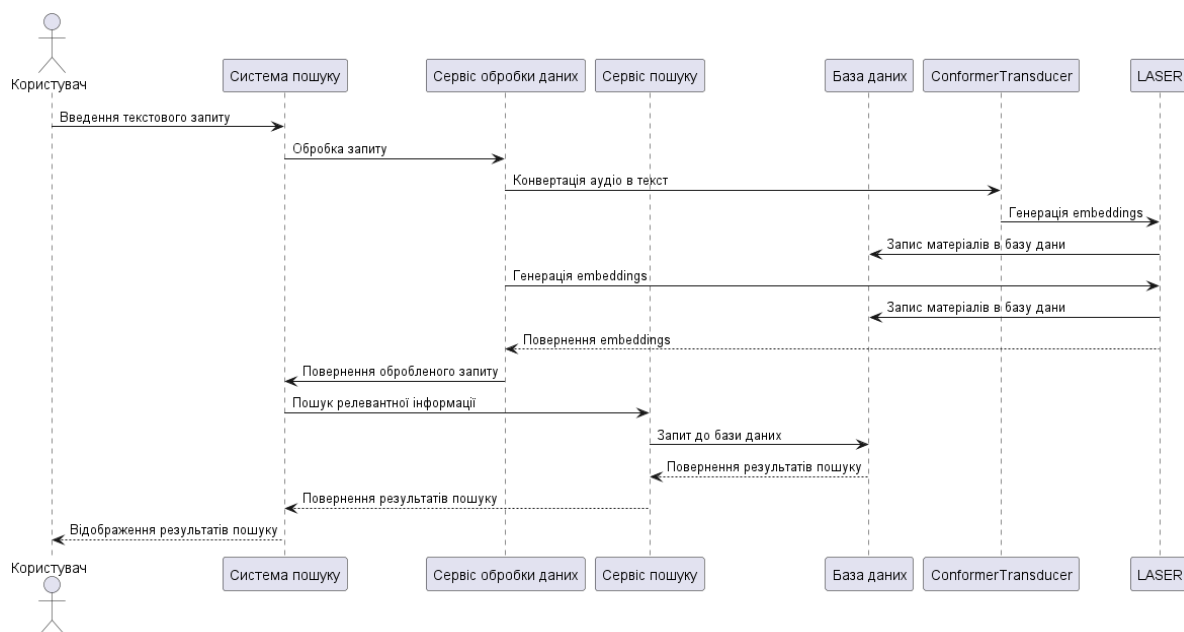
```
                search_results = df.head(50).to_dict(orient='records')
```

```
                return render_template("base.html", search_results=search_results, query=query,
```

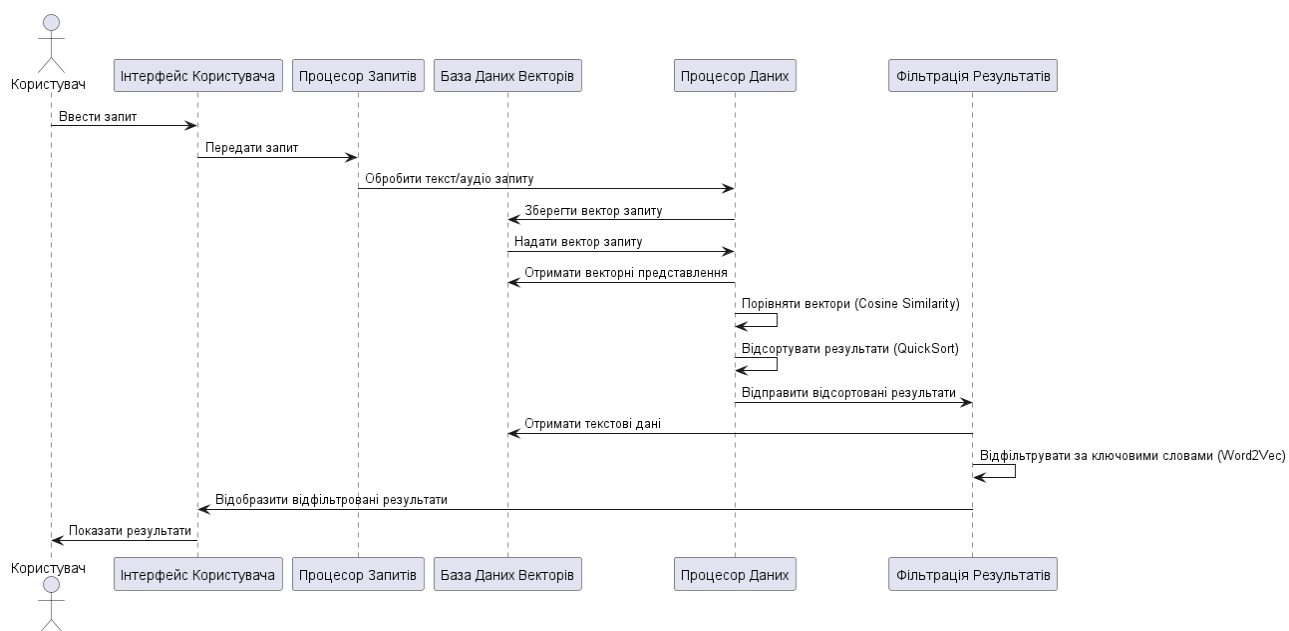
```
none=None)
```

```
            return render_template("base.html", search_results=None, query=None, none=None)
```

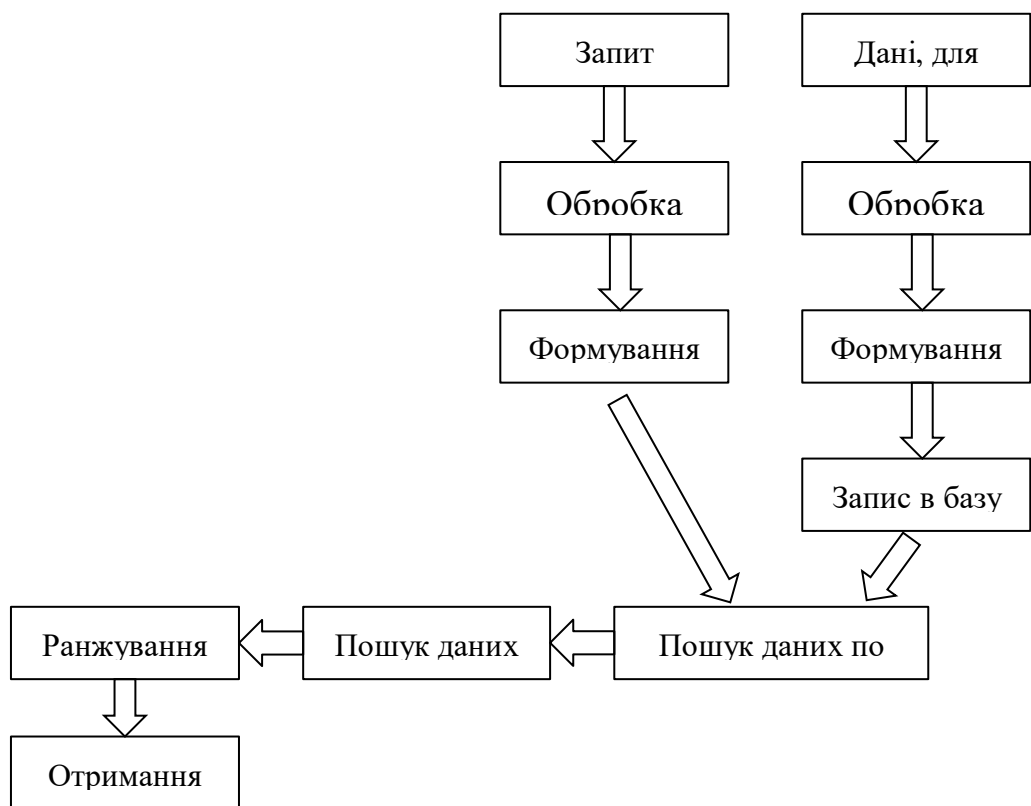
ДОДАТОК Д. ГРАФІЧНА ЧАСТИНА



UML-діаграма взаємодії алгоритмів обробки даних з користувачем



UML-діаграма взаємодії алгоритмів пошуку даних з користувачем



Взаємодія між блоками інтелектуального модуля пошуку різноформатних матеріалів в базах даних.

ІНТЕЛЕКТУАЛЬНИЙ МОДУЛЬ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ

Підготував студент 1КН-20Б
Коханівський Антон Павлович

Введіть запит для пошуку...

Пошук

Налаштування

Скинути

Результати пошуку

#1 Аудіо: Test 1

Опис: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo...

Відповідність: 96.32%

#2 Аудіо: Test 2

Опис: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo...

Відповідність: 94.02%

#3 Текст: Test 3

Опис: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo...

Відповідність: 91.02%

Дизайн вебсайту (користувачького інтерфейсу)

	materialid text	title text	description text	title_keywords text[]	description_keywords text[]	title_embeddings double precision[]
1	36607149-explosive-growth	Explosive Growth: A Few Things I Lear...	*Ranked BEST STARTUP BOOKS O...	[null]	[null]	{0.0049823015,-0.0003385113,0.0067029105,0.034129154,0.00808728
2	310612_A_Confederacy_of_Dunces	A Confederacy of Dunces	Alternate cover for this ISBN can b...	[null]	[null]	{0.012819975,9.492003e-05,0.00017836834,0.007984464,0.046658777
3	17848055-kurathi-mudukku	Kurathi Mudukku		[null]	[null]	{0.006212897,5.9492554e-06,0.0016610442,0.010274799,-0.00087411
4	35068432-ill-be-gone-in-the-dark	Ill Be Gone in the Dark: One Womans O...	A masterful true crime account of ...	[null]	[null]	{0.010700526,-8.980784e-05,-0.001455025,0.018671352,0.024090238,C
5	546018.Roots	Roots: The Saga of an American Family	When he was a boy in Henning, Te...	[null]	[null]	{0.005166715,-0.00032035628,0.0013971849,0.0043896334,0.0221046
6	27494.Leaves_of_Grass	Leaves of Grass		[null]	[null]	{0.0119756,3.5477642e-06,0.0017415744,0.00044893447,0.049537923,
7	409589.Conan_	Conan : die Original-Erzählungen aus d...	Complete Conan of Cimmeria	[null]	[null]	{0.013067927,-0.0001770464,0.005879689,0.015210874,0.009303803,C
8	14905.The_Complete_Novels	The Complete Novels	This volume contains the six major...	[null]	[null]	{0.0012840928,-0.0003346247,0.0023801161,0.029088406,0.00323778
9	11553583-prokieta-avilja	Prokieta avilja	Prokieta avilja/The Damned Yard (...)	[null]	[null]	{0.008111262,-0.00034484547,-0.0020586473,0.014443539,0.02267218
10	38236342-it-is-not-what-it-is	It Is Not What It Is: THE REAL (s)PAIN ...	REVISED, UPDATED and ADDED AU...	[null]	[null]	{0.0070932345,-0.00031022646,-0.002955124,0.009799439,0.02660756
11	2203.John_Adams	John Adams	The enthralling, often surprising st...	[null]	[null]	{0.011200694,-0.00063943875,-0.002771761,0.00061900326,7.475604e
12	1923820.Holy_Bible	Holy Bible: King James Version		[null]	[null]	{0.0053794044,0.0009988862,-0.004823747,0.015187092,0.022381894
13	43822122-the-official-downton-abbey-cookbook	The Official Downton Abbey Cookbook	2020 IBPA Awards Winner! The Off...	[null]	[null]	{0.007325685,-0.00033616886,-0.0012285957,0.0059004696,0.0191885
14	811439.Gigolos_Get_Lonely_Too	Gigolos Get Lonely Too	Three popular authors combine cr...	[null]	[null]	{0.0034097387,3.30587e-05,0.0010671808,0.009885587,0.030323107,C
15	36590307-unblinded	Unblinded: One Man's Courageous Jou...		[null]	[null]	{0.0039394526,0.003953158,0.0032228506,0.014928497,0.033913113,}
16	13221354-take-that-to-the-bank	Take That to the Bank		[null]	[null]	{0.004427637,0.0015935577,0.00055367855,0.0061998735,0.03804468
17	1768603.The_White_Tiger	The White Tiger	Introducing a major literary talent, ...	[null]	[null]	{0.0043030214,-0.0031751508,-0.00102385,0.013953023,0.0127902925
18	21348.Aesop_s_Fables	Aesops Fables		[null]	[null]	{0.02342464,0.00015830244,-0.005790694,0.003756255,0.03323583,-8
19	17184.The_Invisible_Man	The Invisible Man	This masterpiece of science fictio...	[null]	[null]	{-0.0005562299,-0.0021373916,-0.00050916575,0.014916419,-0.004555
20	676737.Grendel	Grendel	The first and most terrifying monst...	[null]	[null]	{0.004561987,-0.0006520462,-0.0006111391,0.0016582242,-0.0017101
21	607639.The_Beach	The Beach	After discovering a seemingly Ede...	[null]	[null]	{0.003750065,-0.0015243056,-0.00015172487,-0.002127778,0.0019092
22	1249630.Os_Maias	Os Maias	Biblioteca Ilisseia de Autores Port...	[null]	[null]	{0.002072322,0.0039243717,-0.00019131348,0.005156271,0.00401025
23	15992857-baby	Baby	A gripping psychological thriller on...	[null]	[null]	{-7.620468e-05,-0.00028729808,-0.0005522816,0.002888368,-0.011307
24	7331435-a-visit-from-the-goon-squad	A Visit from the Goon Squad	Jennifer Egan's spellbinding interlo...	[null]	[null]	{0.0053012306,-0.0009244522,-0.0011608655,0.0014370577,0.0274532
25	23529.From_Hell	From Hell	"I shall tell you where we are. Were...	[null]	[null]	{0.0024066756,0.004084306,-0.0021647462,0.0029243873,0.0712518,

Приклад фрагмента реляційної моделі таблиці TextMaterials

```
C:\Users\balal\Desktop\Code\Diploma\venv\Scripts\python.exe C:\Users\balal\Desktop\Code\Diploma\test.py
8%|███████| 4140/52478 [3:09:44<39:50:06, 2.97s/it, 87640.PostSecret]
```

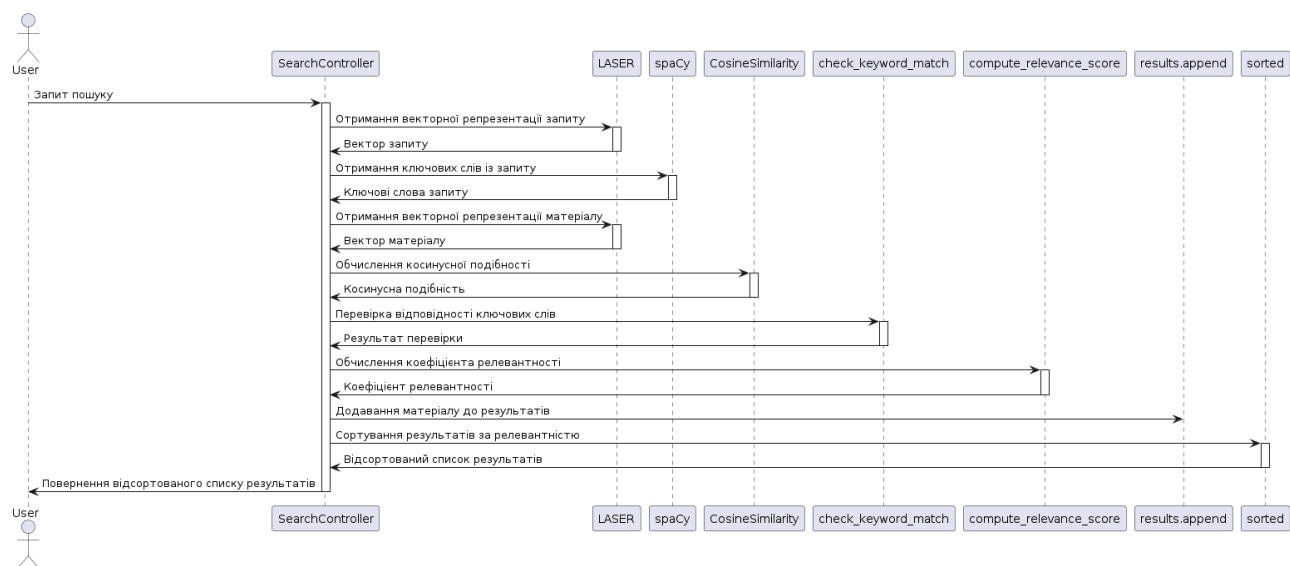
Процес векторизації тексту

```
>>> from laserembeddings.laser import Laser
>>> laser = Laser()
>>> laser.embed_sentences("Leaves of Grass", lang="en").reshape(1024)
array([1.1975600e-02, 3.5477642e-06, 1.7415744e-03, ..., 6.2727188e-03,
       1.3800581e-03, 1.9773073e-02], dtype=float32)
>>> vector = laser.embed_sentences("Leaves of Grass", lang="en").reshape(1024)
>>> vector.shape
(1024,)
>>>
```

Приклад роботи LASER Embeddings

```
>>> doc
There is a door at the end of a silent corridor. And it's haunting H
arry Pottter's dreams. Why else would he be waking in the middle of
the night, screaming in terror?Harry has a lot on his mind for this,
his fifth year at Hogwarts: a Defense Against the Dark Arts teacher
with a personality like poisoned honey; a big surprise on the Gryff
indor Quidditch team; and the looming terror of the Ordinary Wizardi
ng Level exams. But all these things pale next to the growing threat
of He-Who-Must-Not-Be-Named - a threat that neither the magical gov
ernment nor the authorities at Hogwarts can stop.As the grasp of dar
kness tightens, Harry must discover the true depth and strength of h
is friends, the importance of boundless loyalty, and the shocking pr
ice of unbearable sacrifice.His fate depends on them all.
>>> get_entities(doc)
Harry Pottter PERSON
the middle of the night DATE
his fifth year DATE
the Dark Arts FAC
Gryffindor Quidditch ORG
Harry PERSON
>>>
```

Приклад роботи алгоритму отримання ключових слів з тексту.



UML-діаграма взаємодії інтелектуального модуля пошуку різноформатних матеріалів в базах даних з користувачем

ІНТЕЛЕКТУАЛЬНИЙ МОДУЛЬ ПОШУКУ РІЗНОФОРМАТНИХ МАТЕРІАЛІВ В БАЗАХ ДАНИХ
 Підготував студент 1КН-20Б
 Коханівський Антон Павлович

Введіть запит для пошуку...

Пошук

Налаштування

Скинути

Результати пошуку - "When a village succumbs to Taliban rule, a boy is taken to Pakistan by his mother. Sparking a five-year journey through multiple countries before seeking asylum in Italy."

№1 Текст: In the Sea There are Crocodiles: Based on the True Story of Enaiatollah Akbari	Опис: When a ten-year-old boys village in Afghanistan falls prey to Taliban rule, his mother shepherds the boy across the border into Pakistan but has to leave him there all alone to fend for himself. Thus begins Enaiats remarkable and often punishing five...	Відповідність: 54.05%
№2 Текст: Notes from My	Опис: From the ever-intriguing and appealing actress Angelina Jolie comes the	

Результат пошуку твору “В морі є крокодили” в інтелектуальному модулі

Результати пошуку - "When a village succumbs to Taliban rule, a boy is taken to Pakistan by his mother. Sparking a five-year journey through multiple countries before seeking asylum in Italy."

№1 Текст: In the Sea There are Crocodiles: Based on the True Story of Enaiatollah Akbari	Опис: When a ten-year-old boy's village in Afghanistan falls prey to Taliban rule, his mother shepherds the boy across the border into Pakistan but has to leave him there all alone to fend for himself. Thus begins Enaiat's remarkable and often punishing five...	Відповідність:	54.05%
№2 Текст: Notes from My Travels: Visits with Refugees in Africa, Cambodia, Pakistan and Ecuador	Опис: From the ever-intriguing and appealing actress Angelina Jolie comes the personal journals she compiled while performing humanitarian relief efforts in such countries as Sierra Leone and Tanzania, Pakistan and Cambodia. Three years ago, award-winning a...	Відповідність:	41.45%
№3 Текст: For the Love of a Son: One Afghan Woman's Quest for Her Stolen Child	Опис: From the time she was a little girl, Maryam rebelled against the terrible second-class existence that was her destiny as an Afghan woman. Maryam had witnessed the miserable fate of her grandmother and three aunts, and wished that she had been born a b...	Відповідність:	39.92%
№4 Текст: A Map Is Only One Story: Twenty Writers on Immigration, Family, and the Meaning of Home	Опис: From rediscovering an ancestral village in China to experiencing the realities of American life as a Nigerian, the search for belonging crosses borders and generations. Selected from the archives of Catapult magazine, the essays in A Map Is Only One ...	Відповідність:	36.01%
№5 Текст: The Seven	Опис: One of the most dramatic stories of genetic discovery since James Watson's The		

Результати пошуку твору "В морі є крокодили" в інтелектуальному модулі

Результати пошуку - "In a society dominated by television and where literature is fading, certain individuals perform unconventional duties. Among them is a man tasked with a peculiar responsibility—to eradicate forbidden objects, without question. However, an encounter with an intriguing acquaintance sparks a subtle shift in his perception, leading him to reconsider the truths he once took for granted. As unexpected events unfold, he finds himself questioning the very essence of his existence."

№1 Аудіо: Fahrenheit 451	Опис: Guy Montag is a fireman. In his world, where television rules and literature is on the brink of extinction, firemen start fires rather than put them out. His job is to destroy the most illegal of commodities, the printed book, along with the houses i...	Відповідність:	45.45%
№2 Текст: On The Intuitive Spectrum: A Deeper Look into the Amazing Value of Intuition	Опис: The sun god Apollo sentenced Cassandra to be able to prophesy her intuitions but to never be believed. Since ancient times, the gift of prophecy and intuition was a present that was bestowed upon Cassandra by Apollo. Apollo loved Cassandra, and she r...	Відповідність:	32.98%
№3 Текст: When the Heart Waits: Spiritual Direction for Lifes Sacred Questions	Опис: I stood at the window watching the cocoon, which hung in the winter air like an upside-down question mark. That was the moment... I understood. Really understood. Crisis, change, all the myriad upheavals that blister the spirit and leave us groping- ...	Відповідність:	32.89%
№4 Аудіо: Chicken Soup for the Soul: Miracles Happen: 101 Inspirational Stories about Hope, Answered...	Опис: Miracles happen every day—we just have to look to see them. These powerful stories will give you hope and deepen your faith. These 101 amazing stories prove that miracles happen, every day, to people from all walks of life. You will find hope and fait...	Відповідність:	32.81%

Результати пошуку аудіо книги “451 градус за Фаренгейтом” в інтелектуальному модулі



Меню налаштування ваг інтелектуального модуля пошуку різноформатних матеріалів у базах даних

Результати пошуку - "In a society dominated by television and where literature is fading, certain individuals perform unconventional duties. Among them is a man tasked with a peculiar responsibility—to eradicate forbidden objects, without question. However, an encounter with an intriguing acquaintance sparks a subtle shift in his perception, leading him to reconsider the truths he once took for granted. As unexpected events unfold, he finds himself questioning the very essence of his existence."

№1 Текст: The Unlikely Ones

Опис: A band of outcasts begins an arduous journey through a world of evil witches, walking trees, and miraculous gems along a path that will reunite them with their true destinies....

Відповідність: 63.78%

№2 Текст: Pictures of Hollis Woods

Опис: In this Newbery Honor Book, a troublesome 12-year-old orphan, staying with an elderly artist who needs her, remembers the only other time she was happy in a foster home, with a family that truly seemed to care about her....

Відповідність: 63.53%

№3 Текст: Tegami Bachi: Letter Bee, Volume 1

Опис: In Amberground, a dangerous terrain where a man-made star casts a permanent twilight, young Lag Seeing aspires to become a Letter Bee: a postman entrusted to deliver the hearts of people separated from the ones they love....

Відповідність: 62.98%

№4 Текст: Un mago de Terramar / Las Tumbas de Atuan

Опис: En estas dos primeras novelas de la saga, las aventuras vividas por el joven mago Ged y la sacerdotisa Tenar los transformarán profundamente, les harán crecer y liberarse de sus miedos, y se convertirán en auténticos héroes que ayudarán a restablecer...

Відповідність: 62.31%

Результати пошуку аудіо книги “451 градус за Фаренгейтом” в інтелектуальному модулі зі зміненими початковими вагами