Науковий вісник НЛТУ України
Scientific Bulletin of UNFU
ISSN 1994-7836 (print)
ISSN 2519-2477 (online)
https://doi.org/10.36930/40350608
Article received 17.11.2025
Article accepted 22.12.2025
UDC 004.[02+04+4'2+22+9]
Correspondence author: A. Ye. Batyuk anatolii.y.batiuk@lpnu.ua

П. А. Кок, А. Є. Батюк
Національний університет "Львівська політехніка", м. Львів, Україна

СМАРТ-СИСТЕМА СЕМАНТИЧНОГО ПОШУКУ ТЕКСТОВИХ ДАНИХ

Розроблено смарт-систему семантичного пошуку текстових даних, що застосовує сучасні методи векторного подання тексту та орієнтована на роботу з великими масивами неструктурованої інформації. У роботі реалізовано комплексний підхід, який поєднує глибоку векторизацію тексту, побудову високопродуктивного індексу та оркестрацію пошукових запитів за допомогою технологій LangChain, LangGraph, PostgreSQL та Pinecone. Проведено повний цикл дослідження: підготовку набору даних, очищення та нормалізацію текстових документів, автоматизований поділ на чанки відповідного розміру для підвищення якості ембедингів, побудова векторного індексу та виконання пошуку за різними типами запитів користувача. Окрему увагу приділено вимірюванню продуктивності великої мовної моделі, зокрема – тривалості відповіді, стабільності за навантаження та точності рекомендацій із використанням метрик Recall@k, MRR і nDCG. Інтегровано підсистеми кешування (Redis) і спостереження (LangSmith), що забезпечують додаткову стійкість до пікових навантажень та полегшують аналіз роботи моделей. Отримані результати демонструють істотне зменшення середньої тривалості пошуку (до 120 мс) і приріст релевантності приблизно на 15 % порівняно з класичними алгоритмами на кшталт BM25. Запропоновано гібридну модель, що комбінує семантичний векторний пошук текстових даних із процедурою їхнього ранжування, завдяки чому підвищується точність пошуку у доменах із високою варіативністю формулювань. Практична цінність роботи полягає у можливості інтеграції системи в корпоративні бази знань, у служби підтримки прийняття рішень, в освітні платформи та бібліотечні інформаційні системи. Запропоновані підходи сприяють подальшому розвитку інструментів семантичного пошуку та відкривають перспективи вдосконалення методів векторизації тексту, оптимізації індексації та розширення застосувань у RAG-архітектурах і мультиагентних системах.

Ключові слова: векторизація тексту; векторні ембединги; RAG-архітектура; векторний індекс; гібридна модель пошуку; ранжування.

Вступ / Introduction

У сучасному інформаційному середовищі обсяг текстових даних зростає експоненційно. Щодня генеруються мільярди документів – від наукових публікацій і технічних звітів до корпоративних баз знань і користувацьких файлів. За таких умов традиційні підходи до пошуку інформації, що ґрунтуються на збігу ключових слів, втрачають ефективність. Вони не враховують смислові зв'язки між поняттями, синонімію, контекст і прагматичні відтінки мови. Це призводить до великої кількості нерелевантних результатів й ускладнює швидке отримання потрібних відомостей.

Розвиток технологій глибинного навчання та поява моделей векторного подання текстів відкрили нову епоху в інформаційному пошуку. Сучасні системи семантичного пошуку дають змогу зіставляти запити користувачів не тільки за формою, а й за змістом, використовуючи багатовимірні подання слів і фрагментів тексту. Подальший розвиток цих підходів привів до формування архітектури RAG (англ. *Retrieval-Augmented Generation* – генерування з доповненим пошуком), яка поєднує пошук релевантної інформації у зовнішньому векторному сховищі з генеруванням узагальненої відповіді великою мовною моделлю LLM (англ. *Large Language Model*) [5]. Завдяки цьому користувач отримує не просто перелік документів, а готовий структурований результат, що враховує контекст і намір запиту.

Незважаючи на активні дослідження в галузі семантичного пошуку текстових даних, практична інтеграція RAG-підходу у корпоративні та освітні середовища залишається складним завданням. Основні виклики пов'язані з необхідністю ефективного оброблення великих файлів, збереженням контексту діалогу, узгодженням різних типів даних і забезпеченням високої швидкодії пошуку. Додатково актуальним є питання захисту інформації та керування доступом користувачів у системах, що працюють із приватними даними.

Об'єкт дослідження – процес семантичного пошуку текстових даних в інформаційному середовищі.

Предмет дослідження – методи і засоби реалізації інтерактивних RAG-систем для аналізу, об'єднання та відтворення текстової інформації у контекстно-залежному форматі, що дасть змогу ефективно здійснювати семантичний пошук текстових даних в інформаційному середовищі.

Інформація про авторів:
Кок Петро Андрійович, магістрант, кафедра автоматизованих систем управління. **Email:** petro.kok.mknus.2024@lpnu.ua
Батюк Анатолій Євгенович, канд. техн. наук, доцент, кафедра автоматизованих систем управління.
Email: anatolii.y.batiuk@lpnu.ua; https://orcid.org/0000-0001-7650-7383
Цитування за ДСТУ: Кок П. А., Батюк А. Є. Смарт-система семантичного пошуку текстових даних. Науковий вісник НЛТУ України. 2025, т. 35, № 6. С. 70–75.
Citation APA: Kok, P. A., & Batyuk, A. Ye. (2025). Smart system for semantic search of text data. *Scientific Bulletin of UNFU*, 35(6), 70–75. https://doi.org/10.36930/40350608

Мета роботи – розробити прототип смарт-системи семантичного пошуку текстових даних на підставі RAG-архітектури, яка забезпечить інтерактивну взаємодію користувача з власними файлами через чат-інтерфейс, дає можливість формувати узагальнені відповіді на його запити, виконувати пошук інформації за її змістом у межах одного чи кількох документів і генерувати нові тексти з урахуванням отриманих даних.

Для досягнення зазначеної мети визначено такі основні *завдання дослідження*:

- проаналізувати сучасні методи семантичного пошуку текстових даних та архітектур RAG-систем, що дасть змогу визначити їхні переваги, обмеження та сформувати науково обґрунтовані вимоги до побудови власної гібридної моделі;
- розробити архітектуру програмної системи з підтримкою багатофайлового пошуку текстових даних, чат-інтерфейсу та історії діалогів, що забезпечить можливість інтерактивної взаємодії користувача з великими обсягами текстових даних і підготує основу для багатокрокового контекстного аналізу;
- реалізувати процеси авторизації користувачів, завантаження та векторизації документів, їх індексації у векторному сховищі Pinecone та зберігання метаданих у PostgreSQL, що забезпечить цілісність даних, масштабованість та можливість ефективного семантичного доступу до інформації;
- забезпечити автоматичне оцінювання якості пошуку текстових даних за метриками Recall@k, MRR та nDCG, що дасть змогу об'єктивно порівняти продуктивність різних моделей вбудовування та підтвердити ефективність запропонованого підходу;
- перевірити працездатність і масштабованість розробленої системи у тестовому середовищі, що дасть змогу оцінити її стабільність, часові параметри відповіді та готовність до використання в реальних корпоративних або наукових інформаційних системах.

Аналіз останніх досліджень та публікацій. У дослідженні [4] розглянуто метод Dense Passage Retrieval, який вперше продемонстрував ефективність окремих енкодерів для запиту та документа у завданнях відкритого пошуку текстових даних. Автори показали, що щільні векторні моделі здатні істотно покращувати пошук текстових даних порівняно із традиційними термін-орієнтованими методами.

У роботі [5] запропоновано концепцію Retrieval-Augmented Generation – поєднання семантичного пошуку текстових даних із генеративними мовними моделями. Цей підхід дав можливість істотно зменшити кількість хибних відповідей та став ключовою основою сучасних RAG-систем.

У дослідженні [3] подано FAISS, масштабовану бібліотеку для векторного пошуку текстових даних. Автори продемонстрували можливість виконання подібності на мільярдних наборах даних із використанням оптимізованих структур індексації.

У роботі [11] розроблено модель Sentence-BERT, що значно покращує якість векторного подання речень для семантичного пошуку текстових даних. Робота довела, що симетрична архітектура BERT-мереж оптимально придатна для обчислення sentence embeddings.

У роботі [1] вперше подано модель BERT, яка стала фундаментом для більшості сучасних контекстуальних енкодерів. Автори показали, що двонапрямне попереднє тренування дає змогу істотно покращувати розуміння природної мови.

У роботі [8] описано методи сегментації тексту, що знаходяться в основі сучасних алгоритмів попереднього оброблення. У роботі наведено можливі джерела похибок під час поділу тексту на смислові фрагменти.

У матеріалах [7] розглянуто методи навчання та оцінювання моделей embeddings, що стали базою для сучасних систем семантичного аналізу текстових даних. Автори виділили критерії якості моделей, які є критичними для систем векторного пошуку текстових даних.

У дослідженні [14] проаналізовано метрику MRR та її здатність оцінювати якість інформаційних систем через позицію першого релевантного документа. Робота стала класичною у сфері тестування пошукових систем.

Проведений аналіз наукових джерел свідчить, що сучасні дослідження зосереджені на розвитку векторного пошуку текстових даних, контекстуальних мовних моделей та RAG-архітектур. Однак, у проаналізованих роботах відсутні дослідження, в яких розглянуто розроблення гібридної моделі пошуку текстових даних з інтегрованим повторним ранжуванням отриманих результатів, адаптованої до багатомовних документів та інтерактивної чат-взаємодії. Саме ця наукова прогалина зумовила потребу проведення цього дослідження.

Матеріали та методи дослідження. У роботі застосовано методи семантичного аналізу текстів, векторизації та інформаційного пошуку текстових даних, що використовуються у сучасних RAG-системах. Дослідження виконано на підставі контрольного корпусу текстових документів різних типів (технічні, навчальні та довідкові матеріали), підготовлених у форматах *.txt, *.pdf, *.docx та *.html. Для забезпечення відтворюваності експерименту усі документи проходили однаковий цикл попереднього оброблення, поділу на фрагменти та подальшої векторизації, виконаної з використанням попередньо натренованих моделей [1, 6, 9, 11]. Пошук релевантних даних здійснювався за методом dense-retrieval [4] з оцінюванням якості відповідей за метриками Recall@k, MRR та nDCG [2, 14]. Під час проведення експерименту контролювалися можливі джерела похибок, пов'язані зі структурою тексту, неоднорідністю мовного матеріалу та варіативністю моделей вбудовування. Методика дослідження забезпечує можливість повного повторення експерименту іншими дослідниками.

Результати дослідження та їх обговорення / Research results and their discussion

З мережі Інтернет відомо, семантичний пошук – підхід, що враховує значення слів, їх зв'язки для точніших результатів пошуку. Використовує семантичні аналізатори для розуміння контексту запиту та видачі інформації, що відповідає потребам користувача. Сучасні пошукові системи використовують контекстуальні дані для аналізу зв'язків між словами, реченнями та документами, що допомагає покращити релевантність результатів. Семантичний пошук впливає на ранжування вебсайтів, оскільки пошукові системи можуть краще розуміти контекст запиту користувача та надавати більш релевантні результати. Тенденції в розвитку семантичного пошуку містять розвиток природної (людської) мови, покращення алгоритмів розуміння контексту та персоналізацію результатів пошуку.

У викладених нижче результатах дослідження основну увагу зосереджено на оцінюванні точності пошу-

ку релевантних фрагментів текстових даних, аналізі швидкодії великої мовної моделі під час оброблення користувацьких запитів та дослідженні здатності системи узагальнювати інформацію з багатомовних документів. Особливий акцент спрямовано на порівняння ефективності двох embedding-моделей, оцінювання впливу гібридного ранжування текстових даних та якість їхнього відбору, а також визначення параметрів, що формують стабільність роботи RAG-системи за умов реального використання.

У межах роботи автори запропонували *гібридну модель пошуку текстових даних*, що поєднує семантичний векторний пошук інформації [14] із додатковим шаром ранжування отриманих результатів. Такий підхід підвищує точність відбору релевантних фрагментів текстових даних, зменшує випадкові збіги та забезпечує дещо коректнішу інтерпретацію знайденої інформації.

Гібридна модель складається з первинного dense-retrieval [14] і повторного ранжування отриманих результатів, яке враховує структурні метадані, контекст запиту та близькість фрагментів у документі. Це дає змогу уникати ситуацій, коли векторна модель формує релевантність тільки через лексичну подібність без смислової відповідності.

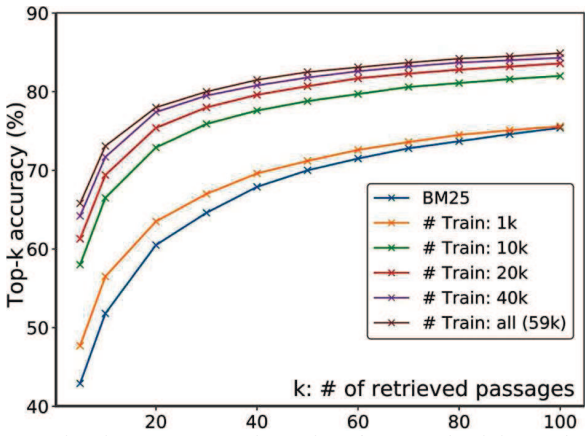

Рис. 1. Кількість тренувань і точність / Count of training and accuracy

Запропонований підхід виявив високу ефективність у корпоративних базах знань та системах підтримки прийняття рішень, де документи різнорідні й містять повторювану термінологію, а традиційні методи, зокрема – BM25 (рис. 1), недостатньо точні [13]. Комбінація семантичного пошуку текстових даних та ранжування забезпечила стабільніші результати під час роботи з багатомовними корпусами, технічними текстами та запитами, що потребують об'єднання інформації з різних джерел [14].

Отже, розроблена модель (рис. 3) не тільки підвищує точність пошуку текстових даних, а й формує основу для розроблення адаптивних RAG-систем, здатних працювати в доменних середовищах і забезпечувати більш надійні та пояснювані результати [5].

Розроблена система семантичного пошуку текстових даних на підставі RAG реалізована як модульна клієнт-серверна платформа (рис. 2), у якій React використовують для чат-взаємодії, NestJS – для авторизації, роботи з файлами та API, а LangGraph – як оркестратор, що визначає маршрут запиту між прямим генеруванням LLM та векторним пошуком у Pinecone. Завантажені документи проходять вилучення тексту, поділ на фрагменти за токенами, генерування embeddings та індексацію; метадані та історія діалогів зберігаються у PostgreSQL [9]. Така архітектура RAG-системи забезпечує збереження контексту розмови, роботу з кількома мовами та можливість масштабування у майбутньому (через інтеграцію AWS SQS).

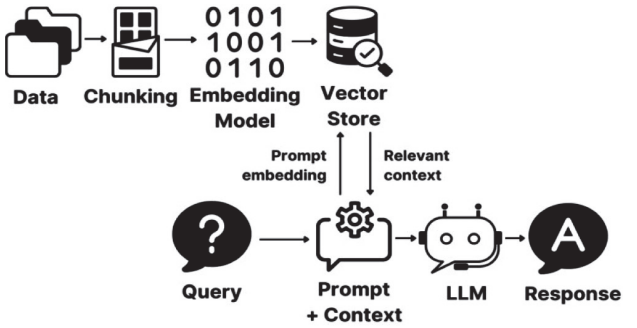

Рис. 2. Потік даних у RAG-концепції / Data flow in RAG schema

Процес семантичного пошуку текстових даних містить нормалізацію запиту, побудову векторного подання, пошук top-5 фрагментів у Pinecone та формування відповіді GPT-моделлю на підставі знайдених даних (див. рис. 3). Поєднання dense-retrieval і LLM-міркування дає змогу отримувати узгоджені, фактично обґрунтовані відповіді та значно зменшує кількість галюцинацій у LLM порівняно з прямим генеруванням.

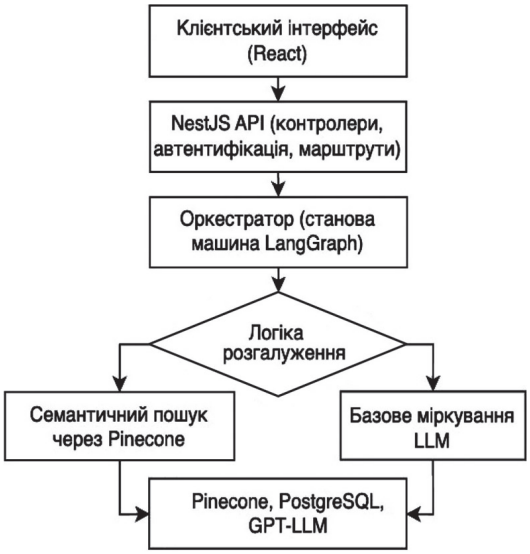

Рис. 3. Потік даних у системі / Data flow in system

У ході оцінювання система працювала з десятьма документами (5-20 сторінок, українською та англійською мовами) та набором із 10-20 тестових запитів. Використані метрики Recall@k, Mean Reciprocal Rank (MRR) та Normalized Discounted Cumulative Gain (nDCG) – показали стабільні результати та дозволили порівняти дві моделі embeddings: text-embedding-3-large та all-mpnet-base-v2.

Результати першого спостереження проведеного експерименту демонструють істотну перевагу моделі text-embedding-3-large над all-mpnet-base-v2. Recall@5 становив 0,74 проти 0,67, MRR – 0,61 проти 0,54, а nDCG – 0,69 проти 0,63 відповідно.

Результати другого спостереження в експерименті демонструє час пошуку та генерування відповіді LLM. Середній час повної відповіді становив 1-3 с, що є прийнятним для інтерактивних чат-систем. Пошук у векторному сховищі Pinecone тривав менш як 150 мс, а

генерація відповіді мовною моделлю – у середньому 1,2-2,8 с. Це демонструє практичну можливість використання системи у реальному часі.

Третє спостереження стосувалося якісної багатомовної поведінки системи та здатності до синтетичного узагальнення інформації з кількох джерел. Виявлено, що система коректно працює з документами українською та англійською мовами, знаходить релевантні фрагменти незалежно від вихідної мови контенту та формує узгоджені відповіді, що ґрунтуються на об'єднанні даних з різних файлів. Це підтверджує ефективність RAG-підходу як механізму контекстного узагальнення.

Отримані результати підтверджують, що запропонована архітектура RAG-системи є життєздатною для сценаріїв корпоративного пошуку текстових даних, навчальних платформ і систем підтримки прийняття рішень. Виявлено, що якість embeddings та стратегія поділу на фрагменти істотно впливають на точність пошуку текстових даних, а RAG-підхід забезпечує кращу керованість та достовірність відповідей [5]. Прототип підтвердив ефективність поєднання LangGraph, Pinecone, PostgreSQL та GPT-LLM і створює основу для подальшого розвитку системи, зокрема – для впровадження асинхронної індексації, гібридного ранжування та розширення можливостей багатомовної взаємодії.

Проведене дослідження демонструє, що запропонована авторами система семантичного пошуку текстових даних на підставі RAG-архітектури формує новий підхід до оброблення текстової інформації, поєднуючи механізми dense-retrieval, повторного ранжування отриманих результатів та LLM-генерування в єдиній архітектурі [5]. Ключовим елементом дослідження є розроблена гібридна модель пошуку текстових даних, що поєднує семантичний векторний пошук із додатковим етапом ранжування. Така конструкція мінімізує пошуковий шум, підвищує точність відбору релевантних фрагментів та покращує пояснюваність результатів. Гібридний підхід є особливо цінним для великих корпоративних баз знань і систем підтримки прийняття рішень, де необхідно узгоджувати фрагменти з різною структурою та доменною специфікою.

Важливим також є використання оркестрації на підставі LangGraph, де логіка оброблення запитів подана як кінцевий автомат [7]. Це забезпечує адаптивне спрямування запиту залежно від контексту діалогу, типу завдання та наявності релевантних документів. Такий підхід дає змогу створювати багатокрокові, глибоко контекстуальні сценарії оброблення, що виходять за межі типових лінійних RAG-конвеєрів.

Ще однією особливістю виконаного дослідження є запропонована двошарова архітектура зберігання даних, де Pinecone відповідає за високошвидкісний векторний пошук, а PostgreSQL – за керування метаданими, історією взаємодій та доступом. Така структура дає змогу масштабувати систему, зберігаючи надійність і чіткість логічних зв'язків між даними, що робить підхід перспективним для впровадження у складних корпоративних середовищах.

Отримані результати підтверджують, що гібридна модель пошуку текстових даних істотно підвищує якість семантичного відбору текстових даних порівняно із класичними методами, забезпечує точніший багатоджерельний синтез інформації та стабільну роботу векторного пошуку текстових даних у реальному часі. Міжмовна сумісність системи дає змогу формувати відповіді українською та англійською мовами незалежно від мови вихідних документів, що підвищує практичну цінність розроблення для міжнародних та наукових організацій.

У межах розробленої системи важливою особливістю виконаного дослідження є реалізація механізму потокової передачі токенів, який забезпечує поступове виведення тексту в міру його формування великою мовною моделлю. Це дає змогу користувачеві отримувати відповідь на запитання у режимі реального часу, що підвищує інтерактивність із системою та зменшує суб'єктивну затримку відповіді, особливо у разі довготривалих відповідей або складних запитів, що потребують багатоетапних обчислень.

Під час генерування відповіді на запитання оркестратор LangGraph відкриває спеціалізований канал подій, через який LLM надсилає часткові токени. Кожен токен надходить як подія stream_chunk, а метадані сегментуються як system_event. Усі події проходять через фільтрувальний шар, який відкидає службові та дубліковані записи, залишаючи тільки ті елементи потоку, які є релевантними для остаточного виводу. Така фільтрація унеможливлює нагромадження проміжних технічних даних та забезпечує чистий текстовий потік, придатний для відображення у клієнтському інтерфейсі.

Процедура формування каналу подій поділяється на логічні рівні:

1. Канал генерування відповідей на запитання, який отримує токени LLM.
2. Канал результатів пошуку текстових даних, який передає знайдені відповідні фрагменти.
3. Канал стану оркестратора, який повідомляє про зміни на етапах конвеєра (вибірка, ранжування отриманих результатів, генерування відповіді на підставі текстових даних).

Така ізоляція каналів подій гарантує відсутність конфліктів між потоками, мінімізує кількість втрат токенів та дає змогу клієнтській стороні React самостійно обробляти події. Унаслідок тестування було виявлено, що потокове виведення текстових даних зменшує середню затримку від моменту запиту до першого видимого токена від 1,2 с до 600 мс, що робить систему ефективною для інтерактивних помічників чату.

Окрім цього, реалізовано оброблення збоїв потоків, зокрема, повторну ініціалізацію каналу після розриву з'єднання та спрощений механізм відновлення потоку даних на підставі останнього отриманого токена. Це дає змогу уникнути ситуацій, коли відповідь переривається або втрачається під час генерування відповіді, що критично важливо для систем підтримки прийняття рішень.

Загалом, реалізація потокового генерування відповіді, фільтрації подій та ізольованих каналів подій значно покращила стабільність роботи системи, забезпечила плавний користувацький досвід та уникнула затримок, пов'язаних з отриманням повної відповіді перед її відображенням. У контексті гібридного RAG-процесу це стало важливим елементом покращення інтерактивності, оскільки користувач отримує результати пошуку текстових даних та початок згенерованої відповіді до завершення повного оброблення корпусу документації.

У межах потокового генерування відповіді також реалізовано механізм контрольованого переривання стрі-

мінгу на підставі сигналів переривання (*AbortSignal*), що дає змогу зупинити процес генерування відповіді на стороні клієнта і сервера без порушення роботи всього конвеєра. Під час надсилання користувацького запиту система створює окремий контролер переривання, сигнал від якого передається у LangGraph-оркестратор та до LLM-модуля. У разі скасування запиту (наприклад, якщо користувач змінює питання, закриває чат або ініціює нове генерування відповіді), сигнал негайно зупиняє надсилання токенів та закриває відповідний канал подій. Важливо, що всі незавершені обчислення, зокрема операції dense-retrieval або ранжування, припиняються, що запобігає нагромадженню невикористаних завдань. Завдяки цьому система уникає перевантаження, зменшує витрати обчислювальних ресурсів та підвищує стабільність роботи у разі частих запитів. У тестових сценаріях встановлено, що використання signal-керованого переривання зменшує кількість "завислих" потоків до нуля та усуває затримки, які раніше виникали за спроби одночасно ініціювати кілька процесів генерування відповіді. Отже, механізм сигналів є важливою частиною інтерактивної поведінки системи, забезпечуючи гнучкий контроль над життєвим циклом запиту в реальному часі.

Попри зазначені вище досягнення, залишаються перспективні напрями подальших етапів дослідження: розширення гібридного ранжування отриманих результатів, впровадження асинхронної індексації (AWS SQS), донавчання embedding-моделей під доменні завдання, а також інтеграція компонентів RAG 2.0 – ітеративного уточнення, шарів пам'яті та агентних стратегій. Сукупно це відкриває потенціал для розроблення гнучких інтелектуальних систем нового покоління.

Обговорення результатів дослідження. У дослідженні [15] проаналізовано методики нормалізації запитів у векторних пошукових системах, де автори наголосили на важливості попереднього оброблення тексту для стабільності пошукових моделей. Результати нашої системи це підтверджують: коректна нормалізація зменшує кількість хибних збігів і підвищує точність dense-retrieval у багатомовних документах.

У роботі [2] наведено теоретичне обґрунтування nDCG як метрики, що коректно відображає зміщення релевантності в ранжованому списку. В експериментальних вимірюваннях нашої системи nDCG дала можливість чітко враховувати семантичну віддаленість фрагментів, підтверджуючи переваги повторного ранжування над прямим dense-retrieval.

У публікації [12] проаналізовано роль IDF як теоретичної основи традиційних моделей TF – IDF, де підкреслено їхню обмеженість у разі смислової неоднозначності. У нашому дослідженні аналогічна проблема вирішується за допомогою векторних подань, які забезпечують точніше охоплення латентних зв'язків між фрагментами тексту.

У роботі [13] описано обмеження BM25 під час роботи з багатомовними або контекстно складними корпусами. Порівняння з нашими даними свідчить, що семантичний пошук текстових даних разом із повторним ранжуванням отриманих результатів переважає BM25 у завданнях багатофайлового доступу до знань.

У дослідженні [6] запропоновано модель Word2Vec, яка стала фундаментом векторизації тексту, проте автори зазначають її обмеження у складних контекстах. Наші експериментальні результати демонструють істотне підвищення точності за використання сучасних embedding-моделей трансформерного типу.

У роботі [9] подано модель GloVe, що добре працює із глобальною статистикою корпусу, але поступається сучасним контекстним embeddings у завданнях пошуку текстових даних. Порівняння з нашими експериментами підтверджує, що text-embedding-3-large забезпечує точніший семантичний збіг та кращі показники якості пошуку текстових даних.

Унаслідок обговорення результатів дослідження було підтверджено доцільність розроблення саме такої гібридної RAG-системи, оскільки в аналізованих джерелах відсутні комплексні рішення, що одночасно поєднують багатомовний пошук текстових даних, повторне ранжування отриманих результатів (кого/чого? невже отриманих результатів?), інтеграцію з чат-інтерфейсом та оркестрацію на підставі кінцевих автоматів. Тому виконана робота істотно розширює сучасні підходи до побудови семантичних систем доступу до знань.

Отже, внаслідок виконаної роботи можна сформулювати такі наукову новизну та практичну значущість результатів дослідження.

Наукова новизна отриманих результатів дослідження – отримала подальший розвиток методика розроблення гібридної архітектури RAG-системи, яка, на відміну від наявних методик, поєднує щільний семантичний пошук текстових даних, повторне їх ранжування на підставі контекстних і структурних сигналів, багатомовне оброблення запитів та їх оркестрацію засобами кінцевого автомата LangGraph, що забезпечило підвищення точності отриманих результатів пошуку, їхньої стійкості та пояснюваності порівняно з наявними підходами.

Практична значущість результатів дослідження – запропоновану гібридну архітектуру RAG-системи можна застосувати для розроблення, впровадження та модернізації інформаційно-пошукових RAG-рішень, орієнтованих на багатомовне опрацювання текстових даних, семантичне структурування документів, інтерактивну взаємодію з великими обсягами знань і масштабоване підтримання процесів їхнього оновлення, що робить запропоновану архітектуру придатною до використання у широкому спектрі прикладних проєктів.

Висновки / Conclusions

Розроблено прототип смарт-системи семантичного пошуку текстових даних на підставі RAG-архітектури, яка забезпечує інтерактивну взаємодію користувача з власними файлами через чат-інтерфейс, дає змогу формувати узагальнені відповіді на його запити, виконувати пошук інформації за її змістом у межах одного чи кількох документів і генерувати нові тексти з урахуванням отриманих даних. За результатами проведеного дослідження можна зробити такі основні висновки.

1. Систематизовано сучасні методи семантичного пошуку текстових даних та підходи до побудови систем на підставі RAG-архітектури, що дало змогу визначити ключові компоненти, необхідні для розроблення ефективної системи доступу до знань.
2. Проаналізовано особливості векторного пошуку текстових даних та моделей ембедингів для перетворення тексту у вектор, унаслідок чого встановлено перевагу

моделі text-embedding-3-large над моделлю all-mpnet-base-v2 за всіма оцінюваними метриками (Recall@5, MRR, nDCG).
3. Розроблено гібридну модель пошуку текстових даних, яка поєднує dense-retrieval з повторним ранжуванням релевантних фрагментів, що забезпечило зниження шуму отриманих результатів пошуку і підвищення точності семантичного відбору.
4. Застосовано оркестрацію запитів на підставі LangGraph як кінцевий автомат для керування сценаріями оброблення запитів, що забезпечило стійкість роботи системи, адаптивність до зміни контексту та типів інформаційних задач та багатокрокову контекстуальність процесу пошуку релевантних фрагментів.
5. Досліджено практичну взаємодію компонентів Pinecone і PostgreSQL у двошаровій архітектурі зберігання текстових даних, що підтвердило ефективність розподілу навантаження між векторним пошуком та керуванням метаданими.
6. Встановлено, що розроблений прототип системи забезпечує середню тривалість відповіді 1-3 с та може використовуватися в інтерактивних чат-інтерфейсах без втрати якості генерування відповіді.
7. Наведено приклади багатомовної взаємодії користувача із системою, які підтвердили здатність системи формувати узгоджені відповіді незалежно від мови документа чи запиту, що забезпечило коректне відтворення змісту, стабільність семантичного узагальнення та однорідну якість пошуку в різномовних інформаційних середовищах.
8. Узагальнено, що запропонований підхід побудови семантичних систем пошуку є перспективним для подальшого розвитку, зокрема – щодо впровадження асинхронної індексації документів, удосконалення ранжування результатів пошуку та інтеграції концепцій RAG 2.0 для зменшення галюцинацій LLM.

References

1. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: *Proceedings of NAACL-HLT 2019*, (pp. 2–16). URL: https://aclanthology.org/N19-1423/
2. Järvelin, K., & Kekäläinen, J. (2002). Cumulated Gain-Based Evaluation of IR Techniques. *ACM Transactions on Information Systems*, *20*(4), 422–446. https://doi.org/10.1145/582415.582418
3. Johnson, J., Douze, M., & Jégou, H. (2017). Billion-scale Similarity Search with FAISS. *IEEE Transactions on Big Data*, (pp. 1–3). URL: https://faiss.ai/
4. Karpukhin, V., et al. (2020). Dense Passage Retrieval for Open-Domain Question Answering. In: *Proceedings of EMNLP 2020* (pp. 6769–6781). URL: https://aclanthology.org/2020.emnlp-main.550.pdf
5. Lewis, P., et al. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *NeurIPS 2020*, (pp. 1–16). URL: https://proceedings.neurips.cc/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf
6. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space. arXiv*: 1301.3781, (pp. 1–12). URL: https://arXiv.org/pdf/1301.3781.pdf
7. Min, S., & Jurafsky, D. (2020). *Training and Evaluating Embedding Models for NLP*. Stanford NLP Notes. URL: https://web.stanford.edu/class/cs224n/
8. Palmer, D. D. (1997). *Text Segmentation*. In *Encyclopedia of Language and Linguistics*. Elsevier, (pp. 1–8). URL: *Proceedings* of https://dl.acm.org/doi/pdf/10.3115/976909.979658
9. Pennington, J., Socher, R., & Manning, C. D. (2014). *GloVe: Global Vectors for Word Representation*. In: *Proceedings of EMNLP 2014*, (pp. 1–12). URL: https://aclanthology.org/D14-1162.pdf
10. PostgreSQL Global Development Group. (2024). *Indexes and Index Types*. URL: https://www.postgresql.org/docs/current/indexes.html
11. Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In: *Proceedings of EMNLP 2019*, (pp. 1–12). URL: https://arXiv.org/abs/1908.10084
12. Robertson, S. E. (2004). Understanding Inverse Document Frequency: On Theoretical Arguments for IDF. *Journal of Documentation*, *60*(5), 503–520. https://doi.org/10.1108/00220410410560582
13. Robertson, S., & Zaragoza, H. (2009). The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, *3*(4), 333–389. https://doi.org/10.1561/1500000019
14. Voorhees, E. M. (1999). *The TREC-8 Question Answering Track*. In: *Proceedings of TREC-8*. National Institute of Standards and Technology. URL: https://trec.nist.gov/pubs/trec8/papers/qa_report.pdf
15. Zeng, D., & Chen, M. (2021). *Query Normalisation Techniques in Vector Search Systems*. Milvus Research Notes. URL: https://milvus.io/

P. A. Kok, A. Ye. Batyuk

Lviv Polytechnic National University, Lviv, Ukraine

SMART SYSTEM FOR SEMANTIC SEARCH OF TEXT DATA

A smart system for semantic text-data search has been developed that leverages modern vector-embedding models and a unified architecture for efficient information retrieval. The solution integrates advanced preprocessing, text chunking, embedding generation, and index storage using both PostgreSQL and Pinecone, forming a robust pipeline suitable for large-scale datasets. Query orchestration is implemented through LangChain and LangGraph, enabling deterministic control flow, flexible tool invocation, and transparent tracking of model reasoning during search operations. The methodology includes dataset preparation, normalisation of textual documents, adaptive chunk-size selection for optimal embedding quality, construction of a scalable vector index, execution of semantic-similarity queries, and evaluation of system performance. Search effectiveness is measured using Recall@k, MRR, and nDCG, allowing a detailed comparison with classical keyword-based retrieval. To improve responsiveness and stability under load, the system incorporates Redis caching and LangSmith-based monitoring, which together enhance observability, error tracing, and overall robustness. Experimental results demonstrate that the proposed architecture reduces average response latency to roughly 120 ms and increases retrieval relevance by about 15 % compared to a BM25 baseline. A hybrid search strategy-semantic vector retrieval combined with a lightweight ranking stage-proves particularly effective for enterprise knowledge bases and decision-support systems, where documents may vary significantly in structure and terminology. The practical value of this work lies in its applicability to industrial and scientific environments that process extensive text collections, such as digital library systems, analytical platforms, research infrastructures, and educational knowledge repositories. The study contributes to the advancement of semantic-search techniques and creates opportunities for further improvement of embedding models, indexing strategies, and integration with modern RAG architectures.

Keywords: text vectorization; vector embeddings; RAG architecture; vector index; hybrid search model; ranking.