ЦИФРОВА ЕКОНОМІКА ТА
ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ
Digital Economy and Information Technologies

УДК 004.89:004.93'1

ЗАСТОСУВАННЯ МЕТОДОЛОГІЇ RAG ДЛЯ РОЗШИРЕННЯ МОЖЛИВОСТЕЙ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ

Наталія Полуектова,
Кафедра інформаційних технологій
Запорізький інститут економіки та інформаційних технологій
Запоріжжя, Україна

Анотація – У роботі озглянуто методологію RAG (Retrieval-Augmented Generation), яка дозволяє синергетично поєднувати параметричні знання моделей із зовнішніми динамічними базами даних. Описано ключові етапи RAG: індексацію (перетворення документів у векторні представлення), семантичний пошук та генерацію відповіді на основі релевантного контексту. У практичній частині реалізовано систему на базі фреймворку Llama-index та мови Python для роботи з локальним архівом наукових статей. Проведено порівняльний аналіз базової моделі індексування (Basic Query Engine) та методу контекстних вікон (Sentence Window Query Engine). Оцінку якості проведено за методикою RAG Triad (релевантність контексту, обґрунтованість, релевантність відповіді) з використанням бібліотеки Trulens. Результати підтверджують, що впровадження RAG суттєво підвищує точність відповідей та дозволяє інтегрувати специфічні дані без перенавчання моделі..

Ключові слова - *Large Language Models (LLM), RAG, semantic search, vector embeddings, Llama-index, AI hallucinations, RAG Triad.*

I.ВСТУП

Великі мовні моделі (large Lanquage Model, LLM) - це тип алгоритмів штучного інтелекту (ШІ), який використовує методи глибокого навчання та величезні набори даних для розуміння, узагальнення, створення та прогнозування нового контенту. LLM спеціально створені та ретельно навчені вирішенню завдань з обробки природної мови. Ці моделі проходять навчання на великих обсягах текстових даних, що дозволяє їм генерувати текст, дуже схожий на людську мову. Вони мають здатність уловлювати контекстуальні нюанси та давати відповіді на запитання. В останні роки були створені декілька високоефективних LLM, таких як серія GPT [1], серія LLama [2], Gemini [3] та ін.

Однак широке застосування LLM продемонструвало суттєві проблеми, що виникають при обробці спеціальних запитів, які потребують використання спеціалізованої інформації. Ці проблеми зазвичай виражаються у виникненні т.з. «галюцінацій» систем пошуку, коли згенеровані штучним інтелектом відповіді не є релевантними, і їх використання для прийняття рішень може стати навіть загрозливим для безпеки користувачів.

Для вирішення цієї проблеми Льюісом та ін [4] було запропоновано використання

методології RAG (Retrieval-Augmented Generation) або «доповненої генерації пошуку». Ця концепція, в спрощеному вигляді, полягає в тому, щоб для генерації відповіді на спеціалізований запит користувача, доповнювати «базу знань» моделі додатковою спеціалізованою інформацією. Це посилює точність і достовірність моделей, особливо для наукомістких завдань, і дозволяє постійне оновлювати знання та інтегрувати актуальну особисту, корпоративну або дослідницьку інформацію з загальнодоступною інформацією LLM. RAG синергетично поєднує внутрішні знання LLM з динамічними сховищами зовнішніх баз даних.

В цій роботі представлені основні складові методології RAG та досліджені шляхи її застосування для покращення ефективності використання великих мовних моделей.

II.МОДЕЛІ ТА МЕТОДИ ДОСЛІДЖЕННЯ

Основою сучасних алгоритмів пошуку за допомогою LLM є методи семантичного аналізу текстів. На відміну від пошуку за ключовими словами, який покладається на точну дослівну відповідність, семантичний пошук розпізнає контекстуальні зв’язки між словами через те, що фрагменти тексту перетворюються в т.з. вектори – набори чисел, що дозволяє визначити ступінь подібності між сенсом цих текстів, розраховуючі розмір кута між такими векторами.

Сутність RAG -методології можна описати на наступному прикладі (рис.1).

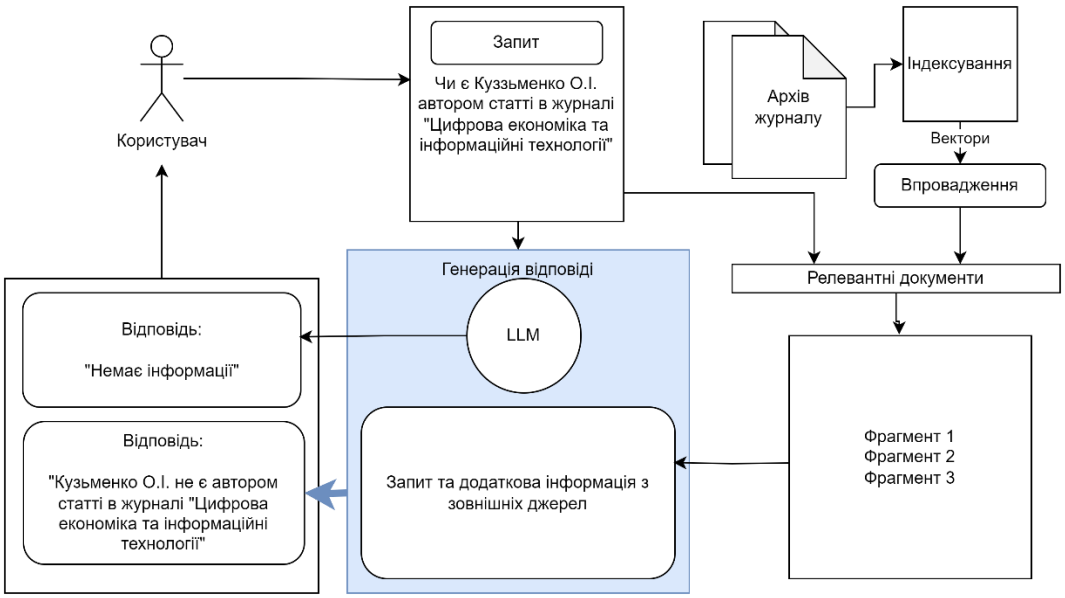

Рис. 1 Ілюстрація методології RAG (побудовано на основі [5])

У цьому сценарії, користувач запитує ChatGPT, наприклад, щодо участі деякого Кузьменка О.І. в якості автора в одному з номерів нового журналу «Цифрова економіка та інформаційні технології». З високою вірогідністю, архив цього журналу ще не був проіндексований LLM, і інформація, яка в ньому зберігається не доступна, якщо використовувати лише базу знань мовної моделі. RAG усуває цю прогалину, отримуючи актуальні витяги з зовнішніх документів. У цьому випадку в систему додається архив журналу. Цей архив, разом з початковим запитом дозволяють згенерувати розширений пошук і отримати релевантну відповідь.

Базова модель RAG використовує 3 основних процеси: індексація, пошук та генерація відповіді.

Індексація (indexing) є важливим початковим кроком який включає завантаження даних з документів різних форматів, як-от PDF, HTML, Word та ін, та перетворення їх на стандартизований простий текст. Далі цей простий текст сегментується на меньші фрагменти або чанки (chunks), які згодом перетворюються на векторні представлення (embedding). Нарешті, створюється індекс для зберігання цих фрагментів тексту та їх векторних представлень як пар ключ-значення, що забезпечує ефективні та масштабовані можливості пошуку.

На етапі пошуку (retrieval) такому ж перетворенню піддається текст запиту користувача. З нього теж отримують векторне представлення, яке порівнюють з векторами доданої зовнішньої інформації. При цьому обчислюються деякі показники подібності, які і дозволяють відібрати з наданої додаткової інформації найбільш релевантні до запиту фрагменти.

Етап генерації відповіді (generation) полягає в тому, що поставлений запит і вибрані фрагменти синтезуються в зв’язне підказування, на яке велика мовна модель має сформулювати відповідь. Підхід моделі до відповідей може змінюватися залежно від конкретних критеріїв завдання, дозволяючи їй спиратися на власні параметричні знання або обмежити відповіді на інформацію, що міститься в наданих документах. У випадках триваючих діалогів, будь-яку існуючу розмовну історію можна інтегрувати в підказку, дозволяючи моделі ефективно брати участь у багатоходовій діалоговій взаємодії.

В теперішній час розроблено багато програмних засобів, які дозволяють реалізувати цей підхід. Основною мовою програмування при цьому виступає Python, для якого створено велику кількість бібліотек, що постійно розвиваються та вдосконалюються.

В даному дослідженні був використаний фреймворк Llama-index. Він дозволяє отримувати дані з API, баз даних, PDF-файлів тощо через гнучкі конектори даних. Ці дані індексуються в проміжні представлення, оптимізовані для LLM. Тоді LlamaIndex дозволяє надсилати запити природною мовою та спілкуватися з вашими даними через системи запитів, інтерфейси чату та агенти даних на базі LLM. Таким чином можна отримати доступ до приватних даних і інтерпретувати їх у великих масштабах без перенавчання моделі на цих даних.

Для оцінки якості відповідей, які користувач отримує при використанні мовних моделей, зокрема доповнених за рахунок технології RAG, необхідно використовувати певні метрики. Головна з цих метрик – релевантність - вимірює ступінь, в якій створені моделлю відповіді відносяться до заданих запитань та безпосередньо зв’язані з заданими запитаннями.

Найбільш використовувана методологія оцінки цих метрик розроблена компанією TruEra носить назву «Rag Triad»[6]. Її сутність представлена на рис. 2.

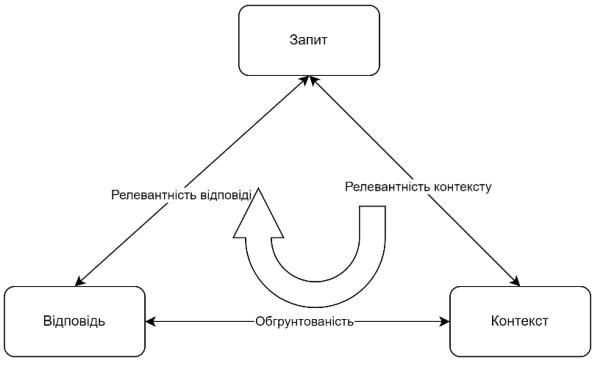

Рис. 2. Rag Triad

Релевантність контексту - це оцінка того, що кожна частина контексту відповідає вхідному запиту. Це критично, оскільки цей контекст використовуватиметься LLM для

формування відповіді, тому будь-яка нерелевантна інформація в контексті може бути вплетена в галюцинацію.

Обгрунтованність (в оригіналі "Groundedness") оцінює, наскільки згенерована відповідь пов’язана або заснована на інформації, яка була зазначена у вхідних даних. Чим більша обгрунтованість, тим більше відповідь відповідає контексту.

І, нарешті, модель має давати корисну відповідь на початкове запитання. Ми можемо перевірити це, оцінивши релевантність кінцевої відповіді введеному користувачем запиту.

Досягнувши задовільних оцінок для цієї тріади, ми можемо з певною долею вірогідності гарантувати правильність висновків, які ми отримуємо за допомогою розробленої системи. Вважається, що система не буде містити галюцинацій у межах своєї бази знань. Іншими словами, якщо векторна база даних містить лише точну інформацію, то відповіді, надані RAG, також точні.

Для реалізації такої кількісної оцінки можуть використовуватися різні Python-бібліотеки. В рамках цього дослідження була використана бібліотека Trulens.

ІІІ.РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

В ході дослідження була побудована програма на мові Python, яка дозволяє підключитися до ChatGPT 3.5 через API.

Далі, за допомогою функцій Llama-index було підключено локальне джерело даних, яке містило наукові статті з журналу «Цифрова економіка та інформаційні технології» за 2022 рік в форматі PDF.

Наступним кроком був опис функції, які створюють індекси – перетворення локальних даних у їх векторне представлення. Перша з цих функцій розраховує базовий індекс на основі завантажених документів (Basic Query Engine), а друга функція використовує метод llama-index SentenceWindowNodeParser, який перед індексуванням дозволяє розбити дані на речення, і об’єднати кожне речення з декількома реченнями до та після поточного. Таким чином очікується підвищення якості контексту для пошуку правильної відповіді. Індекс, створений на цьому підході далі представлений як Sentence Window Query Engine.

Далі був описаний фрагмент, що дозволяє підготувати для використання функції зворотного зв’язку, які зможуть оцінити показники релевантності при роботі моделі. Для оцінювання був підготовлений набір даних в форматі csv, з прикладами запитів та правильних відповідей до них, які спираються на дані, що додаються. При цьому, на додачу до трьох показників релевантності, описаних у RAG TRIADE, був розрахований показник, який оцінює відповідність відповідей правильним відповідям з контрольного набору даних.

Після запуску моделі були отримані значення показників релевантності, представлені на рис. 3.

Для двох описаних вище підходів створення індексів представлені результати, які оцінюють вартість запитів до Chat GPT, а також чотири описаних показники ефективності: Groundeness - обґрунтованість, відповідність відповіді контексту, Answer Correctness – відповідність відповідей та правильних відповідей з контрольного прикладу, Answer Relevance – релевантність відповідей та Context Relevance – релевантність контексту.

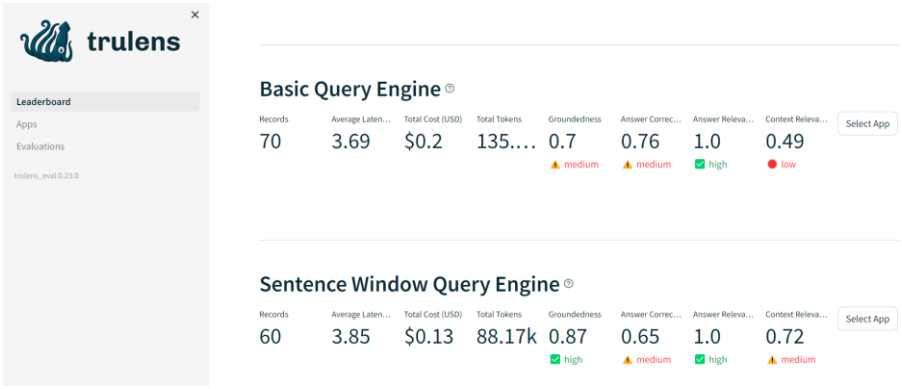

Рис. 3. Результати оцінювання моделі RAG

Аналіз отриманих результатів дозволяє стверджувати, що додавання інформації з зовнішніх джерел дозволяє використовувати LLM без додаткового навчання, при цьому більшість показників навіть базової моделі індексування дозволяють отримати прийнятні результати, але є можливість вдосконалювати якість відповідей системи за рахунок додаткових методів покращення контексту.

IV.ВИСНОВКИ

Retrieval-Augmented Generation є перспективним рішенням для включення знань із зовнішніх баз даних до знань, які використовуються великими мовними моделями. Це посилює точність і достовірність моделей, особливо для наукомістких завдань, і дозволяє постійне оновлювати знання та інтегрувати інформацію з корпоративних та наукових баз даних.

RAG є також підходом, який дозволяє зменшувати вірогідність галюцинацій при генерації відповідей LLM та підвищує якість використання технологій штучного інтелекту в цілому.

При цьому, використання RAG потребує використання спеціальних засобів оцінювання якості впровадження нової інформації до генерації відповідей на запити користувачів, для чого існують розвинуті програмні методи та інструменти.

V.ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

В цій роботі представлені лише базові концепції технології RAG, які відомі як Naive RAG. Для розвитку цих досліджень планується реалізувати та дослідити також такі методології як Advanced та Modular RAG. Перша з них дозволяє вирішити проблеми з індексуванням, з якими стикається Naive RAG, використовуючи такі методи, як ковзне вікно, дрібна сегментація, метадані. і стратегії після пошуку. Modular RAG відрізняється від традиційної структури Naive RAG, забезпечуючи більшу універсальність і гнучкість. Він об’єднує різні методи для покращення функціональних модулів, наприклад, включення модуля для пошуку подібності та застосування підходу для тонкого налаштування в ретривері.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] T. Brown *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1877–1901.

[2] H. Touvron *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[3] Google, "Gemini: A family of highly capable multimodal models," 2023. [Online]. Available: https://goo.gle/GeminiPaper. [Accessed: Mar. 24, 2026].

[4] P. Lewis *et al.*, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 9459–9474.

[5] Y. Gao *et al.*, "Retrieval-augmented generation for large language models: A survey," *arXiv preprint arXiv:2312.10997*, 2023.

[6] TruLens, "The RAG Triad," 2024. [Online]. Available: https://www.trulens.org/trulens_eval/core_concepts_rag_triad/. [Accessed: Mar. 24, 2026].

Отримано 10.11.2024 р.

APPLICATION OF RAG METHODOLOGY FOR EXPANDING THE CAPABILITIES OF LARGE LANGUAGE MODELS

Nataliya Poluektova
Department of Information Technologies
Zaporizhzhia Institute of Economics and Information Technologies
Zaporizhzhia, Ukraine

Abstract - The article explores approaches to addressing the challenges of "hallucinations" and the limitations of static knowledge in Large Language Models (LLMs). The study focuses on the Retrieval-Augmented Generation (RAG) methodology, which enables a synergetic combination of the models' parametric knowledge with dynamic external databases. The key stages of RAG are described: indexing (transforming documents into vector embeddings), semantic retrieval, and response generation based on relevant context. In the practical section, a system was implemented using the Llama-index framework and Python to process a local archive of scientific articles. A comparative analysis was conducted between the Basic Query Engine and the Sentence Window Query Engine. Quality assessment was performed based on the RAG Triad methodology (Context Relevance, Groundedness, Answer Relevance) using the Trulens library. The results confirm that the implementation of RAG significantly improves response accuracy and allows for the integration of specialized data without the need for model fine-tuning.

***Keywords**: Large Language Models (LLM), RAG, semantic search, vector embeddings, Llama-index, AI hallucinations, RAG Triad.*

Recieved 10.11.2024 p.