АРХИТЕКТУРНОЕ РЕШЕНИЕ ПОСТРОЕНИЯ RAG-СИСТЕМЫ АНАЛИЗА ДАННЫХ С ИСПОЛЬЗОВАНИЕМ БИБЛИОТЕКИ LANGCHAIN И ГРАФОВОЙ БАЗЫ ДАННЫХ NEO4J

Батура М. П., Кулевич А. О.
Кафедра информатики, Белорусский государственный университет информатики и радиоэлектороники
Минск, Республика Беларусь
E-mail: bmpbel@bsuir.by, kulevich.01@gmail.com

В статье описывается методология создания RAG-системы анализа данных с использованием графовой базы данных Neo4j и библиотеки LangChain. Рассматриваются этапы построения и применения конвейера для разработки и использования RAG-моделей, ориентированных на обработку и анализ текстовых данных. Приводятся примеры работы системы, демонстрирующие её способность находить ответы на вопросы на основе контекстного поиска и векторного сходства, а также интеграцию с LLM для повышения точности и гибкости анализа.

Введение

Современный объём и сложность данных, с которыми работают организации, быстро растут, делая анализ данных ключевым инструментом для принятия решений и повышения эффективности. Анализ данных позволяет выявлять закономерности и прогнозировать события, однако остаются проблемы, такие как большой объём и неоднородность данных.

Методы искусственного интеллекта решают эти проблемы, ускоряя обработку данных и выявляя скрытые паттерны. Алгоритмы машинного обучения позволяют анализировать как структурированные, так и неструктурированные данные.

Большие языковые модели (LLM), такие как GPT, обрабатывают текстовую информацию и генерируют ответы, что полезно для задач классификации и суммаризации. Технология Retrieval Augmented Generation (RAG) объединяет поиск и генерацию, извлекая релевантные данные и создавая на их основе ответы, повышая точность и скорость обработки информации [1].

I. Конфигурация графовой базы данных и загрузка данных

Для демонстрации анализа данных с использованием RAG-технологий были использованы документы с платформы Wikipedia [2], посвященные графовым базам данных. Для хранения и структурирования информации применялась графовая база данных Neo4j [3], которая отлично справляется с задачами организации и анализа структурированных данных в приложениях RAG. Важно отметить, что Neo4j поддерживает поиск по векторному индексу, что делает её подходящей для приложений RAG, основанных на неструктурированном тексте.

Для работы с векторными индексами Neo4j используется библиотека LangChain [4] — ведущая платформа для создания приложений, основанных на LLM (Large Language Models). LangChain объединяет возможности различных поставщиков LLM, баз данных и других инструментов, что делает её универсальным решением для построения систем анализа данных. Она поддерживает процессы приёма данных, их индексирования, чтения и создания рабочих процессов, что особенно полезно при разработке чат-ботов и систем, отвечающих на вопросы на основе RAG-архитектуры.

В проекте была создана графовая база данных с помощью приложения Neo4j Desktop, а также установлены необходимые плагины — APOC (Awesome Procedures on Cypher) и Graph Data Science Library. Процесс чтения и разбиения статей с сайта Wikipedia, связанных с графовыми базами данных, был реализован с помощью библиотеки WikipediaLoader. Для последующего импорта полученных данных в Neo4j и их индексирования с использованием векторного индекса применялась библиотека Neo4jVector.

В результате в базе данных появились узлы, представляющие статьи о графовых базах данных, готовые для дальнейшего анализа. Несколько полученных узлов и свойства одного из них представлены на рисунке 1.

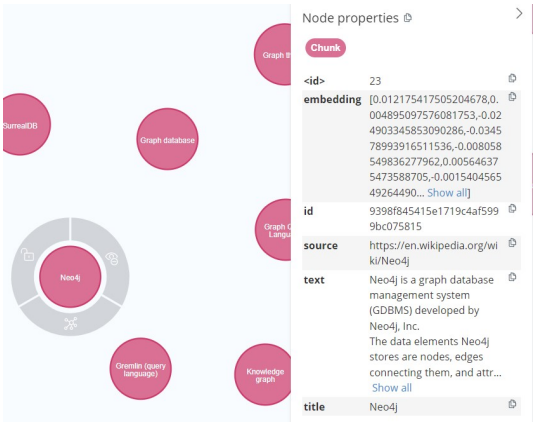

Рис. 1 – Узел и его свойства

II. RAG система анализа данных

На рисунке 2 представлен пример использования RAG-системы для поиска и предоставления ответа на основе векторного сходства. Этот метод измеряет схожесть между векторами в многомерном пространстве, позволяя анализировать и сравнивать объекты на основе их числовых характеристик. В данном примере был получен ответ на запрос: «Что такое графовая база данных Neo4j?».

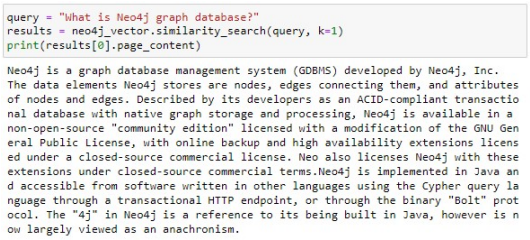

Рис. 2 – Формулирование запроса и поиск векторного сходства

Библиотека LangChain использовала модель OpenAI для встраивания запроса и нахождения наиболее релевантных документов путём сравнения косинусного сходства между вектором вопроса и проиндексированными документами. LangChain также поддерживает рабочий процесс «вопрос-ответ», позволяя создавать системы, которые не только генерируют ответы на основе предоставленного контекста, но и указывают, какие документы использовались в процессе.

Кроме того, библиотека позволяет воссоздать интерфейс в стиле ChatGPT. При добавлении модуля памяти система получает возможность запоминать историю диалогов, что позволяет пользователю задавать уточняющие или последующие вопросы, основываясь на предыдущем контексте.

На рисунке 3 продемонстрирована работа системы: сначала на запрос «Что такое ACID?» был получен ответ, затем на последующий запрос «В каких базах данных он используется?» система также дала точный ответ, распознав контекст и связь с предыдущим вопросом.

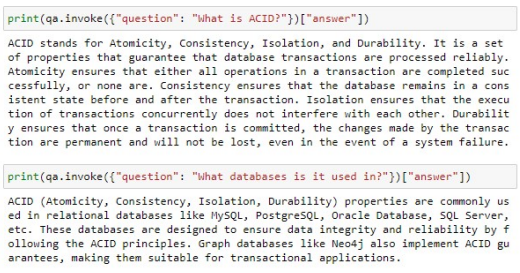

Рис. 3 – Работа системы с модулем памяти

Таким образом, векторный индекс, встроенный в Neo4j, эффективно обрабатывает как структурированные, так и неструктурированные данные, что делает его оптимальным решением для RAG-приложений.

III. Заключение

В данной работе было рассмотрено архитектурное решение для построения RAG-системы анализа данных с использованием графовой базы данных Neo4j и библиотеки LangChain. Описаны основные проблемы, возникающие при анализе данных, и подходы их решения с применением методов искусственного интеллекта и больших языковых моделей.

Разработка RAG-системы необходима для повышения эффективности обработки и анализа больших объемов данных, что становится особенно актуальным в условиях постоянного роста информационных потоков. Использование технологий RAG позволяет извлекать полезную информацию из разнообразных источников, улучшая качество принятия решений в различных областях, таких как бизнес, наука, медицина и другие. Приведенные примеры работы системы показали, как интеграция Neo4j и LangChain способствует ускорению получения ответов на запросы, что является важным для приложений, требующих быстрой реакции на изменения данных.

Таким образом, разработанная методология предоставляет мощные инструменты для создания аналитических систем, способных адаптироваться к различным типам данных и запросов, обеспечивая более глубокое понимание информации и поддержку принятия решений в сложных и динамичных условиях. Это позволяет организациям эффективно реагировать на вызовы современности и извлекать максимальную ценность из имеющихся данных.

Список литературы

1. What is Retrieval-Augmented Generation? [Электронный ресурс]. – Режим доступа: `https://aws.amazon.com/whatis/retrieval-augmented-generation/?nc1=hls`. – Дата доступа: 15.10.24.
2. Wikipedia [Электронный ресурс]. – Режим доступа: `https://ru.wikipedia.org/`. – Дата доступа: 15.10.24.
3. Neo4j [Электронный ресурс]. – Режим доступа: `https://neo4j.com/labs/neosemantics/` Дата доступа: 15.10.24.
4. LangChain [Электронный ресурс]. – Режим доступа: `https://www.langchain.com/`. – Дата доступа: 15.10.24.