СИСТЕМА ИСКУССТВЕННОГО ИНТЕЛЛЕКТА АНАЛИЗА ДАННЫХ ТЕМАТИЧЕСКИХ САЙТОВ С ИСПОЛЬЗОВАНИЕМ RAG ТЕХНОЛОГИЙ, LLM И ГРАФА ЗНАНИЙ

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 подход к созданию системы анализа данных тематических сайтов с использованием технологий RAG, графовой базы данных Neo4j и библиотеки LangGraph. Представлена структура графа знаний и работа агента GraphReader, выполняющего интеллектуальный поиск и генерацию ответов с участием языковой модели. Показана реализация веб-интерфейса, который позволяет загружать данные и получать результаты анализа в режиме диалога.

Введение

Современные информационные системы обрабатывают всё большие объёмы разнородных данных, значительная часть которых представлена в текстовой форме. Для эффективного анализа таких данных требуется не только хранение информации, но и понимание её смысловых связей.

Развитие искусственного интеллекта и появление больших языковых моделей (LLM) сделали возможной автоматическую интерпретацию текстов и извлечение знаний. Однако для повышения точности анализа всё чаще применяются графовые базы данных, отражающие структуру взаимосвязей между понятиями.

Объединение LLM с графами знаний привело к появлению технологий Retrieval-Augmented Generation (RAG) [1], сочетающих поиск и генерацию ответов. Такие системы способны извлекать релевантные сведения, формировать контекст и обеспечивать точный анализ данных тематических сайтов и других источников.

I. Графовая база данных и интеллектуальный GraphReader-агент

Для хранения и анализа информации использовалась графовая база данных Neo4j, обеспечивающая представление знаний в виде взаимосвязанных узлов и отношений. Такой формат позволяет не только хранить данные, но и описывать их смысловые связи, что особенно важно при интеграции с языковыми моделями.

Исходные данные загружались с помощью скрипта на Python, где документы преобразовывались в структурированное графовое представление. Каждый документ разбивался на последовательность фрагментов текста (chunks), которые соединялись между собой отношениями NEXT, отражающими порядок следования частей текста. Далее каждый фрагмент анализировался для выделения атомарных фактов (atomic facts) — минимальных утверждений, содержащих конкретные знания.

На нижнем уровне формировались ключевые элементы (key elements), представляющие наиболее значимые понятия и темы документа. Эти элементы объединяли атомарные факты с более широкими концепциями — такими как исторические личности, географические объекты или научные термины. В результате создавался многоуровневый граф знаний, где документы, факты и ключевые элементы связаны между собой в единую семантическую сеть, общее представление которого продемонстрировано на рисунке 1.

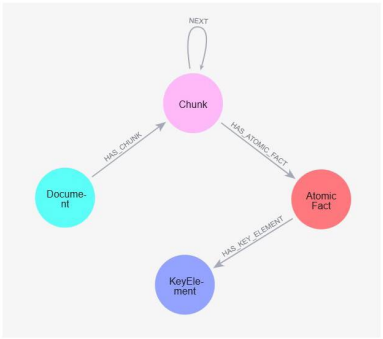

Рис. 1 – Общее представление графовой базы данных

Такое представление данных делает возможным не просто поиск по ключевым словам, а анализ контекста и взаимосвязей между объектами. Именно это используется в работе GraphReader-агента — интеллектуального компонента, способного «читать» граф, интерпретируя его структуру при формировании ответов на запросы [2].

GraphReader-агент взаимодействует с Neo4j через библиотеку LangGraph [3], которая обеспечивает управление цепочками запросов и их обработку с помощью LLM. При поступлении запроса пользовательской системы агент выполняет последовательный обход графа: определяет ключевые элементы, наиболее близкие к запросу, анализирует связанные с ними факты и документы, формирует контекст, отражающий смысловые связи, передаёт этот контекст языковой модели для генерации итогового ответа. Структу-

ра GraphReader-агента представлена на рисунке 2.

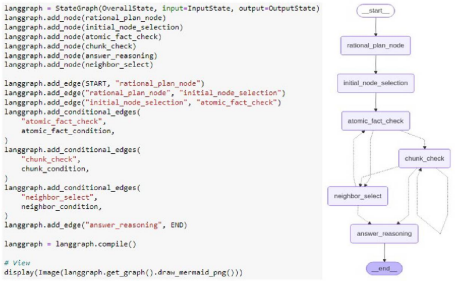

Рис. 2 – Структура GraphReader-агента

Таким образом, GraphReader сочетает семантический поиск с возможностями языковых моделей, обеспечивая точное и интерпретируемое извлечение знаний. В отличие от традиционных подходов, где информация хранится в линейной форме, использование графа позволяет агенту работать с контекстом на уровне понятий и связей, что делает анализ данных более глубоким и гибким.

II. Интеллектуальная система анализа данных на основе RAG-архитектуры

Построенный граф знаний и агент GraphReader стали основой системы, реализующей принципы Retrieval-Augmented Generation (RAG). Эта архитектура объединяет извлечение релевантной информации из базы данных и генерацию ответов при участии языковой модели. В отличие от классического поиска, где пользователь получает набор документов, RAG-подход формирует связный и осмысленный ответ, основанный на контексте, найденном в графе.

Работа системы начинается с пользовательского запроса на естественном языке. Агент GraphReader, используя встроенные механизмы Neo4j и LangGraph, выполняет графовый поиск, выявляя узлы и связи, наиболее близкие к смыслу вопроса. Он анализирует цепочки отношений между документами, фактами и ключевыми элементами, формируя контекстное подграфовое представление — фрагмент знаний, связанный с запросом.

Затем этот контекст передаётся в LLM-модель, которая синтезирует итоговый ответ. Таким образом, модель не «угадывает» ответ из общей информации, а работает с конкретными, извлечёнными из графа фактами. Это повышает точность, снижает вероятность галлюцинаций и делает результат воспроизводимым.

Особенностью системы является её интерактивный интерфейс. Был разработан веб-сайт, позволяющий пользователю напрямую взаимодействовать с базой знаний. Через него можно загружать новые документы, которые автоматически преобразуются в графовую структуру в Neo4j, а также использовать чат-интерфейс для постановки вопросов. На рисунке 3 представлена работа пользовательского интерфейса системы.

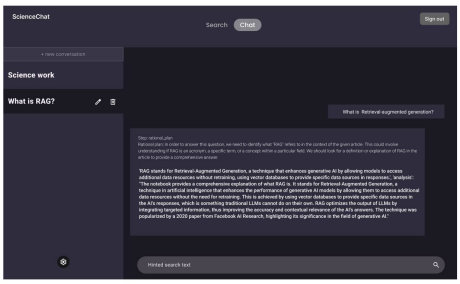

Рис. 3 – Пример работы веб-интерфейса системы анализа данных

Запросы, поступающие через чат, обрабатываются в реальном времени: GraphReader извлекает нужные данные из графа, формирует контекст и передаёт его в языковую модель, которая генерирует ответ в человекопонятной форме. Таким образом, пользователь получает полноценный интеллектуальный инструмент для анализа тематических сайтов, где объединяются точность графовых технологий и гибкость больших языковых моделей

III. Заключение

В работе рассмотрено архитектурное решение системы анализа данных тематических сайтов, основанной на технологиях RAG, LLM и графовой базе данных Neo4j. Использование GraphReader и LangGraph позволило объединить хранение знаний в виде семантического графа и интеллектуальный поиск с генерацией ответов на естественном языке.

Такая архитектура обеспечивает контекстное понимание данных и выявление смысловых связей между понятиями, повышая точность и полноту анализа. Применение RAG-подхода делает систему эффективным инструментом для обработки растущих объёмов информации и ускоряет получение релевантных ответов.

Интеграция графовых технологий и языковых моделей создаёт основу для интеллектуальных аналитических решений, способных глубоко интерпретировать данные и поддерживать принятие решений в условиях сложных информационных потоков.

Список литературы

1. What is Retrieval-Augmented Generation? [Electronic resource]. – Mode of access: `https://aws.amazon.com/whatis/retrieval-augmented-generation/?nc1=hls`. – Date of access: 04.10.25.
2. GraphReader: Building Graph-based Agent to Enhance Long-Context Abilities of Large Language Models [Electronic resource]. – Mode of access: `https://arxiv.org/pdf/2406.14550v1` Date of access: 04.10.25.
3. LangGraph [Электронный ресурс]. – Mode of access: `https://www.langchain.com/langgraph`. – Date of access: 04.10.25.