



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ _____

КАФЕДРА _____ СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ _____

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

НА ТЕМУ:

*Исследование применения больших языковых
моделей и методов RAG в модулях синтаксического
анализа текста*

Студент ИУ5И-35М
(Группа)

(Подпись, дата) Цзя Чанцин
(И.О.Фамилия)

Руководитель

(Подпись, дата) Ю.Е. Гапанюк
(И.О.Фамилия)

Консультант

(Подпись, дата)

(И.О.Фамилия)

20 24 г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ

Заведующий кафедрой ИУ5
(Индекс)

В.И. Терехов
(И.О.Фамилия)

« ____ » _____ 20 24 __ г.

ЗАДАНИЕ

на выполнение научно-исследовательской работы

по теме Исследование применения больших языковых моделей и методов RAG в модулях синтаксического анализа текста

Студент группы ИУ5И-35М

Цзя Чанцин

(Фамилия, имя, отчество)

Направленность НИР (учебная, исследовательская, практическая, производственная, др.)

Исследовательская

Источник тематики (кафедра, предприятие, НИР) КАФЕДРА

График выполнения НИР: 25% к 6 нед., 50% к 5 нед., 75% к 12 нед., 100% к 16 нед.

Техническое задание В данной главе посредством обзора литературы рассматриваются и исследования, связанные с большими языковыми моделями (LLM), технологиями поиска с усиленной генерацией (RAG), а также с преобразованием текста в структуру данных метаграфа. Анализируются существующие достижения в каждой из этих областей и их применение в рамках данного исследования.

Оформление научно-исследовательской работы:

Расчетно-пояснительная записка на _____ листах формата А4.

Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)

Дата выдачи задания « 19 » _____ декабрь _____ 2024 __ г.

Руководитель НИР

Ю.Е. Гапанюк

Студент группы ИУ5И-32М

Цзя Чанцин

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

Введение

С быстрым развитием искусственного интеллекта и технологий обработки естественного языка крупные языковые модели (Large Language Models, LLM) демонстрируют огромный потенциал в понимании языка, генерации текста и системах диалога. В последние годы модели на основе архитектуры Transformer, такие как GPT и BERT, благодаря своим выдающимся способностям к пониманию контекста и генерации текста, стали основным движущим двигателем инноваций в области обработки естественного языка (NLP). Однако полагаться исключительно на языковые модели бывает недостаточно для решения сложных задач, требующих обращения к внешним источникам знаний. Именно поэтому была разработана технология RetrievalAugmented Generation (RAG). Она сочетает в себе мощные возможности генерации LLM и способности извлекать информацию из внешних данных, что позволяет решать сложные задачи анализа текста и вопросов с ответами, обеспечивая более точную и богатую информацией поддержку.

В этом контексте эффективное использование технологий LLM и RAG для анализа текста становится темой, заслуживающей глубокого изучения. Особенно важно это в условиях постоянно растущих объемов данных и их усложнения. Простое линейное представление текста уже не может удовлетворять потребности современных приложений. Преобразование текста в формат метаграфа (metagraph) с иерархической структурой может эффективно поддерживать представление и вывод сложных отношений. Metagraph позволяет не только демонстрировать сущности и их взаимосвязи в тексте, но и обладает высокой масштабируемостью и гибкостью, что делает его полезным для анализа больших данных и построения онтологий. Это открывает новые возможности для таких областей, как анализ больших данных и автоматизированное управление знаниями.

Настоящая работа посвящена исследованию того, как использовать технологии LLM и RAG для анализа текста и преобразования его в метаграф. Изучая процесс анализа и преобразования текста в системе RAG, мы стремимся построить эффективный модуль анализа текста, который автоматически преобразует неструктурированные данные в структурированные метаграфы. Это исследование не только повысит наглядность и управляемость текстовой информации, но и предоставит новые решения для создания интеллектуальных систем управления данными и построения онтологий.

Основной целью данного исследования является использование технологий LLM и RAG для извлечения ценной информации из сложных текстов на естественном языке и преобразования их в структурированное представление данных в виде метаграфов. Этот подход имеет важное теоретическое и практическое значение для решения проблем, связанных с анализом текста и построением онтологий.

Обзор источников

В данной главе посредством обзора литературы рассматриваются исследования, связанные с большими языковыми моделями (LLM), технологиями поиска с усиленной генерацией (RAG), а также с

преобразованием текста в структуру данных метаграфа. Анализируются существующие достижения в каждой из этих областей и их применение в рамках данного исследования.

Развитие больших языковых моделей (LLM)

Большие языковые модели (LLM) достигли значительного прогресса в области обработки естественного языка. Развитие LLM началось с моделей на основе глубокого обучения, и наиболее знаковым прорывом стало введение архитектуры Transformer. В 2017 году Васвани и коллеги в своей работе «Attention is All You Need» представили архитектуру Transformer, которая с помощью механизма самовнимания решила проблему низкой эффективности традиционных последовательных моделей (таких как RNN и LSTM) при обработке длинных текстов. Благодаря этой особенности Transformer превосходно справляется с пониманием контекста и обработкой дальних зависимостей, что способствовало появлению моделей GPT, BERT, LLaMA и других.

Модели серии GPT: Предложенные OpenAI модели серии GPT, особенно последняя версия GPT4, демонстрируют чрезвычайно высокие способности в генерации и понимании языка. GPT-модели используют методы несупервизорного предобучения и авторегрессии, способны генерировать высококачественные длинные тексты и проявляют исключительную адаптивность в многозадачных сценариях. Эти возможности сделали GPT-модели широко применимыми в задачах парсинга текста, генерации диалогов и создания аннотаций [Vaswani et al., 2017].

Модель BERT: В отличие от GPT, BERT (Bidirectional Encoder Representations from Transformers) — двунаправленная модель Transformer, предложенная Google в 2018 году. Она использует метод маскированного языкового моделирования (MLM) для предобучения, что позволяет ей понимать текст, учитывая контекст с обеих сторон. Это позволяет BERT показывать выдающиеся результаты во многих задачах понимания текста, таких как системы вопросов-ответов, классификация текстов и распознавание именованных сущностей [Devlin et al., 2018].

Модели серии LLaMA: LLaMA (Large Language Model Meta AI) — серия легковесных больших языковых моделей, представленных Meta в 2023 году. По сравнению с моделями GPT и BERT, цель серии LLaMA — предоставить модели, способные поддерживать высокую производительность при меньшем количестве параметров. LLaMA использует архитектуру Transformer и предобучается на большом объеме общедоступных данных, что делает её подходящей для различных задач обработки естественного языка. Модели LLaMA, сохраняя низкие требования к вычислительным ресурсам, способны достигать производительности, сопоставимой или даже превосходящей крупные языковые модели в задачах генерации текста, машинного перевода и ответа на вопросы. Значительным преимуществом LLaMA является её открытость и эффективность, позволяющая исследователям и разработчикам проводить обучение и тонкую настройку при относительно небольших вычислительных затратах. LLaMA демонстрирует, как в условиях ограниченных ресурсов можно сбалансировать размер модели и её

производительность, достигая при этом отличных результатов в различных задачах [Touvron et al., 2023] .

Общей чертой этих больших языковых моделей является их мощная способность к обработке контекста и универсальность. Они могут служить базовыми моделями в модулях парсинга текста для извлечения полезной информации из сложных текстов.

Технология поиска с усиленной генерацией (RAG)

Технология RAG сочетает в себе генеративные способности больших языковых моделей и возможности поиска внешних источников данных для повышения релевантности и точности выполнения генеративных задач. Традиционные генеративные модели (например, GPT), основывающиеся исключительно на языковом моделировании, ограничены объемом данных, на которых они обучены, и не могут динамически использовать последние или специализированные данные. RAG вводит шаг поиска в процессе генерации, извлекая релевантную информацию из внешних баз знаний или баз данных и включая её в качестве дополнительного входа, что позволяет улучшить способность отвечать на вопросы и повышает точность генерируемого контента.

Оригинальная статья о модели RAG : Модель RAG, предложенная Льюисом и его коллегами, сочетает плотный поиск (dense retrieval) и генеративные языковые модели. Путем поиска внешних знаний и их комбинирования с языковой моделью она решает недостатки чисто генеративных моделей при выполнении задач с открытыми вопросами. Два ключевых компонента RAG — это поисковик и генератор: первый находит релевантную информацию в больших наборах данных или базах знаний с помощью векторного поиска, а второй генерирует итоговый ответ на основе полученной информации [Lewis et al., 2020] .

Примеры применения : RAG демонстрирует отличные результаты в таких задачах, как системы вопросоответ, поиск по документам, диалоговые системы. Например, системы вопросоответ с использованием технологии RAG могут предоставлять более точные и обогащённые ответы, осуществляя запросы к внешним базам данных, а не полагаясь только на память модели. Эта технология особенно полезна для обработки задач, требующих актуальных или специализированных данных, например, систем поддержки принятия решений в медицинской диагностике [Izacard & Grave, 2021] .

Преобразование текста в метаграф

Преобразование неструктурированного текста в структурированный формат данных метаграфа является одной из ключевых целей данного исследования. Метаграф представляет собой расширенную форму традиционной графовой структуры, которая может отображать сложные отношения нескольких измерений, такие как иерархия между сущностями, семантические связи и контекстные ассоциации. Он имеет широкий потенциал применения в графах знаний, концептуальных картах и реляционных базах данных.

Модель метаграфа :

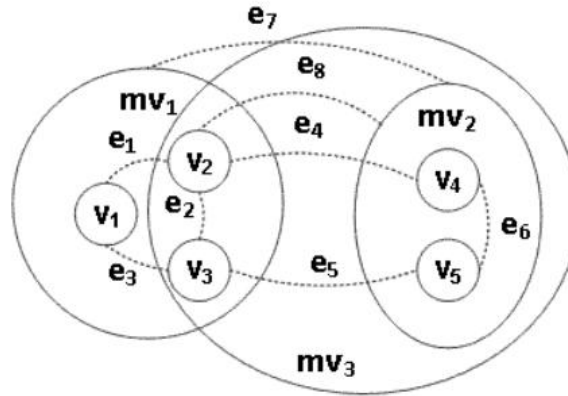


Fig. 2. The example of metagraph representation.

Модель метаграфа — это сложная графовая модель: $(MG = (V, MV, E))$, где (MG) — метаграф; (V) — множество вершин метаграфа; (MV) — множество метавершин метаграфа; (E) — множество рёбер метаграфа. Пример представления метаграфа представлен на Рис. 2. В примере содержатся три метаверши: $(mv1)$, $(mv2)$ и $(mv3)$. Метаверша $(mv1)$ содержит вершины $(v1)$, $(v2)$, $(v3)$ и соединяющие их рёбра $(e1)$, $(e2)$, $(e3)$. Метаверша $(mv2)$ содержит вершины $(v4)$, $(v5)$ и соединяющее их ребро $(e6)$. Рёбра $(e4)$, $(e5)$, соединяющие вершины $(v2)$ и $(v4)$, а также $(v3)$ и $(v5)$, находятся в разных метавершинах $(mv1)$ и $(mv2)$. Ребро $(e7)$ является примером ребра, соединяющего метаверши $(mv1)$ и $(mv2)$. Ребро $(e8)$ является примером ребра, соединяющего вершину $(v2)$ и метавершину $(mv2)$. Метаверша $(mv3)$ содержит метавершину $(mv2)$, вершины $(v2)$, $(v3)$ и ребро $(e2)$ из метавершины $(mv1)$, а также рёбра $(e4)$, $(e5)$, $(e8)$, показывающие эмерджентную природу структуры метаграфа.

Построение графа знаний :

Граф знаний (Knowledge Graph, KG) является типичной графовой структурой, используемой для хранения и организации знаний о мире. В графах знаний Google и других крупных графах знаний преобразование текста в граф знаний основывается на извлечении сущностей и отношений из текста и их отображении в графовой структуре. С помощью больших языковых моделей и технологий RAG ключевые сущности, атрибуты и отношения из текста могут быть автоматически извлечены и сохранены в метаграфе, что предоставляет основу для управления и вывода сложных данных [Singhal, 2012] .

Графовые базы данных и Neo4j :

На практике графовые базы данных (например, Neo4j) стали важным инструментом для хранения и обработки графовых данных. Neo4j поддерживает запросы, анализ и визуализацию больших объемов графовых данных, что предоставляет технологическую основу для хранения и извлечения метаграфа после его преобразования из текста. В литературе применение графовых баз данных широко используется для обработки структурированных и полуструктурированных данных и демонстрирует отличные результаты в области управления знаниями, анализа социальных сетей и других сферах [Angles & Gutierrez, 2008] .

Другие ключевые технологии

Помимо больших языковых моделей (LLM) и технологий RAG, преобразование текста в метаграф также связано с другими важными технологиями, такими как проектирование запросов (Prompt engineering) и технологии поиска в графах.

Проектирование запросов: Проектирование запросов играет ключевую роль в генеративных моделях, поскольку оно напрямую влияет на качество результатов, получаемых от LLM. С помощью тщательно разработанных запросов модель может более точно понять потребности пользователя, что позволяет генерировать высококачественные тексты или структурированные данные. Исследования в области проектирования запросов предоставляют теоретическую поддержку для того, как строить эффективные подсказки, а также повышают качество выходных данных модели путем постоянной оптимизации подсказок [Brown et al., 2020].

Поиск в графах: В контексте применения метаграфа важным направлением исследований является эффективный поиск сущностей и отношений в графе. В существующей литературе технологии поиска в графах уже широко применяются в семантическом поиске, анализе социальных сетей и выводах графов знаний. Графовые базы данных (такие как Neo4j) и алгоритмы поиска (например, обход графа, поиск на основе векторов) могут предоставить эффективную поддержку запросов [He et al., 2021].

Из обзора соответствующей литературы видно, что сочетание технологий LLM, RAG и графовых баз данных предоставляет прочную теоретическую и техническую основу для реализации преобразования текста в метаграф. Будущая работа будет основана на этих существующих результатах, с целью дальнейшей оптимизации процесса генерации текста в метаграф и повышения точности и эффективности системы.

Первоначальные предложения по реализации темы исследования

Целью данного исследования является преобразование текстов на естественном языке в структурированный формат данных метаграфа с использованием больших языковых моделей (LLM) и технологий, основанных на увеличении поиска (RAG). Этот процесс включает в себя предварительную обработку текста, идентификацию сущностей и отношений, а также окончательное графическое хранение и организацию данных. В следующем разделе подробно описывается, как реализовать этот процесс преобразования с помощью технологий LLM и RAG.

Проектирование технической архитектуры

Для осуществления преобразования текста в метаграф, в данной статье предлагается следующая системная архитектура, которая включает предварительную обработку текстовых данных, интеграцию LLM и RAG, а также хранение в графовой базе данных.

1. Предварительная обработка текста и разбивка на блоки (Chunking)

Во-первых, входной текст на естественном языке будет предварительно обработан с использованием модели RAG, которая применит технологию

разбивки на блоки (chunking) для разделения текста на несколько фиксированных или семантически обоснованных сегментов. Этот шаг способствует повышению эффективности поиска и анализа, особенно в обработке длинных текстов.

В процессе предварительной обработки, помимо разбивки, также осуществляется удаление несущественной информации, очистка шумовых данных, стандартизация формата и другие действия, что закладывает основу для дальнейшего анализа.

2. Анализ текста с помощью LLM и дополнение

В рамках системы RAG LLM будет глубоко анализировать текст, идентифицируя в нем сущности, отношения и намерения. Конкретные действия включают в себя дополнение неполных текстовых сегментов с помощью LLM, что обеспечивает целостность данных, а также использование контекстных возможностей LLM для генерации семантически точных результатов.

LLM отвечает за извлечение ключевых сущностей (таких как лица, организации, места и т.д.) и отношений между ними из сегментированного текста. Кроме того, она может использовать технологию дополнения для создания необходимой контекстной информации, что гарантирует, что даже фрагментированный текст будет иметь полное семантическое значение.

3. Конструирование метаграфа

Извлеченные из текста сущности и отношения будут отображаться в структуре метаграфа. Метаграф — это графовый формат данных, который позволяет представлять различные типы отношений и их атрибуты в виде узлов и ребер графа. Эта структура особенно подходит для обработки сложных графов знаний и семантических сетей, что помогает организовать и управлять большими объемами неструктурированных данных.

При построении метаграфа графовая база данных (такая как Neo4j) будет хранить эти сущности и отношения, одновременно поддерживая дальнейшие запросы и анализ.

Процесс системы

1. Ввод: Пользователь предоставляет текст на естественном языке в качестве входных данных, которые могут касаться любой области, например, новостей, юридических документов, технических отчетов и т.д.

2. Анализ: Система сначала использует модель RAG для поиска соответствующей информации из внешних баз данных или знаний, чтобы восполнить недостаток информации в тексте. В то же время LLM выполняет семантический анализ входного текста, извлекая ключевые сущности, отношения и намерения. Модуль поиска RAG сочетается с генеративной способностью LLM, что гарантирует, что система не только извлекает информацию из входного текста, но и может дополнять недостающую информацию с помощью внешних источников данных.

3. Генерация: Извлеченные сущности и отношения будут отображены на узлах и ребрах метаграфа, создавая структурированный графовый формат данных. Этот процесс будет осуществляться графовой базой данных, что обеспечит эффективное и масштабируемое хранение и управление данными.

Предварительные предложения по реализации

С помощью такой проектной архитектуры данное исследование не только сможет извлечь структурированную информацию из текстов на естественном языке, но и организовать эту информацию в виде графа для дальнейшего поиска, анализа и вывода. Этот процесс преобразования текста в метаграф предоставляет эффективные решения для таких приложений, как построение графов знаний и семантических сетей.

Необходимые задачи

Для выполнения темы исследования «Исследование применения больших языковых моделей и технологии RAG в модулях анализа текста» необходимо поэтапно разработать, протестировать и внедрить различные ключевые модули. Весь процесс можно разделить на следующие ключевые задачи:

1. Литературный обзор

Глубокое исследование анализа текста: Необходимо получить полное представление о исследованиях и применениях, связанных с анализом текста, особенно о методах анализа, основанных на LLM и RAG. Основное внимание следует уделить передовым технологиям семантического анализа, извлечения знаний и понимания естественного языка.

Исследование графовых баз данных и метаграфов: Глубоко изучить структуру и применение графовых баз данных, особенно как в этих базах данных хранить и управлять метаграфами. Включает изучение графовых баз данных (таких как Neo4j, ArangoDB) и способов эффективного хранения сущностей и отношений.

Последние достижения в смежных областях: Ознакомиться с последними разработками в области LLM и RAG, такими как новые модели, оптимизированные фреймворки и успешные примеры применения, и использовать это для теоретической основы проектирования системы.

2. Проектирование и разработка системы

Выбор и обучение подходящей модели LLM: В зависимости от требований темы выбрать подходящую большую языковую модель (например, GPT, BERT, LLaMA) и дообучить ее для конкретного сценария применения. Можно рассмотреть возможность применения переноса обучения на основе уже существующих предобученных моделей для повышения качества анализа текста.

Интеграция RAGрамки: Выбрать подходящую RAGрамку (например, FAISS, Haystack, LangChain и др.) для реализации функций поиска и генерации. RAG может сочетать внешние источники данных для дополнения и оптимизации результатов, сгенерированных LLM, что делает его ключевым компонентом системы.

Разработка модуля предварительной обработки текста: Спроектировать и разработать модуль предварительной обработки текста для очистки данных, разбивки на блоки и других операций. Обеспечить разумное разбиение и

обработку входного текста, чтобы гарантировать точность последующего процесса анализа.

Анализ текста и хранение метаграфа: Разработать основной модуль для преобразования текста в сущности и отношения, а также для хранения этой информации в графовой базе данных. Обеспечить точное отображение извлеченных сущностей и отношений в структуре метаграфа и поддержку последующих запросов и анализа.

3. Источники данных и аннотация

Получение разнообразных источников данных: Получить богатые обучающие и тестовые данные из различных текстовых источников (например, открытые базы данных, API, внутренние данные компании), чтобы гарантировать широкое покрытие различных областей и сценариев применения.

Аннотация и обработка данных: Создать аннотированные наборы данных для обучения и оценки системы, особенно в задачах идентификации сущностей и извлечения отношений, необходимо обеспечить качество и согласованность аннотированных данных.

Разнообразие и обобщающая способность: Обеспечить сильную обобщающую способность модели путем использования разнообразных источников данных и различных типов текстов, чтобы она могла адаптироваться к требованиям анализа текста в различных сценариях применения.

4. Оптимизация алгоритмов

Оптимизация модуля поиска RAG: Провести оптимизацию алгоритмов для части поиска RAG, чтобы повысить точность и эффективность системы в области информационного поиска. Исследовать способы оптимизации стратегий поиска (например, вычисление семантической схожести, стратегии запросов и др.), чтобы улучшить релевантность сгенерированного текста.

Увеличение точности анализа: Дальнейшая доработка и оптимизация LLM для обеспечения высокой точности в задачах идентификации сущностей и извлечения отношений. Оптимизация параметров модели, улучшение процесса обучения и другие методы для повышения общей эффективности генерации системы.

5. Тестирование и оценка

Проектирование экспериментов и тестирование: Провести обширное тестирование системы через множество экспериментов, чтобы убедиться, что модули анализа, поиска и генерации метаграфа функционируют должным образом. Разработать различные тестовые сценарии, охватывающие разнообразные текстовые ситуации.

Проектирование оценочных критериев: Разработать адекватные оценочные критерии для оценки производительности системы. Возможные критерии включают:

Точность анализа: точность идентификации сущностей и отношений.

Релевантность результатов поиска: соответствие извлеченных внешних данных содержанию текста.

Качество генерации метаграфа: эффективность сгенерированной структуры метаграфа с точки зрения точности и полноты.

Оптимизация производительности модели: На основе результатов тестирования провести дальнейшую настройку и оптимизацию модели, чтобы улучшить общую производительность и скорость отклика системы.

С выполнением вышеуказанных задач система постепенно сформирует полный процесс преобразования от анализа текста к генерации метаграфа, что заложит прочную основу для последующего построения графов знаний и семантического поиска.

Ожидаемые результаты

В ходе данного исследования планируется достичь следующих основных результатов:

Результат 1: Система преобразования текста в метаграф

Основным результатом исследования станет система, способная преобразовывать текст на естественном языке в структуру данных метаграфа. Эта система будет обладать следующими функциями:

Извлечение сущностей и отношений: Способна автоматически распознавать ключевые сущности, события и их взаимосвязи в вводимом тексте на естественном языке.

Генерация метаграфа: На основе извлеченных сущностей и отношений система будет генерировать данные, соответствующие структуре метаграфа, поддерживающие многоуровневую связь знаний и моделирование сложных отношений.

Интеграция с графовой базой данных: Извлеченные метаграфные данные будут храниться в графовой базе данных, что упростит последующий поиск, запросы и управление знаниями.

Результат 2: Оценка эффективности технологий LLM и RAG

Посредством экспериментов и тестирования исследование проведет всестороннюю оценку эффективности технологий LLM и RAG в процессе анализа и генерации текста. Ожидаемые результаты оценки включают:

Анализ производительности модели: Количественный анализ работы больших языковых моделей (таких как GPT, BERT и т.д.) в задачах анализа текста, оценка их применимости в различных сценариях.

Вклад технологии RAG: Анализ роли технологии RAG в повышении эффективности поиска информации и генерации, оценка ее влияния на точность анализа текста и эффективность генерации.

Сводка преимуществ и недостатков модели: На основе результатов экспериментов составление сводки преимуществ и недостатков технологии LLM+RAG при анализе сложных текстов, что обеспечит направления для будущих исследований.

Результат 3: Связанный код и техническая документация

После завершения исследования будет подготовлен полный набор технических реализаций и документации, включая:

Исходный код: Включает детали реализации системы, код для обучения и дообучения моделей, а также полный процесс анализа текста и генерации метаграфа.

Техническая документация: Подробно фиксирует проектирование архитектуры, процесс реализации модулей, а также возникшие проблемы и их решения, что будет служить справочным материалом для последующей разработки и улучшения.

Предложения по улучшению и перспективы: На основе открытий в процессе исследования будут предложены рекомендации по оптимизации моделей, алгоритмов и производительности системы, а также обсуждены перспективы применения в более крупных наборах данных и сложных задачах.

С помощью этих ожидаемых результатов данное исследование предоставит практическое решение для анализа текста и построения графов знаний на основе технологий LLM+RAG, а также создаст технологическую основу для дальнейших исследований в этой области.

Ссылки:

1. Vaswani, A., Shard, N., Sutskever, I., et al. Attention is All You Need. 2017.
2. Devlin, J., Chang, M.-W., Lee, K., et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2018.
3. Touvron, P., Roli, A., et al. LLaMA: Open and Efficient Foundation Language Models. 2023.
4. Lewis, M., O'Connor, B., et al. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. 2020.
5. Izacard, M., Grave, A. Leveraging Pre-trained Checkpoints for Sequence Generation Tasks. 2021.
6. Singhal, A. "Introduction to the Theory of Logic." 2012.
7. Angles, R., Gutierrez, C. "Survey of Graph Database Models." ACM Computing Surveys 40, no. 1 (2008): 1-39.