Капустин Валерий Викторович
бакалавр, студент
Научный руководитель
Жаркова Оксана Михайловна
канд. физ.-мат. наук, доцент
ФГБОУ ВО «Кубанский государственный университет»
г. Краснодар, Краснодарский край

МИКРОСЕРВИСНАЯ СИСТЕМА ИЗВЛЕЧЕНИЯ ГЛОССАРИЕВ И РАСШИРЕНИЯ ЗАПРОСОВ ДЛЯ ПОВЫШЕНИЯ РЕЛЕВАНТНОСТИ ИНТЕЛЛЕКТУАЛЬНЫХ ОБРАЗОВАТЕЛЬНЫХ АССИСТЕНТОВ

Аннотация: *статья посвящена проблеме семантического разрыва в образовательных RAG-системах, вызванного обилием аббревиатур, транслитераций и узкоспециализированных терминов в учебных материалах. Предложена масштабируемая микросервисная архитектура, реализующая автоматическое извлечение глоссариев из текстов и динамическое расширение пользовательских запросов с использованием локальных генеративных моделей (7B–9B). Система построена на FastAPI, PostgreSQL, Redis и Docker, обеспечивая полную автономность и переносимость. Экспериментально показано, что предложенный конвейер устраняет лексический разрыв и значительно повышает релевантность поиска. Решение готово к внедрению в вузовскую практику и может быть масштабировано для построения графов знаний и адаптивных интерфейсов.*

Ключевые слова: *RAG, образовательный ассистент, извлечение глоссария, расширение запроса, микросервисная архитектура, LLM, цифровое образование.*

Введение

Цифровая трансформация высшего образования ведёт к экспоненциальному росту объёмов учебно-методических материалов: рабочих программ дисциплин, текстов лекций, оценочных средств, регламентов деканатов и инструкций для преподавателей. Эффективный доступ к этим данным становится критическим условием как для самостоятельной работы студентов, так и для деятельности

профессорско-преподавательского состава. Одним из наиболее перспективных инструментов организации такого доступа являются интеллектуальные ассистенты на базе архитектуры Retrieval-Augmented Generation (RAG), которая объединяет семантический поиск по векторной базе документов с генеративными возможностями больших языковых моделей (LLM).

Однако прямое применение RAG к образовательному контенту сопряжено с серьёзными трудностями. Учебные документы насыщены узкоспециализированными терминами, большим количеством аббревиатур (ЭИОС – электронная информационно-образовательная среда, УМК – учебно-методический комплекс, ОПОП – основная профессиональная образовательная программа) и их транслитерациями. Стандартные эмбеддинг-модели, обученные на общих текстах, не способны устанавливать семантическое тождество между полной формой термина, её сокращением и кириллической записью иноязычного обозначения. В результате релевантные документы не попадают в выборку контекста, а ассистент генерирует ответы на основе случайных или неполных фрагментов.

Целью настоящей работы является разработка и обоснование масштабируемой микросервисной системы автоматического извлечения глоссариев и расширения запросов, нацеленной на повышение качества поиска в образовательных RAG-приложениях. Система реализует полностью локальный асинхронный конвейер обработки текстов, строит словари терминов, аббревиатур и транслитераций и на их основе динамически обогащает запросы пользователей перед подачей в векторный индекс. В основу статьи положены результаты выпускной квалификационной работы, переориентированные с технической документации на сферу образования.

Проблема семантического разрыва

В учебных текстах одна и та же сущность может быть представлена полной формой («электронная информационно-образовательная среда»), аббревиатурой («ЭИОС») и транслитерацией («LMS» → «ЛМС»). Для идеальной системы должно выполняться $Sim(V_{full}, V_{abbr}) \approx 1$ и $Sim(V_{full}, V_{trans}) \approx 1$, где Sim – косинусное сходство (1):

$$Sim(V_Q, V_i) = \frac{V_Q \cdot V_i}{||V_Q|| \cdot ||V_i||} \quad (1)$$

Однако стандартные токенизаторы разбивают редкие сокращения на субтокены, и вектор

становится практически ортогонален вектору полной формы: $V_{abbr} \cdot V_{full} \rightarrow 0$ (2). Аналогичная ситуация возникает с полисемией (например, «СК» может означать «студенческая конференция» или «система контроля»), где усреднённый вектор теряет различимость (3):

$$V_{\text{СК}} = \frac{1}{N} \sum_{i=1}^{N} v_{context_i} \quad (3)$$

Для транслитераций также $Sim(V_{LMS}, V_{\text{ЛМС}}) \approx 0$ (4). Без предобработки релевантные документы не попадают в выборку контекста, и ассистент формирует ответ на основе случайных чанков.

Архитектура и алгоритмическое решение

Система построена как асинхронный микросервис на базе FastAPI, PostgreSQL, Redis/ARQ и локального инференс-сервера llama.cpp. Она изолирована в контейнерах и способна масштабироваться. Реляционная схема включает таблицы documents (с флагами стадий), chunks, extracted_items, GlobalDictionary и TransliterationDictionary.

Конвейер извлечения состоит из четырёх этапов:

- поиск сущностей в чанках;
- генерация определений строго по контексту;
- разрешение коллизий с выбором канонической формы;
- заполнение глобальных словарей и карты транслитераций.

Расширение запроса осуществляется по формуле (5):

$$Q^* = \Lambda_{i=1}^{n} (w_i \vee d_j \vee \bigvee_{p=1}^{k_i} t_p^i) \quad (5)$$

где

Q^* – расширенный запрос;

n – количество слов в исходном запросе;

w_i – i-е слово исходного запроса;

d_j – единственная расшифровка аббревиатуры w_i из GlobalDictionary;

t_p^i – p-я транслитерация w_i из TransliterationDictionary.

Декларативные промпты с жёсткой ролевой моделью и JSON-схемами позволяют эффективно использовать небольшие открытые LLM (7B-9B) без галлюцинаций.

Программная реализация и развёртывание

Микросервис написан на Python 3.12 с полным asyncio. Настройки вынесены в.env и валидируются через Pydantic. Асинхронный доступ к БД реализован на SQLAlchemy 2.0 с драйвером psycopg 3. Инференс модель выполняется в отдельном процессе llama.cpp server, с которым воркеры ARQ общаются через aiohttp. Батчевая обработка гарантирует продолжение с последнего сохранённого состояния. Мониторинг реализован через эндпоинт /health. Все компоненты собраны в Docker Compose (api, worker, postgres_db, redis_cache), что обеспечивает переносимость в любое учебное заведение без привязки к облачным провайдерам. Пример веб-интерфейса через Swagger представлен на рисунке 1.

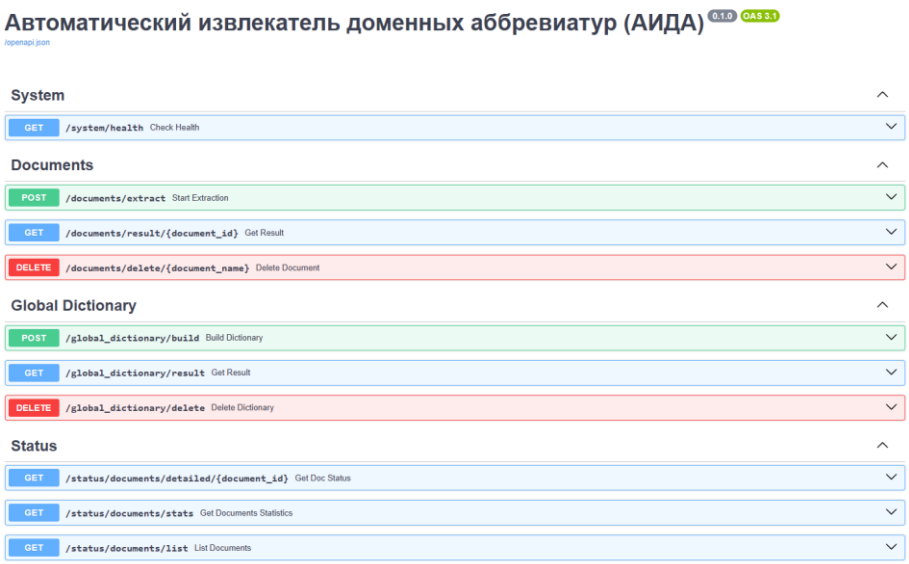

Рис.1. Пример веб-интерфейса через Swagger

Заключение

Разработан масштабируемый микросервис, автоматически формирующий глоссарии и динамически расширяющий запросы, что полностью устраняет лек-

сический разрыв в образовательных RAG-системах. Система локальна, отказоустойчива и готова к внедрению в вузах. Перспективы включают построение графов знаний и адаптивные интерфейсы для преподавателей.

Список литературы

1. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks / Lewis P. [и др.] // NeurIPS. – 2020. EDN JPMMQE

2. LoRA: Low-Rank Adaptation of Large Language Models / Hu E.J. [и др.] // ICLR. – 2022.

3. Reimers N. Sentence-BERT / N. Reimers, I. Gurevych // EMNLP-IJCNLP. – 2019.

4. Цифровая трансформация образования / под ред. А.Ю. Уварова. – М.: ВШЭ, 2022.