УДК 004.8

РАЗРАБОТКА НЕЙРОСЕТЕВОГО КОНСУЛЬТАНТА ДЛЯ СТУДЕНТОВ НА БАЗЕ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ

Т.А. Васяева
Декан факультета информационных систем и технологий, доцент кафедры автоматизированных систем управления ДонНТУ, кандидат технических наук, доцент
tanechka.vasyaeva@yandex.ru

М.В. Душа
студент IV-го курса бакалавриата кафедры автоматизированных систем управления ДонНТУ
dushamihail@yandex.ru

Т.А. Васяева
Окончила Донецкий национальный технический университет. Область научных интересов связана с разработкой методов и алгоритмов искусственного интеллекта, организацией учебного и научно-исследовательского процессов в техническом университете.

М.В. Душа
Студент Донецкого национального технического университета. Область научных интересов связана с разработкой систем машинного обучения

Аннотация. Статья посвящена разработке нейросетевого консультанта информационной поддержки студентов на базе технологии Retrieval-Augmented Generation (RAG) с использованием российских больших языковых моделей. Проведен сравнительный анализ эмбеддинг моделей, методов индексации документов инструментами фреймворка LlamaIndex, генеративных моделей GigaChat-2-Lite, GigaChat-2-Pro и DeepSeek-V3.2. Продемонстрирован оптимальный набор средств для построения RAG системы на основе внутривузовских нормативных актов: ru-en-RoSBERTa, векторное индексирование, GigaChat

Ключевые слова: большие языковые модели, RAG, GigaChat, LlamaIndex, эмбеддинг-модели, векторная индексация, LLM as a Judge, нейросетевой консультант, информационная поддержка студентов.

Введение. Деятельность современного студента состоит из множества зачастую разрозненных направлений, включая учебную, научную, административную и внеучебную сферы. В связи с этим критически важно получать актуальную информацию своевременно. Однако структура информационной поддержки высшего учебного заведения не может в полной мере закрыть эту потребность. Работа в строго ограниченное время и необходимость в физическом присутствии сотрудника деканата для получения консультации способствуют замедлению учебного процесса и повышению стресса студентов. Автоматизированные цифровые консультанты могут стать практичным решением, благодаря доступу к информации 24/7. Применение генеративных нейронных сетей позволит получать простые для понимания ответы, а формат чат-бота – удобный и интуитивно понятный интерфейс взаимодействия с консультантом.

В связи с нынешними санкционными ограничениями использование популярных западных нейронный сетей ограничено. Поиск качественных отечественных больших языковых моделей позволит разрабатывать современные продукты на базе искусственного интеллекта поддерживая политику цифрового суверенитета.

Цель исследования: решение проблемы ограниченного доступа к актуальной информации путём обеспечения информационной поддержки студентов 24/7 за счет разработки нейросетевого консультанта на базе технологии RAG с использованием российских больших языковых моделей.Задача исследования: провести сравнительный анализ и выбор эмбеддинг моделей, методов индексации, больших языковых моделей наиболее оптимальных для разработки нейросетевого консультанта.

Разработка нейросетевого ассистента с технологией RAG. Технология генерации с дополнительной выборкой (Retrieval Augmented Generation, RAG) – механизм, расширяющий генеративные модели техникой извлечения информации [1]. Приложения, использующие технологию RAG в общем виде, состоят из двух основных компонентов: поискового модуля (retrieval, ретривер) и модуля генерации.

Рисунок 1. Схема пайплайна RAG системы

Рисунок 1 иллюстрирует типовой процесс работы Retrieval-Augmented Generation (RAG) – архитектуры, которая сочетает в себе поиск информации (Retrieval) и генерацию текста (Generation). Основная цель системы – дать языковой модели (LLM) доступ к своим данным, которые изначально не входили в её обучающую выборку.

Описание работы:

1. Запрос пользователя (Question). Взаимодействие с системой начинается с того, что пользователь задает вопрос (Prompt) через интерфейс чата.

2. Ретривер. Полученный текстовый вопрос передается в поисковый модуль, задачей которого является нахождение наиболее релевантных документов в базе знаний.

3. База знаний. В общем виде содержит информацию разных форматов, отвечающей специализированной области.

4. Получение релевантного контекста (Relevant Chunks). В результате поиска извлекаются фрагменты текста (чанки), которые наиболее близки по смыслу к заданному вопросу. Эти фрагменты содержат актуальную информацию, необходимую для ответа.

5. Формирование расширенного промпта. Исходный вопрос пользователя и найденные релевантные фрагменты объединяются в один расширенный промпт. Таким образом, языковой модели передается не только вопрос, но и информацию, на основе которой необходимо дать ответ.

6. Генерация финального ответа (LLM Answer). Большая языковая модель (LLM) получает этот обогащенный промпт и генерирует на его основе итоговый ответ пользователю. Благодаря добавленному контексту, ответ получается основанным на предоставленных данных.

На основании доступных инструментов была выбрана следующая последовательность оценки системы: анализ и оптимизация контекста, поиск наилучшей эмбеддинг модели, поиск лучшего способа индексирования документов, сравнительный анализ результатов генерации языковых моделей.

Компоненты системы и данные. Информационным контекстом системы являются нормативные акты ДонНТУ, поставляются как не оцифрованные файлы формата pdf. Для извлечения информации из таких файлов используется технология оптического распознавания символов (OCR), позволяющая преобразовать текст с изображений в машиночитаемый формат. В первой реализации использовалась библиотека pyteseract, представляющим собой обертку для Google's Tesseract-OCR Engine [2]. В ходе вычисления метрики accuracy, означающую долю правильно найденных документов на общее число запросов, для эмбеддинг модели, значение метрики в среднем принимала значение 0.1, что указывает на крайне низкие поисковые возможности ретривера. Причиной низкого значения является наличие сложных таблиц в документах и перекрытие некоторой информации подписями и печатями, которые pyteseract не смог обработать правильно. Во второй реализации был использован инструмент фрейморка LlamaParse [3], режим сохранения текста в формате языка разметки markdown. По результатам оценки accuracy в среднем увеличилась до 0.8. Основным инструментом анализа компонентов RAG системы в исследовании выступает фреймворк с открытым исходным кодом LlamaIndex [4], доступный в РФ, предоставляющий множество инструментов обработки и извлечения информации, имеет множество интеграций с популярными большими языковыми моделями.

В ходе исследования были протестированы 3 механизма индексации LlamaIndex: Vector Index, List Index и Keyword Table Index. Vector Index – общепринятый стандарт основанный на семантическом поиске. List Index представляет все документы в виде списка. Для поисков релевантных документов передает пользовательский запрос и каждый документ в промпт языковой модели для оценки релевантности. Keyword Table Index формирует таблицу ключевых слов из документов. Поиск релевантных документов осуществляется по наибольшему числу совпавших ключевых слов. Для оценки методов эффективность был использован инструмент фреймворка RetrieverEvaluator, автоматический генерирующий вопросы на основе загруженных документов и вычисляя метрики ранжирования. Тестирование проводилось на 208 вопросах, включающих в себя, например, «На основании каких документов и в каком порядке стипендиальная комиссия определяет список обучающихся, рекомендованных к назначению ПГАС?», «Рассчитайте общее количество баллов, которое получит студент, если он получил только оценки «отлично» в течение трёх семестров подряд, предшествующих назначению ПГАС. Объясните, как вы пришли к этому расчёту.». Vector Index использовался с эмбеддинг модель ru-en-RoSBERTa со следующими параметрами: размер батча – 10; максимальное количество используемых токенов – 512, нормализация векторов включена. List Index и Keyword Table Index использовался с большой языковой моделью DeepSeek-3.2-chat со значением температуры 0.1.

По результатам тестирования (табл. 1) List Index завершил обработку значительно быстрее, получив оценку по метрикам Hit Rate и Recall максимальное значение. Полученные результаты объясняются особенностью работы списочной индексации – List Index оценивает релевантность для всех документов, что гарантировано находит подходящие к запросу, однако полностью исключает ранжирование по семантике, в результате этого значения остальных метрик находятся на низком уровне.

Оценки метрик для Vector Index и Keyword Table Index практически равны, однако скорость обработки запросов Vector Index выше, что является важным фактором для высоконагруженных систем. Исходя из вышеперечисленных выводов, для ретривера нейросетевого консультанта был выбран векторный метод индексирования на основе Vector Index.

Таблица 1. Средние результаты оценок методов индексирования

Метрика	Vector Index	List Index	Keyword Table Index
hit_rate	0,620192	1,000000	0,623188
mrr	0,415385	0,050254	0,352243
precision	0,124038	0,009615	0,070911
recall	0,620192	1,000000	0,623188
ap	0,415385	0,050254	0,352243
ndcg	0,466158	0,207080	0,417133
Общее время обработки, с	47,21	2,63	445,58

Выбор эмбеддинг модели для векторного индексирования прямо влияет на скорость обработки документов и выдачу наиболее релевантной к запросу информации. Для систем RAG используются модели на базе трансформеров. При загрузке документов модель разбивает информацию на чанки, и преобразует их в вектор. Пользовательские запросы также преобразуются в вектор и при помощи метрики косинусного сходства модель извлекает наиболее близкие по значениям вектора документов.

В оценке использовалось 4 open source модели:

- multilingual-e5-small. Мультиязычная модель на 100 млн параметров;
- all-MiniLM-L6-v2. Мультиязычная модель семейства Sentence transformers на 22.7 млн параметров;
- rubert-tiny2. Модель, обученная на русскоязычных текстах на 29.4 млн параметров;
- ru-en-RoSBERTa: Обученная на русскоязычных и англоязычных текстах модель на 400 млн параметров

Оценка проводилась при помощи инструмента фрейморка LlamaIndex InformationRetrievalEvaluator. Оценка работы системы предполагает создание валидационного датасета, строки которого содержат три компонента: запрос пользователя, корпус документов и указание идентификаторов документов, соответствующих данному запросу. Оценка проводилась по основным метрикам ранжирования для 8 пользовательских запросов (табл. 2) [5]. Приведем перечень запросов:

- Что такое повышенная государственная академическая стипендия?
- Что из себя представляет процедура рейтинговой оценки студентов?
- Кто входит в комиссию по предоставлению академического отпуска?
- Какая стипендия положена аспирантам?
- Кто имеет право получить академический отпуск?
- Размер стипендии призедента РФ
- Влияет ли выход на академический отпуск на получение повышенной стипендии?
- Что включает в себя материальное обеспечение детей сирот?

Для вычисления метрик использовались различные значений параметра k, означающего количество релевантных к контексту чанков. При значении k=10 значение метрики recall для всех моделей принимало значение 1 и приводило к падению метрики precision с 0,6250–0,4750 до 0,3750. Такое поведение обусловлено ситуацией, когда найденные документы имеют высокое семантическое сходство, но низкую релевантность к контексту. Наилучшее значение показала модель ru-en-RoSBERTa, сохранив высокие значения precision для k в диапазоне от 1 до 10 и показав лучшие значения для метрик интегрального ранжирования. Для дальнейшего анализа выбрано значение k=5, что соответствует высоким значениям метрики экономит входные токены. По результатам оценок, значительных статистических отличий между мультиязычными и русскоязычными моделями нет, однако русскоязычные модели предпочтительны в выборе с точки зрения построения более структурно правильных ответов для русскоязычных запросов. В качестве эмбеддинг модели была выбрана ru-en-RoSBERTa, поскольку модель показала наилучшие значения и обучена на преимущественно русскоязычных текстах.

Таблица 2. Средние результаты оценок эмбеддинг моделей

Метрика	multilingual-e5-small	all-MiniLM-L6-v2	rubert-tiny2	ru-en-RoSBERTa
accuracy	0,843750	0,843750	0,781250	0,843750
map	0,649681	0,631548	0,605506	0,687798
mrr	0,681250	0,531250	0,503906	0,691406
ndcg	0,638635	0,562953	0,525936	0,657915
precision	0,483333	0,450000	0,433333	0,516667
recall	0,552083	0,581845	0,519345	0,550595

Оценка генеративных моделей. Модулем генерации в целом может выступать любая большая языковая модель, однако в современных реалиях критическими факторами становиться доступность и возможность качественной генерации русскоязычного текста. Немаловажными являются характеристики, формирующие общий показатель качества модели: высокие значения при оценках на популярных бенчмарках, использование современной архитектуры, наличие application programming interface (API), подробная документация. Приведем описание особенностей современных больших языковых моделей.

GigaChat – семейство языковых моделей, разработанных компанией Сбер в России в 2023 году [6]. Взаимодействие с моделью доступно через чат-интерфейс и API. Через API доступны следующие модели: GigaChat-2-Lite для быстрого решения рутинных задач без высоких ресурсных затрат; GigaChat-2-Pro для сложных запросов и прикладных областей; GigaChat-2-Max для работы в сложных научных областях и длинного контекста. Бесплатно предоставляется ограниченное количество токенов для работы с API [7]. Часть моделей выложены в открытый доступ (open source) [8].

DeepSeek – семейство языковых моделей, разработанных компанией DeepSeek в Китае в 2023 году [9]. DeepSeek были выбраны для анализа как представители моделей, разработанных в дружественных странах. Доступ к чату-интерфейсу свободно доступен в России. Использование API платное, однако пополнить баланс возможно только через сторонние сервисы. Через API доступны следующие модели: DeepSeek-v3.2-chat с меньшим количеством выходных токенов и DeepSeek-V3.2-reasoner с большим количеством выходных токенов и дополнительными рассуждениями при обработке запроса [10]. Модели выложены в открытый доступ [11]. GPT – семейство языковых моделей, разработанных компанией Open AI в США [12], модель приведена как представитель западных решений. Доступ к чату-интерфейсу в России заблокирован. Платное API, пополнение доступно только через сторонние сервисы. Доступно множество моделей, включающие новейшие реализации GPT-5.3, так и не поддерживаемые, например, GPT-4.

Таблица 3. Сравнительный анализ моделей языковых моделей

Характеристика	DeepSeek-V3.2	Chat GPT-4o	GigaChat-2-Lite	GigaChat-2-Pro
Прямое пополнение API из РФ	Нет	Нет	Да	Да
Поддержка русского языка	Да	Да	Да	Да
Открытый исходный код	Да	Нет	Да	Да
Архитектура	MoE	-	MoE	MoE
Размер контекстного окна	128 тыс токенов	128 тыс токенов	128 тыс токенов	128 тыс токенов
MMLU	87%	88%	72%	82%
MATH	61%	76%	64%	78%
HumanEval	65%	90%	74%	91%

Перефразированный и уточнённый вариант:

Обобщим обзорную информацию, дополнив её результатами оценки моделей на бенчмарках из статей [11–15] (таблица 3).

Значения в таблице приведены в процентах правильно решённых задач.

Описание бенчмарков:

- MMLU. мультимодальный бенчмарк, оценивающий знания и способность модели решать задачи в 57 различных предметных областях (гуманитарные науки, социальные науки, STEM и др.) с помощью вопросов с выбором ответа.
- MATH. набор из 12 500 сложных математических задач разной сложности, предназначенный для оценки математического мышления и навыков решения задач.
- HumanEval. бенчмарк для оценки способности модели генерировать функционально корректный код.

На основании выбранных технологий ретривера проведена оценка генерации ответов больших языковых моделей.

Для оценки использовались метрики предоставляемые Llama index, использующие метод оценки llm-as-judge, автоматизирующий экспертную оценку путем использования другой большой языковой модели в качестве оценщика.

В проведенном анализе арбитром выступает модель DeepSeek-V3.2-reasoner.

Корпус пользовательских запросов состоял из 8 вопросов, использованных при оценке эмбеддингов.

Параметры генерации для всех моделей были одинаковые: контекстное окно – 128 тыс. токенов, температура – 0.1, k – 5.

Таблица 4. Средние результаты оценок генерации моделей

Модель	Faithfulness score	Answer Relevancy score	Correctness score	Relavency score
DeepSeek-V3.2-chat	1,000	0,928571	4,5625	1,000
GigaChat-2-Lite	0,625	0,500000	3,3125	0,500
GigaChat-2-Pro	0,625	0,562500	3,0625	0,625

Для оценки каждая модель запускалась в формате чата, позволяя использовать системный промпт для регулирования ответов моделей. На основании исследования [16], инструкции, требующие исключить генерацию ответов, если переданный контекст пуст или не релевантен запросу, записаны прописными буквами.

Однако модели GigaChat при таком подходе перестали генерировать ответы на пользовательские запросы, отправляя ответ-заглушку, в общем виде означающий, что модель «не может обсуждать чувствительные темы» или «не обладает собственным мнением». Формирование системного промпта без заглавных букв и общее «смягчение» требований к ответам уменьшило частоту появления ответов-заглушек.

Итоговый системный промпт представлен ниже:

«Ты консультант ДонНТУ. Отвечай строго по документам ниже.

Если в контексте нет ответа – напиши "Информация в документах отсутствует или недостаточна".

Контекст из документов:

{context_str}

Вопрос пользователя:

{query_str}

Ответ:»

По результатам оценок (табл. 4) DeepSeek-V3.2-chat показал значительное преимущество по метрикам. Низкие оценки средних метрик моделей GigaChat обусловлены ответами-заглушками, которые все ещё встречаются в ответах вследствие чего в среднюю

оценку добавляются нули. Ответы такого типа встречаются на такие запросы как «Кто имеет право получить академический отпуск?» или «Что включает в себя материальное обеспечение детей сирот?». Модель DeepSeek-V3.2-chat для тех же запросов генерирует ответы, метрики которых близки к максимальным.

На Рисунке 2 представлены результаты оценок метрик для каждого вопроса для каждой языковой модели. По оси x обозначен номер вопроса, по оси y – значение метрики. Оценивая значения метрик без учета ответов-заглушек, метрики моделей GigaChat-2-lite и GigaChat-2-pro близки к ответам DeepSeek-V3.2-chat.

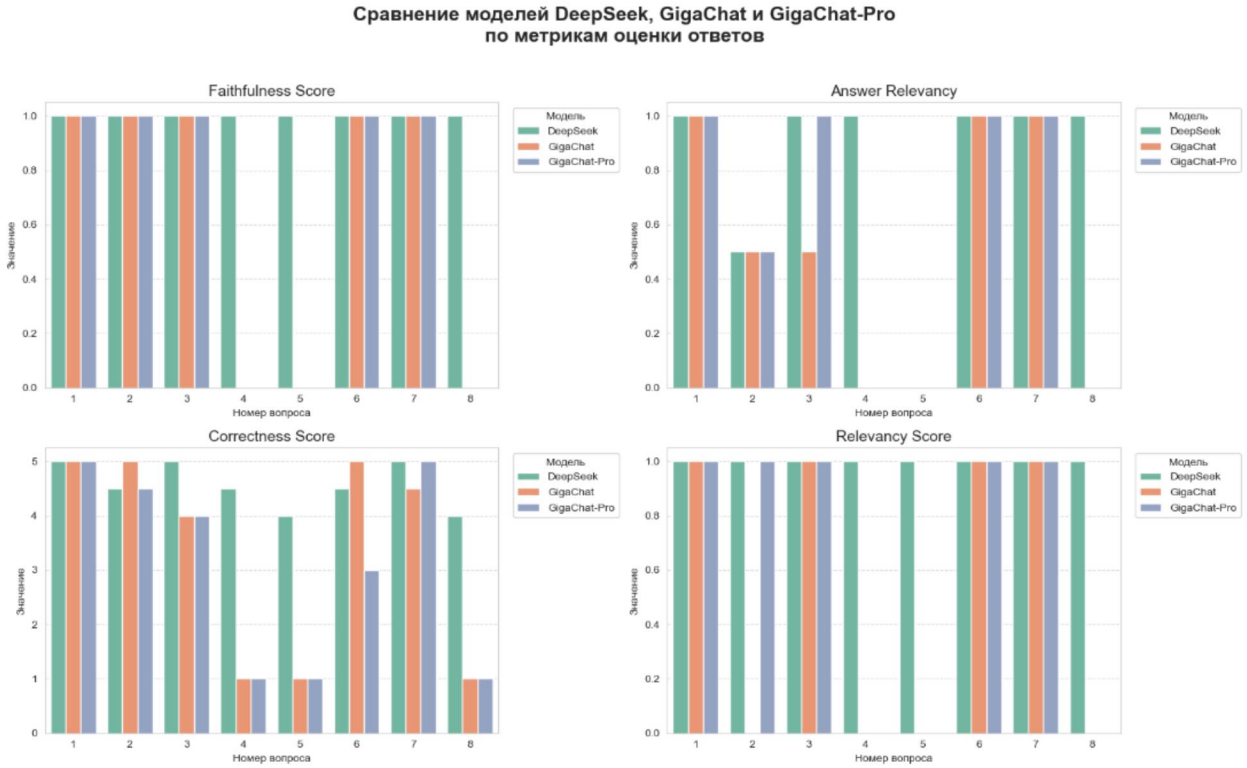

Рисунок 2. Сравнение моделей по метрикам оценки ответов

Таким образом, уровень генерации ответов моделей GigaChat-2-lite и GigaChat-2-pro сопоставим с уровнем генерации DeepSeek-V3.2-chat, что в совокупности с доступностью моделей GigaChat на территории РФ, склоняет к выбору моделей от компании Сбер для построения консультирующей RAG системы.

Заключение. В результате исследования подобрана оптимальная конфигурация технологий для построения нейросетевого консультанта: извлечение информации из неоцифрованных документов при помощи llama parse, векторная индексация с эмбендинг моделью ru-en-RoSBERTa, модуль генерации ответов на запросы на основе GigaChat-2-Lite и GigaChat-2-Pro.

Использование Open Source технологий и отечественных больших языковых моделей позволяет исключить влияние санкционных ограничений на стабильность работы системы и использовать современные подходы построения RAG систем.

В качестве дальнейшего исследование планируется разработка дружественного пользовательского интерфейса для студента и для администратора, который занимается поддержкой системы (управление хранилищем, актуализация нормативной базы и тестирование консультанта).

Список литературы

[1] Ротман Д. RAG и генеративный ИИ. Создаем собственные RAG-пайплайны с помощью LlamaIndex, Deep Lake и Pinecon. – СПб. : Питер, 2026. – 320 с.

[2] Tesseract OCR [Электронный ресурс] // GitHub. – URL: https://github.com/UB-Mannheim/tesseract/wiki.

[3] LlamaParse [Электронный ресурс] // GitHub. – URL: https://github.com/maxgoff/llama_parse.

[4] LlamaIndex [Электронный ресурс]. – URL: https://www.llamaindex.ai/.

[5] Задача ранжирования [Электронный ресурс] // Яндекс Образование. – URL: https://education.yandex.ru/handbook/ml/article/zadacha-ranzhirovaniya.

[6] Зачем нужен GigaChat? [Электронный ресурс] // SberDevices. – URL: https://sberdevices.ru/press/blog/detail/zachem-nuzhen-giga-chat/.

[7] Обновления моделей GigaChat [Электронный ресурс]. – URL: https://developers.sber.ru/docs/ru/gigachat/models/updates.

[8] GigaChat3 [Электронный ресурс] // Hugging Face. – URL: https://huggingface.co/collections/ai-sage/gigachat3.

[9] DeepSeek [Электронный ресурс] // Википедия. – URL: https://ru.wikipedia.org/wiki/DeepSeek.

[10] DeepSeek API Docs [Электронный ресурс]. – URL: https://api-docs.deepseek.com.

[11] deepseek-ai/DeepSeek-V3.2-Exp [Электронный ресурс] // Hugging Face. – URL: https://huggingface.co/deepseek-ai/DeepSeek-V3.2-Exp.

[12] GPT-4o [Электронный ресурс]. – URL: https://openai.com/ru-RU/index/hello-gpt-4o/. – (Дата обращения: 31.03.2026).

[13] GigaChat Lite [Электронный ресурс]. – URL: https://developers.sber.ru/docs/ru/gigachat/models/gigachat-2-lite.

[14] GigaChat Pro [Электронный ресурс]. – URL: https://developers.sber.ru/docs/ru/gigachat/models/gigachat-2-pro.

[15] Оценка производительности DeepSeek-V3 [Электронный ресурс] // Habr. – URL: https://habr.com/ru/articles/955282/.

[16] CAPS for prompt engineering [Электронный ресурс]. – URL: https://kth.diva-portal.org/smash/record.jsf?dswid=-5361&pid=diva2%3A1954037.

Авторский вклад

Васяева Татьяна Александровна – руководство исследованием, участие в проведении анализа полученных данных.

Душа Михаил Валерьевич – реализация RAG системы, подбор метрик и параметров системы, анализ полученных данных.

DEVELOPMENT OF A NEURAL NETWORK CONSULTANT FOR STUDENTS BASED ON LARGE LANGUAGE MODELS

T.A. Vasyaeva

Dean of the Faculty of Information Systems and Technologies, Associate Professor of the Department of Automated Control Systems, Donetsk National Technical University, Candidate of Technical Sciences, Associate Professor

tanechka.vasyaeva@yandex.ru

M.V. Dusha

4th-year Bachelor Student, Department of Automated Control Systems, Donetsk National Technical University

dushamihail@yandex.ru

Abstract. The article is devoted to the development of a neural network consultant for student information support based on Retrieval-Augmented Generation (RAG) technology using Russian large language models. A comparative analysis of embedding models, document indexing methods using the LlamaIndex framework, and generative models GigaChat-2-Lite, GigaChat-2-Pro, and DeepSeek-V3.2 was conducted. The optimal set of tools is demonstrated: ru-en-RoSBERTa, vector indexing, and GigaChat.

Keywords. large language models, RAG, GigaChat, LlamaIndex, embedding models, vector indexing, LLM as a Judge, neural network consultant, student information support.