Информатика и вычислительная техника
Informatics and computer engineering

Научная статья
УДК 004.89
DOI: 10.14529/ctcr250201

СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ RAG ДЛЯ ПОСТРОЕНИЯ РУССКОЯЗЫЧНЫХ ИНТЕЛЛЕКТУАЛЬНЫХ СЕРВИСОВ

А.В. Мельников[1]*, MelnikovAV@uriit.ru, https://orcid.org/0000-0002-1073-7108*
И.Е. Николаев[2]*, ivan_nikolaev@csu.ru, https://orcid.org/0000-0002-9686-2435*
М.А. Русанов[3]*, RusanovMA@uriit.ru, https://orcid.org/0000-0002-9926-4609*
В.Р. Аббазов[1]*, AbbazovVR@uriit.ru, https://orcid.org/0009-0008-9315-2041*

[1] *Югорский научно-исследовательский институт информационных технологий, Ханты-Мансийск, Россия*
[2] *Челябинский государственный университет, Челябинск, Россия*
[3] *Югорский государственный университет, Ханты-Мансийск, Россия*

Аннотация. В статье рассматривается один из наиболее популярных в настоящее время подходов к построению различных типов интеллектуальных помощников и запрос-ответных систем на базе больших языковых моделей (LLM), основанный на in-context learning или retrieval augmented generation (RAG). Появившееся в последнее время множество публикаций на эту тему в первую очередь ориентировано на английский язык и использует такие ведущие по качеству модели, как GPT-4o и их развитие. В то же время оценки методов поиска контекста RAG для задач на русском языке практически отсутствуют, что делает актуальной задачу проведения исследований, направленных на адаптацию и оценку этих методов для русского языка. **Цель исследования:** изучить эффективность различных подходов retrieval augmented generation (RAG) для русскоязычных задач, учитывая, что большинство исследований в этой области ориентированы на английский язык и используют ведущие модели, такие как GPT-4. **Материалы и методы.** В статье рассматриваются три базовых подхода к построению RAG: naive RAG, HyDE и вероятностный подход, основанный на функции BM25. Особое внимание уделяется оценке качества этих методов по метрике mean average precision (mAP) для трех областей знаний. Комбинированные методы RAG, такие как SelfRAG, не использовались, чтобы получить отдельные оценки каждого подхода. Для экспериментов были отобраны корпуса текстов на русском языке для областей знаний – нефтегазовой промышленности и юриспруденции. **Результаты.** Проведенное исследование позволило получить оценки качества для каждого из рассмотренных методов. Результаты хорошо согласуются с данными других исследований, но уступают известным RAG на английском языке. **Заключение.** Полученные результаты могут быть использованы как базовые оценки (baseline) и в качестве основы для принятия решений по выбору оптимальных архитектур RAG для русскоязычных задач. Дальнейшие исследования будут направлены на интеграцию комбинированных методов и адаптацию моделей для повышения качества генерации на русском языке.

Ключевые слова: вопросно-ответные системы, большие языковые модели, LLM, RAG, оценка качества RAG, HyDE, BM25

Для цитирования: Сравнительный анализ методов RAG для построения русскоязычных интеллектуальных сервисов / А.В. Мельников, И.Е. Николаев, М.А. Русанов, В.Р. Аббазов // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». 2025. Т. 25, № 2. С. 5–18. DOI: 10.14529/ctcr250201

Original article
DOI: 10.14529/ctcr250201

COMPARATIVE ANALYSIS OF RAG METHODS FOR BUILDING RUSSIAN-SPEAKING INTELLIGENT SERVICES

A.V. Melnikov[1], *MelnikovAV@uriit.ru, https://orcid.org/0000-0002-1073-7108*
I.E. Nikolaev[2], *ivan_nikolaev@csu.ru, https://orcid.org/0000-0002-9686-2435*
M.A. Rusanov[3], *RusanovMA@uriit.ru, https://orcid.org/0000-0002-9926-4609*
V.R. Abbazov[1], *AbbazovVR@uriit.ru, https://orcid.org/0009-0008-9315-2041*
[1] *Ugra Research Institute of Information Technologies, Khanty-Mansiysk, Russia*
[2] *Chelyabinsk State University, Chelyabinsk, Russia*
[3] *Yugra State University, Khanty-Mansiysk, Russia*

Abstract. The paper discusses one of the currently most popular approaches to building various types of intelligent assistants and query-response systems based on large language models (LLMs), based on in-context learning or retrieval augmented generation (RAG). The recent proliferation of publications on this topic is primarily English-oriented and utilizes leading-quality models such as GPT-4o and their developments. At the same time, evaluations of RAG context search methods for Russian language tasks are practically absent, which makes it an urgent task to conduct research aimed at adapting and evaluating these methods for the Russian language. **Aim.** To study the effectiveness of different retrieval augmented generation (RAG) approaches for Russian-language tasks, given that most studies in this area are English-oriented and use leading models such as GPT-4. **Materials and Methods.** The paper reviews three basic approaches to RAG construction: naive RAG, HyDE, and a probabilistic approach based on the BM25 function. Particular attention is paid to assessing the quality of these methods in terms of the mean average precision (mAP) metric for the three knowledge domains. Combined RAG methods such as SelfRAG were not used to obtain separate evaluations of each approach. Russian language text corpora were selected for the experiments for the knowledge domains: oil and gas industry and jurisprudence. **Results.** The conducted study allowed us to obtain quality scores for each of the considered methods. The results agree well with the data of other studies, but are inferior to the known RAGs in English. **Conclusion**. The obtained results can be used as baseline evaluations and as a basis for making decisions on selecting optimal RAG architectures for Russian-language tasks. Further research will be aimed at integrating combined methods and adapting models to improve the quality of Russian language generation.

Keywords: question and answer systems, large language models, LLM, RAG, quality assessment RAG, HyDE, BM25

For citation: Melnikov A.V., Nikolaev I.E., Rusanov M.A., Abbazov V.R. Comparative analysis of RAG methods for building Russian-speaking intelligent services. *Bulletin of the South Ural State University. Ser. Computer Technologies, Automatic Control, Radio Electronics*. 2025;25(2):5–18. (In Russ.) DOI: 10.14529/ctcr250201

Введение

В последние годы наблюдается значительный рост интереса к интеллектуальным помощникам и вопросно-ответным системам, базирующимся на больших языковых моделях (от англ. LLM – Large Language Model). Успех таких технологий во многом обусловлен их способностью обрабатывать и генерировать текст, что делает их важными инструментами в различных сферах – от образования до бизнеса. Особенно актуально применение таких подходов для работы с русскоязычными данными, однако большинство существующих исследований сосредоточено на английском языке, оставляя пробелы в понимании и оценке методов, применимых к русскоязычным системам.

Модели, основанные на подходах in-context learning и retrieval augmented generation (от англ. RAG – Retrieval-Augmented Generation), способны значительно повысить качество взаимодействия с пользователем, однако их адаптация к русскому языковому контексту требует дополнительного анализа и исследования. В то время как технология RAG достигла значитель-

ных успехов в англоязычной среде, отсутствие систематических оценок и сравнений для русского языка ограничивает возможности разработки эффективных технологий в этом языковом сегменте.

Актуальность данного исследования заключается в необходимости создания и оценки методик применения RAG в контексте русскоязычных данных. Результаты исследования помогут создать базис для дальнейших сравнений и улучшений в области разработки интеллектуальных систем, а также способствовать более широкому внедрению технологий обработки языка в русскоязычной среде.

Большие языковые модели демонстрируют впечатляющие возможности, но сталкиваются с такими проблемами, как галлюцинации, устаревшая информация и не отслеживаемые процессы аргументации модели. Технология RAG стала многообещающим решением, включающим информацию из внешних баз данных. Применение RAG повышает точность и достоверность ответов системы, особенно для задач, требующих больших объемов данных, и позволяет непрерывно обновлять и интегрировать информацию, специфичную для предметной области [1].

Было предложено много подходов RAG для улучшения больших языковых моделей посредством зависимых от запроса изменений [1–3]. Типичный алгоритм RAG содержит несколько последовательных этапов обработки:

1) классификация запроса – определение необходимости поиска и набора документов для заданного входного запроса;

2) поиск – эффективное получение релевантных документов для запроса;

3) повторное ранжирование – уточнение порядка найденных документов на основе их релевантности запросу;

4) переупаковка – организация найденных документов в структурированную форму для повышения эффективности;

5) обобщение – извлечение ключевой информации для создания ответа из переупакованного документа и устранения избыточности.

Реализация RAG также требует принятия решений о том, как правильно разбивать документы на фрагменты, какие модели эмбеддинга использовать для семантического представления этих фрагментов, как выбрать векторные базы данных для эффективного хранения представлений признаков и как эффективно настраивать большие языковые модели.

Дополнительную сложность и трудность представляет вариативность реализации каждого этапа типичного рабочего процесса RAG. Например, при поиске релевантных документов по входному запросу могут использоваться различные методы. Одним из таких методов может быть переписывание запроса, а затем использование его для поиска [4]. В качестве альтернативы можно сначала сгенерировать псевдоответы на запрос, а затем сравнить сходство между этими псевдоответами и документами для поиска [5]. Другой вариант – непосредственное использование эмбеддинга, которому обычно обучаются контрастным способом на парах положительных и отрицательных запросов-ответов [6, 7]. Выбранные для каждого этапа методы и их комбинации существенно влияют как на эффективность, так и на производительность RAG. При этом следует учитывать, что для существенной части прикладных решений требуется обеспечение конфиденциальности данных, и, как следствие, появляется ограничение на использование только локальных моделей.

В контексте RAG очень важно эффективно извлекать соответствующие документы из источника данных [1], при этом одним из ключевых вопросов является выбор соответствующей модели эмбеддинга.

В RAG поиск осуществляется путем вычисления сходства между эмбеддингами запросов и фрагментов документов, при этом ключевую роль играет способность моделей эмбеддингов к семантическому представлению. Наиболее популярные модели эмбеддингов – BERT. Наравне с моделями эмбеддингов применяется вероятностный алгоритм BM25, однако в последних исследованиях были представлены такие известные модели эмбеддингов, как AngIE, Voyage, BGE [8–10]. Стоит отметить, что не существует универсального ответа на вопрос, какую модель эмбеддинга использовать; как указано в статье [1], модели с разной архитектурой лучше подходят для конкретных случаев использования.

1. Смешанный или гибридный поиск применяется, когда разреженные и плотные модели эмбеддинга могут извлечь выгоду друг из друга, так как отражают различные характеристики релевантности информации. Например, разреженные модели поиска могут быть использованы для получения начальных результатов поиска для обучения плотных моделей поиска. Также разреженные модели могут улучшить возможности плотных моделей поиска без примеров и помочь плотным моделям обрабатывать запросы, содержащие редкие сущности, тем самым повышая устойчивость.

2. Тонкая настройка модели эмбеддинга применяется в случаях, когда контекст значительно отличается от обучающего набора данных, особенно в узкоспециализированных дисциплинах, таких как здравоохранение, юриспруденция и другие отрасли, изобилующие специализированными терминами. Тонкая настройка модели эмбеддинга происходит на собственном наборе данных по конкретной тематике и уменьшает расхождение в семантике текстов.

Наличие шума или противоречивой информации во время поиска может негативно повлиять на качество работы RAG. Эту ситуацию образно описывают, как «дезинформация может быть хуже, чем отсутствие информации вообще». Повышение устойчивости RAG к таким нежелательным входным данным становится популярным в исследованиях и стало ключевой метрикой производительности [11–13]. Согласно [14], результаты проведенного анализа типа извлекаемых документов и оценки релевантности документов запросу, их положение и количество, включенное в контекст, показывают, что включение нерелевантных документов может неожиданно повысить точность более чем на 30 %, что противоречит первоначальному предположению о снижении качества. Эти результаты подчеркивают важность разработки специализированных стратегий для интеграции поиска с моделями генерации языка, а также необходимость дальнейших исследований и изучения надежности RAG.

Важно отметить, что для оценки выбора релевантных документов и оценки ответов больших языковых моделей нет общепризнанных метрик оценки качества. Чаще всего метрики EM, F1, BLEU или ROUGE используют для оценки ответа на вопрос [4, 15–17], accuracy используют для оценки наличия факта в ответе [15, 18]. Для оценки качества выбора релевантных документов используются такие метрики, как mean average precision (mAP), которая учитывает как точность извлечения, так и порядок извлеченных документов, а также mean reciprocal rank (MRR) [19, 20].

Отдельно выделим автоматизированные метрики библиотеки RAGAS (от англ. Retrieval Augmented Generation Assessment) [21, 22], для оценки ответа на вопрос используются метрики *Answer relevance, Answer correctness, Faithfulness*, а для оценки выбора контекста для ответа на вопрос используются метрики *Context precision, Context recall, Context utilization* [23].

В работе проводится сравнение различных базовых подходов к построению RAG, включающих naive RAG, HyDE и BM25, с возможностью последующего построения гибридного RAG для достижения наилучших результатов под различные задачи.

Данные и модели

Источники данных

Для проведения экспериментов использовалось 3 источника данных, разделенных по предметным областям: информационные технологии (на английском языке), нормативно-правовые акты ХМАО-Югры (на русском языке) и учебно-методические издания по нефтегазовой отрасли (на русском языке). Так как цель эксперимента – сравнить модели для русского языка, то основными считались датасеты по нормативно-правовым актам ХМАО-Югры и учебно-методические издания по нефтегазовой отрасли, а датасет по информационным технологиям был вспомогательным. Информация в источниках данных была представлена в виде книг, статей, отзывов, различного вида нормативных документов. Далее они были преобразованы в текстовые документы и разделены на чанки (текстовые блоки) с использование RecursiveCharacterTextSplitter из библиотеки LangChain [24]. Итоговая информация по исходным датасетам представлена в табл. 1.

Таблица 1

Исходные датасеты

Table 1

Source datasets

Предметная область	Виды документов	Кол-во чанков
Юриспруденция (LAW)	Нормативно-правовые акты, действующие на территории Ханты-Мансийского автономного округа – Югры	15 803
Нефтегазовая промышленность (OIL)	Специализированные тексты, охватывающие вопросы разработки месторождений, геологии и технологий добычи нефти и газа	4 708
Информационные технологии (IT)	Открытый набор данных WMT 2016 IT Translation Task, содержащий ответы на вопросы по устранению неполадок в сфере аппаратного и программного обеспечения	217

Данные для оценки

Для каждой предметной области было сформировано по 40 образцов. Примеры данных из оценочных датасетов представлены в табл. 2. Каждая единица данных в оценочных датасетах представляет собой комплексную структуру, состоящую из следующих элементов:

1. Вопрос. Сформулированный запрос, требующий ответа.

2. Контексты. Подбор релевантных текстовых фрагментов, служащих основой для формирования ответа.

3. Правильный ответ. Эталонный ответ, соответствующий заданному вопросу.

4. Категоризация вопроса по типу: простой, требующий рассуждения (вопросы, требующие от модели рассуждения для эффективного ответа), условный (вопрос, основанный на цепочке связей «A→B→C»), мультиконтекстный (вопрос сформирован на основании нескольких фрагментов текста).

Всего правильных контекстов (текстовых фрагментов) для областей данных «информационные технологий», «юриспруденция», «нефтегазовая промышленность» – 52, 52, 53 соответственно.

Таблица 2

Пример данных из оценочных датасетов

Table 2

Example of data from estimated datasets

Область данных	Вопрос	Контекст	Правильный ответ
Информационные технологии (IT)	How can you reset the browser settings to default?	Update the network card driver. Install the drivers for your wireless card. Try with another computer and browser. If the situation persists, the problem is with the website itself. Please check if the network cable is properly connected. Check the IP settings and open the respective ports on the router VPI = 0, VCI = 35 You must access the internal page of the router and perform the opening via 'port forwarding' or DMZ host. In case you changed the password, I suggest you reset the equipment to get back to factory settings…	Try to delete the navigation history, the temporary files and restore to default the browser settings

Окончание табл. 2
Table 2 (end)

Область данных	Вопрос	Контекст	Правильный ответ
Юриспруденция (LAW)	Каковы основные функции Управления по делам архивов Ханты-Мансийского автономного округа?	Постановление Правительства Ханты-Мансийского автономного округа от 16 октября 2000 г. N 21-п Об Управлении по делам архивов Ханты-Мансийского автономного округа – Югры. В целях приведения Положения об Управлении по делам архивов Ханты-Мансийского автономного округа – Югры…	Основные функции Управления по делам архивов Ханты-Мансийского автономного округа включают проведение государственной политики в сфере архивного дела, контроль за сохранностью, комплектованием и использованием документов
Нефтегазовая промышленность (OIL)	Какую роль играют газовые сепараторы в повышении эффективности работы насосов в скважинах с подводной устьевой арматурой?	Недавно системы винтовых насосов, извлекаемых при помощи канатно-тросовых операций, были использованы в скважинах с большим отклонением от вертикали в регионе Юго-Восточной Азии. Применение винтовых насосов в этом случае было усложнено проблемами выноса пластового песка, отложениями солей, добычей тяжелой нефти и заканчиванием скважин с малым диаметром НКТ…	Газовые сепараторы способствуют повышению эффективности работы насосов в скважинах с подводной устьевой арматурой за счет уменьшения объемов свободного газа, поступающего на вход насоса

Векторные модели

Одним из главных элементов RAG-систем является модель генерации эмбеддингов. Для генерации эмбеддингов использовались модели intfloat/e5-mistral-7b-instruct [25] (далее mistral), которая является архитектурой для генерации текстовых эмбеддингов, основанной на большой языковой модели Mistral-7B, и модель infloat/multilingual-e5-large [26] (далее e5), которая основана на модели xml-roberta-large. Отличительной чертой модели e5-mistral-7b-instruct является дообучение на синтетических данных, сгенерированных с помощью GPT-4, что позволило достичь высоких показателей на бенчмарках MTEB и BEIR. Указанные модели выбраны из-за их способности эффективно обрабатывать различные задачи, связанные с текстовыми эмбеддингами, что подтверждается высокими позициями в бенчмарке MTEB для русского языка [27]. В качестве векторного индекса использовалась библиотека FAISS [28]. FAISS является мощным инструментом для поиска документов на основе их векторных представлений, обеспечивая высокую скорость и точность поиска на больших наборах данных.

Описание экспериментов

Подготовительный и финальный этапы – общие для всех экспериментов

Для экспериментов, требующих генерации эмбеддингов, использовались две векторные модели – mistral размерностью 4096 и e5 размерностью 1024. При формировании чанков использовался метод RecursiveCharacterTextSplitter из библиотеки langchain [29], формирующий чанки размером 2048 и перекрытием 256 символов.

Алгоритм действий финального этапа генерации ответов:

1. Отбор двух наиболее релевантных чанков.

2. Формирование контекста. Отобранные чанки объединялись с исходным вопросом в единый контекст.

3. Генерация финального ответа языковой моделью, используя предоставленный контекст и собственные знания.

Для каждого эксперимента были определены уровни количества чанков в поисковой выдаче ретриверов: 1, 3, 5, 10, 20, 50, 100. Все последующие метрики рассчитывались для этих уровней отдельно.

Далее представлены эксперименты с описанием особенностей их реализации.

Эксперимент 1. Наивный RAG

В первом эксперименте была применена стандартная архитектура RAG. Данная архитектура представляет собой гибридный подход, сочетающий преимущества информационного поиска и генеративных языковых моделей.

Алгоритм эксперимента № 1:

1. Подготовительный этап:

a) формирование хранилища чанков;

b) построение векторного хранилища чанков;

c) настройка конфигурации ретривера по векторному хранилищу чанков;

2. Процесс извлечения:

a) векторизация запроса, совместимая с векторами чанков;

b) семантический поиск. Отбор наиболее релевантных чанков между вектором запроса и векторами чанков;

c) ранжирование отобранных чанков по степени релевантности.

3. Генерация ответа (см. описание выше).

Эксперимент 2. RAG + HyDE

Во втором эксперименте была использована модифицированная архитектура RAG с интеграцией метода HyDE (Hypothetical Document Embeddings) [30]. Данная модификация направлена на улучшение процесса извлечения релевантной информации путем генерации гипотетического ответа.

Алгоритм эксперимента № 2:

1. Подготовительный этап (см.описание выше):

a) формирование хранилища чанков;

b) построение векторного хранилища чанков;

c) настройка конфигурации ретривера по векторному хранилищу чанков;

d) оптимизация параметров HyDE. Настройка гиперпараметров для генерации гипотетического ответа и его интеграции в процесс поиска.

2. Генерация гипотетического ответа языковой моделью:

a) для генерации гипотетического ответа использовался промпт: «Ответь на вопрос пользователя. Твой ответ должен быть не длиннее 50 слов».

3. Усовершенствованное извлечение:

a) векторизация гипотетического ответа. Сгенерированный ответ преобразовывался в векторное представление, совместимое с векторами чанков;

b) семантический поиск наиболее релевантных чанков между вектором гипотетического ответа и векторами чанков;

c) ранжирование отобранных чанков по степени релевантности.

4. Генерация ответа (см. описание выше).

В данном эксперименте проводилось дополнительное исследование влияния длины сгенерированных гипотетических ответов на качество информационного поиска. Тестировались 4 вариации длины сгенерированных ответов: короткая (short), средняя (medium), длинная (long), без ограничений (unlimited) (табл. 3).

Таблица 3

Средние длины сгенерированных гипотетических ответов по методу HyDE

Table 3

Average lengths of generated hypothetical responses using the HyDE method

Область данных	Короткая (short)	Средняя (medium)	Длинная (long)	Без ограничений (unlimited)
LAW	345	1078	1419	3303
OIL	318	885	1470	3110
IT	397	855	1173	2710

Эксперимент 3. BM25

В третьем эксперименте был использован метод Okapi BM25 [31], один из наиболее широко используемых и эффективных методов ранжирования в информационном поиске. Разработанный в 1990-х годах, этот алгоритм основан на пробабилистической модели поиска и улучшает релевантность документов, учитывая частоту терминов и длину документов.

Okapi BM25 эффективно ранжирует документы, справляясь с избыточностью терминов в длинных документах, и хорошо масштабируется для больших коллекций. Он широко используется в современных поисковых системах, таких как Elasticsearch и Apache Lucene, благодаря своей эффективности и простоте реализации. В последние годы наблюдается рост интереса к гибридным методам, сочетающим BM25 с нейронными сетями и машинным обучением, что позволяет улучшить качество ранжирования, используя преимущества как традиционных, так и современных алгоритмов.

Описание метрик оценки RAG-систем

Mean Average Precision (MAP) – это метрика, используемая для оценки качества систем поиска информации, таких как системы поиска текста. Она учитывает как точность (precision), так и полноту (recall) поиска. Давайте рассмотрим формулы для расчета MAP для задачи text retrieval.

1. Точность@k (precision@k, P) на позиции k (Precision@k) определяется как доля релевантных документов среди первых k документов, возвращаемых системой:

$$\text{Precision@k} = (\text{Количество релевантных документов среди первых } k)/k. \quad (1)$$

2. Средняя точность (average precision, AP) для одного запроса – это среднее значение точности в точках где найдены релевантные документы:

$$\text{AP} = \frac{1}{N}\sum_{k=1}^{M} P(k) \cdot r(k), \quad (2)$$

где N – количество релевантных документов;

M – общее количество документов;

$P(k)$ – точность на позиции k;

$r(k)$ – бинарная метка релевантности документа на позиции k (1 – если документ релевантен, и 0 – если нет).

3. Усредненная средняя точность (mean average precision, MAP) – это усредненное значение средней точности по всем запросам:

$$\text{MAP} = \frac{1}{Q}\sum_{q=1}^{Q} AP_q, \quad (3)$$

где Q – количество запросов;

AP_q – средняя точность для запроса q.

Рассмотрим пример с одним запросом и списком документов, возвращаемых системой:

- возвращаемые документы: [D1, D2, D3, D4, D5];
- релевантные документы: [D1, D3, D5].

1. Расчет точности@k:

- $P(1) = \frac{1}{1} = 1$;
- $P(1) = \frac{1}{2} = 0{,}5$;

- $P(1) = \frac{2}{3} = 0{,}67$;
- $P(1) = \frac{2}{4} = 0{,}5$;
- $P(1) = \frac{3}{5} = 0{,}6$.

2. Расчет средней точности:

$AP = \frac{1}{3} = (1 \cdot 1 + 0{,}67 \cdot 1 + 0{,}6 \cdot 1) \approx 0{,}76.$

3. Расчет усредненной средней точности. Если у нас несколько запросов, тогда мы усредняем полученные средние точности для всех запросов.

Результаты и обсуждение

В данном разделе представлены результаты экспериментов, проведенных для оценки эффективности различных методов retrieval augmented generation (RAG).

На рис. 1 представлен график зависимости количества правильных документов от размера поисковой выдачи для метода BM25 для набора данных «нефтегазовая промышленность». График демонстрирует, что с увеличением размера поисковой выдачи количество правильных документов также увеличивается, достигая пика на уровне 49 правильных документов из 53 возможных при 50 документах в выдаче.

Исходя из анализа графика на рис. 1, можно заключить, что размер поисковой выдачи в 20 чанков является оптимальным, так как он обеспечивает баланс между качеством и вычислительными затратами. На этом уровне достигается значительное количество правильных документов (90 % от максимума), и дальнейшее увеличение размера выдачи не приводит к существенному росту точности, что не оправдывает дополнительных вычислительных ресурсов.

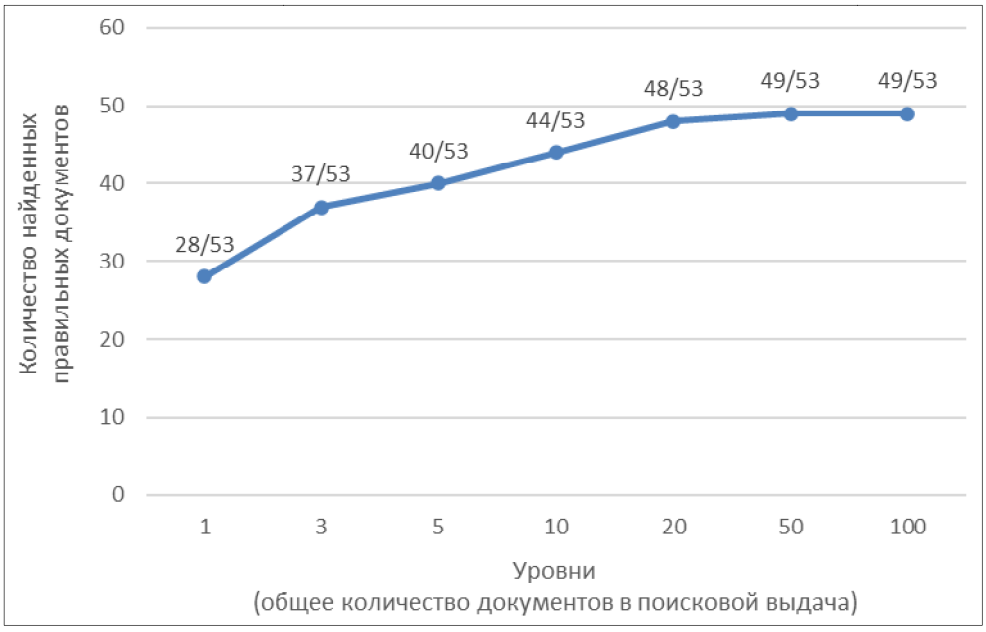

Рис. 1. График зависимости количества правильных документов от размера поисковой выдачи для метода bm25 для набора данных «нефтяная промышленность»
Fig. 1. Graph of the dependence of the number of correct documents on the size of the search output for the bm25 method for the oil industry dataset

На рис. 2 приведено сравнение эффективности методов BM25, naive_e5 и naive_mistral для набора данных «нефтегазовая промышленность». Из графика видно, что метод BM25 показывает наилучшие результаты на малых уровнях поисковой выдачи. Методы naive_e5 и naive_mistral также демонстрируют хорошие результаты, но уступают BM25, особенно на малых уровнях выдачи.

Табл. 4 представляет результаты экспериментов для 20 документов в поисковой выдаче по трем предметным областям. Из табл. 4 видно, что метод BM25 показывает наилучшие результаты для набора данных «нефтегазовая промышленность», достигая 49 правильных документов из

53 возможных с mAP@20, равным 0,737. Метод HyDE с моделью mistral также показывает высокие результаты, особенно при использовании длинных гипотетических ответов. Например, для набора данных «нефтегазовая промышленность» HyDE с моделью mistral показывает 46 правильных документов из 53 возможных с mAP@20, равным 0,562.

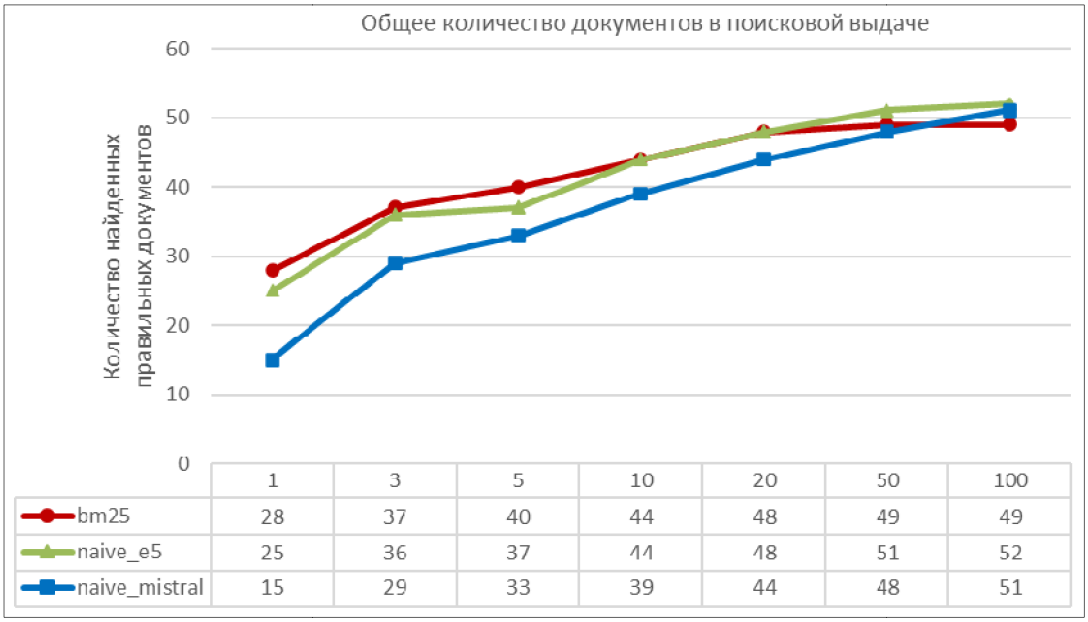

Рис. 2. График зависимости количества правильных документов от размера поисковой выдачи для метода bm25, naive_e5, naive_mistral для набора данных «нефтяная промышленность»
Fig. 2. Graph of the dependence of the number of correct documents on the size of the search output for the bm25, naive_e5, and naive_mistral methods for the oil industry dataset

Таблица 4

Результаты экспериментов для 20 документов в поисковой выдаче по трем предметным областям

Table 4

The results of experiments for 20 documents in search results in three subject areas

Набор данных	Название метода	Векторная модель	Количество правильных документов	mAP@20	Процент по уровням	Не найдено документов
IT	naive	e5	43	0,51	0,83	9
IT	naive	mistral	45	0,53	0,87	7
IT	hyde (unlimited)	e5	44	0,49	0,85	8
IT	hyde (unlimited)	mistral	46	0,58	0,88	6
IT	bm25lc		44	0,52	0,85	8
IT	bm25lc (preproc)		43	0,55	0,83	9
OIL	naive	e5	48	0,68	0,92	4
OIL	naive	mistral	44	0,52	0,85	8
OIL	hyde (unlimited)	e5	40	0,53	0,77	12
OIL	hyde (unlimited)	mistral	46	0,56	0,88	6
OIL	bm25lc		48	0,74	0,92	4
OIL	bm25lc (preproc)		49	0,74	0,94	3
LAW	naive	e5	37	0,34	0,71	15
LAW	naive	mistral	39	0,40	0,75	13
LAW	hyde (unlimited)	e5	30	0,31	0,58	22
LAW	hyde (unlimited)	mistral	37	0,38	0,71	15
LAW	bm25lc (preproc)		36	0,39	0,69	16
LAW	bm25lc (preproc)		37	0,40	0,71	15

Табл. 5 представляет результаты экспериментов для различных вариаций метода HyDE. Из табл. 5 видно, что длина сгенерированных гипотетических ответов существенно влияет на качество результатов. Например, для набора данных «нефтегазовая промышленность» метод HyDE с моделью mistral и длинными гипотетическими ответами показывает 46 правильных документов из 53 возможных с mAP@20, равным 0,562, в то время как короткие гипотетические ответы дают менее стабильные результаты.

Таблица 5

Результаты экспериментов для 20 документов в поисковой выдаче для вариаций методов hyde

Table 5

Experimental results for 20 documents in search results for variations of hyde methods

Набор данных	Название метода	Векторная модель	Количество правильных документов	mAP@20	Процент по уровням	Не найдено документов
IT	hyde (unlimited)	e5	44	0,489	0,85	8
IT	hyde (unlimited)	mistral	46	0,577	0,88	6
IT	hyde (short)	e5	42	0,515	0,81	10
IT	hyde (short)	mistral	45	0,526	0,87	7
IT	hyde (medium)	e5	44	0,532	0,85	8
IT	hyde (medium)	mistral	46	0,515	0,88	6
IT	hyde (long)	e5	41	0,484	0,79	11
IT	hyde (long)	mistral	46	0,526	0,88	6
OIL	hyde (unlimited)	e5	40	0,528	0,77	12
OIL	hyde (unlimited)	mistral	46	0,562	0,88	6
OIL	hyde (short)	e5	43	0,541	0,83	9
OIL	hyde (short)	mistral	46	0,408	0,88	6
OIL	hyde (medium)	e5	46	0,571	0,88	6
OIL	hyde (medium)	mistral	46	0,505	0,88	6
OIL	hyde (long)	e5	44	0,571	0,85	8
OIL	hyde (long)	mistral	46	0,532	0,88	6
LAW	hyde (unlimited)	e5	30	0,308	0,58	22
LAW	hyde (unlimited)	mistral	37	0,382	0,71	15
LAW	hyde (short)	e5	28	0,261	0,54	24
LAW	hyde (short)	mistral	33	0,29	0,63	19
LAW	hyde (medium)	e5	30	0,227	0,58	22
LAW	hyde (medium)	mistral	35	0,318	0,67	17
LAW	hyde (long)	e5	33	0,256	0,63	19
LAW	hyde (long)	mistral	40	0,321	0,77	12

Заключение

В проведенном исследовании была проанализирована эффективность различных подходов к построению систем на базе технологий retrieval augmented generation (RAG) для работы с текстами на русском языке. Были рассмотрены несколько базовых методов RAG, включая наивный RAG, HyDE и BM25, и проведена их оценка по метрикам качества с использованием метрики mean average precision (mAP).

Основные выводы исследования можно сформулировать следующим образом:

1. Наивный RAG: Этот метод продемонстрировал стабильные результаты, особенно в сочетании с векторными моделями, такими как mistral и e5. Например, для набора данных «нефтегазовая промышленность» наивный RAG с моделью e5 показал 48 правильных документов из 53 возможных при 20 документах в поисковой выдаче с mAP@20, равным 0,677. Это подтверждает его эффективность для задач поиска и генерации ответов на русском языке.

2. HyDE: Метод HyDE, основанный на генерации гипотетических ответов, показал различные результаты в зависимости от длины сгенерированных ответов и используемых векторных моделей. В некоторых случаях HyDE превосходил наивный RAG, особенно при использовании модели mistral и длинных гипотетических ответов. Например, для набора данных «нефтегазовая промышленность» HyDE с моделью mistral показал 46 правильных документов из 53 возможных при 20 документах в поисковой выдаче с mAP@20, равным 0,562. Это свидетельствует о потенциале HyDE для улучшения качества ответов при правильной настройке параметров.

3. BM25: Традиционный метод BM25 также показал высокие результаты, особенно в предметной области «нефтегазовая промышленность». Например, BM25 показал 49 правильных документов из 53 возможных при 20 документах в поисковой выдаче с mAP@20, равным 0,737. Это делает его конкурентоспособным по сравнению с более современными методами.

Результаты исследования подчеркивают важность выбора подходящей векторной модели и метода ранжирования для достижения оптимальных результатов в системах RAG. Наивный RAG и BM25 могут служить надежной основой для разработки эффективных систем, в то время как HyDE предлагает перспективные возможности для улучшения качества ответов при дальнейшей оптимизации.

Таким образом, для достижения наилучших результатов в задачах генерации ответов на русском языке рекомендуется использовать гибридные подходы, сочетающие преимущества различных методов RAG. Это позволит создать более точные и контекстуально релевантные системы, способные эффективно работать с русскоязычными данными.

Список литературы/References

1. Gao Y., Xiong Y., Gao X., Jia K., Pan J., Bi Y., Dai Y., Sun J., Wang H. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*. 2023.

2. Li H., Su Y., Cai D., Wang Y., Liu L. A survey on retrieval-augmented text generation. *arXiv preprint arXiv:2202.01110*. 2022.

3. Cai D., Wang Y., Liu L., Shi S. Recent advances in retrieval-augmented text generation. In: *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. 2022. P. 3417–3419. DOI: 10.1145/3477495.3532682

4. Ma X., Gong Y., He P., Zhao H., Duan N. Query rewriting for retrieval-augmented large language models. *arXiv preprint arXiv:2305.14283*. 2023.

5. Gao L., Ma X., Lin J., Callan J. Precise zero-shot dense retrieval without relevance labels. *arXiv preprint arXiv:2212.10496*. 2022.

6. Wang L., Yang N., Huang X., Jiao B., Yang L., Jiang D., Majumder R., Wei F. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*. 2022.

7. Xiao S., Liu Z., Zhang P., Muennighof N. C-pack: packaged resources to advance general Chinese embedding. *arXiv preprint arXiv:2309.07597*. 2023.

8. Li X., Li J. Angle-optimized text embeddings. *arXiv preprint arXiv:2309.12871*. 2023.

9. VoyageAI, Voyage's embedding models. Available at: https://docs.voyageai.com/embeddings (accessed 30.01.2025).

10. BAAI, Flagembedding. Available at: https://github.com/FlagOpen/FlagEmbedding (accessed 30.01.2025).

11. Yoran O., Wolfson T., Ram O., Berant J. Making retrieval-augmented language models robust to irrelevant context. *arXiv preprint arXiv:2310.01558*. 2023.

12. Yu W., Zhang H., Pan X., Ma K., Wang H., Yu D. Chain-of-note: Enhancing robustness in retrieval-augmented language models. *arXiv preprint arXiv:2311.09210*. 2023.

13. Baek J., Jeong S., Kang M., Park J.C., Hwang S.J. Knowledge-augmented language model verification. *arXiv preprint arXiv:2310.12836*. 2023.

14. Cuconasu F., Trappolini G., Siciliano F., Filice S., Campagnano C., Maarek Y., Tonellotto N., Silvestri F. The power of noise: Redefining retrieval for rag systems. *arXiv preprint arXiv:2401.14887*. 2024.

15. Shao Z., Gong Y., Shen Y., Huang M., Duan N., Chen W. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. *arXiv preprint arXiv:2305.15294*. 2023.

16. Ke Z., Kong W., Li C., Zhang M., Mei Q., Bendersky M. Bridging the preference gap between retrievers and llms. *arXiv preprint arXiv:2401.06954*. 2024.

17. Berchansky M., Izsak P., Caciularu A., Dagan I., Wasserblat M. Optimizing retrieval-augmented reader models via token elimination. *arXiv preprint arXiv:2310.13682*. 2023.

18. Izacard G., Lewis P., Lomeli M., Hosseini L., Petroni F., Schick T., Dwivedi-Yu J., Joulin A., Riedel S., Grave E. Few-shot learning with retrieval augmented language models. *arXiv preprint arXiv:2208.03299*. 2022.

19. Harman D. *Information retrieval evaluation*. Morgan & Claypool Publishers; 2011.

20. Saraiva T., Sousa M., Vieira P., Rodrigues A. Telco-DPR: A Hybrid Dataset for Evaluating Retrieval Models of 3GPP Technical Specifications. *arXiv preprint arXiv:2410.19790*. 2024.

21. Es S., James J., Espinosa-Anke L., Schockaert S. Ragas: Automated evaluation of retrieval augmented generation. *arXiv preprint arXiv:2309.15217*. 2023.

22. Edwards C. Hybrid Context Retrieval Augmented Generation Pipeline: LLM-Augmented Knowledge Graphs and Vector Database for Accreditation Reporting Assistance. *arXiv preprint arXiv:2405.15436*. 2024.

23. RAGAS Library. Section. Description of evaluation metrics. Available at: https://docs.ragas.io/en/stable/concepts/metrics/index.html (accessed 30.01.2025).

24. LangChain Library. Available at: https://www.langchain.com/ (accessed 30.01.2025).

25. Wang L., Yang N., Huang X., Yang L., Majumder R., Wei F. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*. 2023.

26. Wang L., Yang N., Huang X., Yang L., Majumder R., Wei F. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*. 2024.

27. Massive Text Embedding Benchmark. Available at: https://huggingface.co/spaces/mteb/leaderboard (accessed 30.01.2025).

28. Douze M., Guzhva A., Deng C., Johnson J., Szilvasy G., Mazaré P.E., Jégou H. The faiss library. *arXiv preprint arXiv:2401.08281*. 2024.

29. RecursiveCharacterTextSplitter method from langchain library. Available at: https://api.python.langchain.com/en/latest/character/langchain_text_splitters.character.RecursiveCharacterTextSplitter.html (accessed 30.01.2025).

30. Gao, L., Ma, X., Lin, J., & Callan, J. Precise zero-shot dense retrieval without relevance labels. *arXiv preprint arXiv:2212.10496*. 2022.

31. Robertson S.E., Walker S. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: *SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University*. Springer, London; 1994. P. 232–241.

Информация об авторах

Мельников Андрей Витальевич, д-р техн. наук, проф., директор, Югорский научно-исследовательский институт информационных технологий, Ханты-Мансийск, Россия; MelnikovAV@uriit.ru.

Николаев Иван Евгеньевич, старший преподаватель кафедры информационных технологий и экономической информатики, Челябинский государственный университет, Челябинск, Россия; ivan_nikolaev@csu.ru.

Русанов Михаил Александрович, старший преподаватель инженерной школы цифровых технологий, Югорский государственный университет, Ханты-Мансийск, Россия; RusanovMA@uriit.ru.

Аббазов Валерьян Ринатович, ведущий программист, Югорский научно-исследовательский институт информационных технологий, Ханты-Мансийск, Россия; AbbazovVR@uriit.ru.

Information about the authors

Andrey V. Melnikov, Dr. Sci. (Eng.), Prof., Director, Ugra Research Institute of Information Technologies, Khanty-Mansiysk, Russia; MelnikovAV@uriit.ru.

Ivan E. Nikolaev, Senior Lecturer of the Department of Information Technologies and Economic Informatics, Chelyabinsk State University, Chelyabinsk, Russia; ivan_nikolaev@csu.ru.

Mikhail A. Rusanov, Senior Lecturer of the Engineering School of Digital Technologies, Yugra State University, Khanty-Mansiysk, Russia; RusanovMA@uriit.ru.

Valerian R. Abbazov, Lead Programmer, Ugra Research Institute of Information Technologies, Khanty-Mansiysk, Russia; AbbazovVR@uriit.ru.

Вклад авторов: все авторы сделали эквивалентный вклад в подготовку публикации.
Авторы заявляют об отсутствии конфликта интересов.
Contribution of the authors: the authors contributed equally to this article.
The authors declare no conflicts of interests.

Статья поступила в редакцию 17.02.2025
The article was submitted 17.02.2025