УДК 004.8

# ОШИБОЧНЫЕ ВЫВОДЫ МОДЕЛИ, ИСКАЖЕНИЯ И УСТАРЕВАНИЕ ЗНАНИЙ В RAG-СИСТЕМАХ

Васильев Е.В., аспирант
Прокопенко Е.В., к.ф.-м.н., доцент, заведующий кафедрой
Кузбасский государственный технический университет имени Т.Ф. Горбачева
Россия, г. Кемерово

Быстрое развитие больших языковых моделей (LLM) и распространение архитектуры RAG расширили возможности применения интеллектуальных систем в корпоративных и управленческих задачах. Использование внешних данных позволяет повысить точность ответов и снизить количество фактических ошибок, что делает RAG востребованным инструментом анализа и поддержки принятия решений.

Однако RAG-системы сохраняют ряд ограничений, связанных с ошибочными выводами моделей, информационными искажениями и устареванием знаний. Даже при наличии внешних источников данных возможны недостоверные ответы, если поисковая подсистема извлекает нерелевантные или устаревшие сведения. Исследования показывают, что добавление механизма извлечения снижает риск галлюцинаций, но не устраняет его полностью, а качество результатов зависит от точности поиска и актуальности базы знаний [1; 2; 3].

Эти ошибки могут искажать корпоративные знания, приводить к неверным управленческим решениям и снижать эффективность организационных процессов. В этой связи анализ причин ошибок RAG, факторов искажений и механизмов устаревания данных становится важной научной и практической задачей. Цель статьи – обобщить современные исследования, связанные с проблемами достоверности выводов, ошибками извлечения информации и вопросами актуальности знаний в RAG-системах [3; 4].

Подход RAG возник как ответ на фундаментальные ограничения больших языковых моделей, связанных с ограниченностью встроенных знаний, отсутствием механизмов актуализации и склонностью к генерации недостоверных утверждений. В отличие от классических LLM, которые опираются исключительно на параметрическую память и статистические зависимости в обучающих данных, RAG-системы используют внешний источник знаний, извлекая релевантную информацию во время выполнения запроса [1].

Архитектура RAG традиционно разбивается на два ключевых компонента: извлечение информации и генерация ответа.

На этапе извлечения информации входной запрос пользователя преобразуется в векторное представление, после чего выполняется поиск наиболее релевантных документов или фрагментов в базе знаний. В качестве репозитория могут использоваться: большие коллекции текстовых данных, корпоративные документы, базы данных, векторные индексы, построенные с помощью TF-

IDF, BM25, BERT-эмбеддингов или специализированных моделей (в частности Sentence-Transformers).

Качество этапа извлечения информации в значительной степени определяет итоговую точность вывода: ошибки индексации, неудачное разбиение текста на фрагменты, устаревшие сведения или недостаточно точное ранжирование приводят к подаче нерелевантного контекста на вход LLM [2; 5].

На втором этапе модель получает исходный запрос вместе с найденными фрагментами и формирует итоговый ответ с опорой на внешний контекст. В отличие от параметрического вывода, где модель опирается на вероятностные зависимости, RAG позволяет использовать проверяемые внешние данные. Такой механизм существенно снижает вероятность ошибочных выводов и повышает достоверность ответов, что подтверждается исследованиями и корпоративными отчётами Microsoft и Google [3; 6].

Основные преимущества RAG-систем, выделяемые в литературе [1]:

– актуальность знаний: внешняя база данных может обновляться независимо от модели;

– повышение точности: модель получает проверяемый контекст, что снижает вероятность ошибочных выводов;

– прозрачность и интерпретируемость. Можно проследить, какие документы использованы;

– модульность: возможность заменить поисковую систему, индекс или саму LLM без полной переработки архитектуры.

Несмотря на преимущества, подход имеет и значимые слабые места [1]:

– ошибки извлечения (извлечение информации bias): модель может получить нерелевантный или неполный контекст;

– зависимость от качества базы знаний: устаревшие документы автоматически ухудшают качество вывода;

– искажения ранжирования: высокие значения сходства не всегда означают фактическую релевантность;

– сложность настройки: качество работы системы существенно зависит от выбора параметров – модели для формирования векторных представлений, алгоритма поиска, а также способа разбиения документов на фрагменты;

– современные исследования показывают, что архитектуры типа GraphRAG или HybridRAG частично решают эти проблемы за счёт встроенной структуризации и многоуровневого анализа документа, однако базовые ограничения процесса извлечения данных остаются [2; 5].

Большие языковые модели продолжают демонстрировать склонность к формированию недостоверных или логически ошибочных ответов даже при детерминированных режимах генерации. Причины таких ошибок связаны с ограниченностью параметрических знаний, отсутствием механизмов проверки фактов и ошибочной интерпретацией контекста [4; 6]

Использование RAG снижает частоту подобных ошибок за счёт подключения внешних источников данных, однако полностью проблему не устраняет. Качество ответа зависит от полноты извлечённого контекста: если извлечение

информации возвращает неполные, нерелевантные или устаревшие фрагменты, модель формирует выводы, которые выглядят корректно, но фактически содержат ошибки [2; 3].

Искажения в RAG-системах возникают по двум основным направлениям: на этапе извлечения информации и на этапе последующей генерации ответа. Ошибки поиска проявляются в виде нерелевантных результатов, избыточного дробления текста, недостаточной точности векторных моделей и несоответствия между метрикой сходства и фактической релевантностью документов. Ошибки генерации, в свою очередь, возникают при неправильной интерпретации контекста, смешении разнородных фрагментов или выборочном использовании данных. Эти эффекты усиливаются при наличии противоречивых сведений или отсутствии ключевой информации в базе знаний [1; 5].

Значительный вклад в искажения вносит устаревание знаний. Поскольку механизм извлечения опирается на внешнюю базу данных, своевременное обновление которой не всегда обеспечивается, модель может получать фрагменты, не соответствующие текущей предметной области. Это приводит к формированию неполных или некорректных ответов, искажению корпоративных знаний и ошибкам в ранжировании найденных фрагментов. Проблема наиболее выражена в корпоративных системах, где информация обновляется неравномерно и значительно быстрее устаревает [7; 8].

Ошибки и искажения данного рода оказывают прямое влияние на организационные процессы. Недостоверные ответы могут приводить к снижению качества аналитики, нарушать корректность обработки нормативной или технической информации, ухудшать работу систем поддержки принятия решений. Особенно уязвимы области, связанные с документооборотом, техническим обслуживанием, обучением персонала и управлением корпоративными знаниями. Накопление ошибочных выводов формирует эффект «цепного искажения», когда неточности в retrieved-контенте последовательно воспроизводятся и усиливаются на последующих этапах обработки информации.

Таким образом, надёжность RAG-систем определяется не только возможностями языковой модели, но и состоянием базы знаний, качеством механизмов извлечения и устойчивостью организационных процессов к ошибкам обработки данных [3; 5; 8].

Снижение ошибок в RAG-системах требует комплексного подхода, охватывающего этапы извлечения информации, генерации ответа и обслуживание базы знаний. Современные исследования показывают, что наибольшее влияние на качество оказывает корректная работа поисковой подсистемы и поддержание актуальности данных [2; 1; 3; 5].

Одним из ключевых направлений является повышение качества извлечения информации. Наиболее эффективными считаются использование современных моделей векторизации текста, учитывающих контекст (например, BERT-производные), применение многоэтапного поиска с повторным ранжированием результатов и корректное разбиение документа на смысловые части. Гибридные методы, объединяющие классические алгоритмы поиска (BM25) с

векторными моделями, позволяют существенно повысить точность извлечения и снизить вероятность получения нерелевантного контекста [8].

Второе направление связано с повышением достоверности генерации. Для снижения ошибок интерпретации используются специальные формулировки подсказок, ограничивающие модель рамками предоставленного контекста, а также методы контроля, требующие включения ссылок на использованные фрагменты. Исследования OpenAI и DeepMind показывают, что применение цепочек рассуждений оправдано только при наличии проверяемой информации, поскольку в противном случае такие механизмы усиливают искажения. Более строгие режимы генерации позволяют уменьшить вариативность, но сами по себе не устраняют ошибок [4; 6].

Третьим направлением является обеспечение актуальности базы знаний. Регулярное обновление документов, переиндексация и мониторинг изменений позволяют снизить влияние устаревших сведений. В крупных корпоративных системах применяются такие меры, как удаление устаревших фрагментов и управление версиями контента, что предотвращает накопление ошибок и повышает достоверность извлечённых данных [7; 3].

Не менее важным остаётся постоянный контроль качества работы системы.

К наиболее распространённым практикам относятся тестирование поиска на эталонных наборах запросов, проверка корректности ответов экспертами или вспомогательными средствами, регулярный анализ журналов ошибок и использование метрик полноты и точности извлечения. Эти меры позволяют своевременно выявлять проблемные места и обеспечивать устойчивость RAG-систем к накоплению системных ошибок [5].

Таким образом, минимизация ошибок RAG требует сочетания технических решений, связанных с улучшением поиска и генерации, и организационных процедур, направленных на поддержание актуальности базы знаний и регулярный мониторинг качества [2].

Системы Retrieval-Augmented позволяют повысить достоверность работы больших языковых моделей за счёт использования внешних источников данных, однако ошибки извлечения, искажения интерпретации и устаревание знаний продолжают существенно влиять на качество ответов. Анализ литературы показывает, что именно корректность поисковой подсистемы и актуальность базы знаний определяют надёжность RAG в практических задачах.

Для эффективного применения RAG необходим комплексный подход: регулярное обновление данных, оптимизация механизмов извлечения информации, контроль корректности генерации и постоянный мониторинг качества системы. При соблюдении этих условий RAG способен значительно повысить устойчивость интеллектуальных решений и качество информационного обеспечения в организационных системах.

Список литературы:

1. Науменко А. О. Технология RAG (Retrieval-Augmented Generation) как инновационный подход в LLM // Вестник науки. 2025. Т. 5. № 8. С. 280–289.

2. Бойко А. Ю. Анализ эффективности интерактивных чат-ботов для работы с документами: сравнительный анализ базовой LLM, наивного RAG и метода GraphRAG // Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. 2025. № 4–2. С. 41–48.

3. Microsoft. Best Practices for Mitigating Hallucinations in Large Language Models (LLMs). 2025. URL: https://techcommunity.microsoft.com/blog/azure-ai-foundry-blog/best-practices-for-mitigating-hallucinations-in-large-language-models-llms/4403129 (дата обращения: 20.11.2025).

4. OpenAI. Why Language Models Hallucinate. 2025. URL: https://cdn.openai.com/pdf/d04913be-3f6f-4d2b-b283-ff432ef4aaa5.pdf (дата обращения: 20.11.2025).

5. NVIDIA. What Is Retrieval-Augmented Generation, aka RAG. 31.01.2025. URL: https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/ (дата обращения: 20.11.2025)

6. DeepMind. 2023: A Year of Groundbreaking Advances in AI and Computing. 22.12.2023. URL: https://deepmind.google/discover/blog/2023-a-year-of-groundbreaking-advances-in-ai-and-computing/ (дата обращения: 20.11.2025).

7. Хафизов Д. М. Интеллектуальный ассистент на основе LLM и RAG для поддержки принятия решений в корпоративной среде // Advances in Science and Technology. Труды LXVIII международной научно-практической конференции. Москва, 2025. С. 293–297.

8. Yandex Cloud. RAG: учим искусственный интеллект работать с новыми данными // Блог Yandex Cloud. 19.05.2025. URL: https://yandex.cloud/ru/blog/posts/2025/05/retrieval-augmented-generation-basics (дата обращения: 20.11.2025).