

ВАРИАНТЫ АССИСТИРОВАННОГО ПОИСКА ИНФОРМАЦИИ В ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ С ПРИМЕНЕНИЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Жарко Е.Ф., Промыслов В.Г., Семенков К.В., Степанов В.Н., Шумов А.С.

Институт проблем управления им. В.А. Трапезникова РАН, Москва, Россия

poletik@inbox.ru, vp@ipu.ru, semenkovk@ipu.ru, vnstepanov@yandex.ru

Аннотация. В работе приведены результаты анализа и экспериментального тестирования различных вариантов организации ассистированного поиска информации в технической документации на примере библиотеки документов по АСУ ТП АЭС. Проведена экспертная оценка эффективности поиска с применением облачных больших языковых моделей (БЯМ), в том числе с использованием специальных поисковых настроек, локальных дистиллированных БЯМ (с различными вариантами встраивания библиотеки документов), классическими поисковыми системами, а также ручным контекстным поиском. Рассматриваются вопросы безопасности применения искусственного интеллекта на АЭС.

Ключевые слова: ИИ, БЯМ, AI, LLM, искусственный интеллект, языковая модель, поисковая система, индексация, поиск в локальных документах, RAG, Graph RAG, local document embedding.

Введение

Применение систем отдельных алгоритмов, которые сейчас принято относить к искусственному интеллекту (ИИ) [1] для управления объектами, в частности для атомных станций (АЭС), имеет долгую историю [2]. И хотя они показали себя полезными инструментами для таких задач, как диагностика неисправностей, обеспечение кибербезопасности, поддержка оператора и т.д., есть и обоснованные трудности их применения. Прежде всего данные трудности связаны с проблемами верификации и валидации систем ИИ [3], объяснимости полученных в ходе их функционирования результатов [4]. Несмотря на известные трудности системы ИИ находят свое место в структуре АСУ ТП, что отражено в литературе и принимаемой их классификации для безопасности [5, 6].

Современные промышленные объекты, такие как АЭС, характеризуются огромными массивами технической документации. Оперативный и точный поиск информации в таких библиотеках критически важен для обеспечения безопасности, эффективности и бесперебойной работы оборудования. Однако традиционные методы поиска, такие как ручной контекстный поиск оказываются недостаточно эффективными, т.к. требуют высокой квалификации оператора и углубленного знания им предметной области. В этой связи актуальным становится поиск решений для построения систем ассистированного поиска информации, позволяющих получать релевантные результаты даже в условиях нечетких формулировок. В данной работе исследуются различные подходы к организации интеллектуального поиска на примере фрагмента технической документации АСУ ТП АЭС, оценивается их эффективность и безопасность.

1. Виды поисковых ассистентов и варианты их реализации в АСУ ТП АЭС

Поиск в локальной библиотеке технической документации может быть реализован с использованием различных подходов, начиная от простых методов и заканчивая сложными системами с применением искусственного интеллекта [7, 8].

Наиболее простой метод, основанный на индексировании документов и поиске по точному или частичному совпадению запроса с текстом. Этот вариант наименее гибкий, хоть и поддерживает регулярные выражения и маски, не учитывает семантику поисковых запросов. Поэтому метод очень чувствителен к точности формулировки запроса и требует высокой квалификации и знания предметной области оператором.

Более сложный поисковый ассистент – это классическая поисковая система (например Elasticsearch [9]), которая предварительно индексирует библиотеку данных и поддерживает семантический анализ поискового запроса, что делает ее менее чувствительной к точности совпадений фрагментов текста в запросе.

В настоящее время большое распространение получили системы так называемого искусственного интеллекта на базе больших языковых моделей (БЯМ) [10-12]. Эти системы представляют собой нейронные сети, обученные на очень большом объеме текстовой информации. Такие модели можно использовать для поиска информации в локальной библиотеке данных, которые не использовались при обучении модели и не известны ей. Для этого используется предварительная векторизация данных (RAG [13], Graph RAG [14]). Затем, по близости векторов, осуществляется поиск наиболее релевантных

фрагментов документа и эти фрагменты передаются в БЯМ вместе с запросом для формирования окончательного ответа.

Многие полноразмерные БЯМ находятся в открытом доступе и могут быть запущены локально, в изолированной среде. Но они требуют огромных вычислительных ресурсов и объемов оперативной памяти, для обеспечения приемлемого времени реакции. Для уменьшения требования к ресурсам применяют процесс дистилляции данных. В этом случае создается небольшая модель, обученная не на реальных данных, а на выводах другой полноразмерной БЯМ. Такие дистиллированные модели работают быстро и могут запускаться на незначительных вычислительных мощностях, но имеют пониженную точность выводов.

Существуют облачные поисковые системы, такие как perplexity [15], поддерживающие автоматический подбор наиболее подходящей для запроса БЯМ и подключение локальной библиотеки документов. Это наиболее совершенные поисковые системы на данный момент, но их и другие облачные БЯМ невозможно установить в локальной изолированной среде, что не позволяет их использовать на объектах с повышенным требованием к безопасности.

2. Эксперимент по проверке эффективности поиска с применением различных ассистирующих систем в локальной библиотеке документов

В данном исследовании была проведена предварительная экспертная оценка потенциальной эффективности разных вариантов реализации поисковых ассистентов. Оценка проводилась субъективно, т.к. получить объективную оценку текстового ответа затруднительно. Оценка проводилась в 2 этапа. На первом этапе поиск осуществлялся в одном документе «Автоматизированная система управления технологическими процессами. Система верхнего (блочного) уровня. Руководство оператора-технолога. Учебное пособие». В Таблице 1 приведены результаты этой оценки. На втором этапе поиск производился в библиотеке из 22 документов раздела «Автоматизированная система управления технологическими процессами. Программное обеспечение системы подготовки данных для автоматизированного внесения изменений в прикладное программное обеспечение СВБУ, СВСУ, СРВПЭ (ПО СПД)».

Таблица 1. Экспертная оценка точности, скорости и ресурсоёмкости различных реализаций ассистированного поиска информации в документе «Автоматизированная система управления технологическими процессами. Система верхнего (блочного) уровня. Руководство оператора-технолога. Учебное пособие»

Инструмент ассистента поиска	Точность ответа	Скорость нахождения ответа	Ресурсоемкость реализации
Облачные БЯМ нового поколения (Claude 3.5 sonnet, gpt4o, Gemini 2.5 pro)	хорошо	хорошо	плохо (невозможно)
Поисковая система на основе ИИ (БЯМ с надстройкой) (perplexity ai)	хорошо	Хорошо	плохо (невозможно)
Локальные дистиллированные БЯМ + RAG (llama 3.1, Gemma3, deepseek, qwen)	удовлетворительно	Удовлетворительно	удовлетворительно
Локальные дистиллированные БЯМ + Graph RAG (llama 3.1, Gemma3, deepseek, qwen)	удовлетворительно	Удовлетворительно	удовлетворительно
Классическая поисковая система (Google, Yandex, Elasticsearch)	хорошо	отлично	Хорошо

Инструмент ассистента поиска	Точность ответа	Скорость нахождения ответа	Ресурсоемкость реализации
Ручной контекстный поиск (оператор средней квалификации)	отлично	Хорошо	Отлично

В эксперименте участвовали следующие облачные БЯМ: Anthropic Claude 3.5 Sonnet, OpenAI GPT 4o, Google Gemini 2.5 Pro. Работа с моделями осуществлялась с помощью сервиса Coze, предварительная обработка документов осуществлялась с использованием иерархической сегментации, наиболее подходящий для структурированной технической документации. Поисковый сервис Perplexity AI применялся в режиме поиска с автоматическим выбором наиболее подходящей модели. Для эксперимента с локальными дистиллированными моделями применялась оболочка Nomic AI GPT4All, поддерживающая настраиваемую RAG векторизацию локальных данных с помощью модели Nomic Embed Text. Подключались следующие модели: Llama 3.1 8b, Qwen 3 8b, Qwen 3 14b, Gemma 3 4b, Gemma 3 12b, Deepseek r1 8b, Deepseek r1 14b. Для тестирования классических поисковых систем набор документов был размещен на временном хостинге, после чего осуществлялся поиск с ограничением по расположению. Для эксперимента с ручным поиском был выбран сотрудник, не знакомый с данной предметной областью, но являющийся техническим специалистом. Ему была предоставлена папка с документами и список вопросов. Оценка результатов поиска в библиотеке документов представлена в Таблице 2.

Таблица 2. Экспертная оценка точности, скорости и ресурсоемкости различных реализаций ассистированного поиска информации в локальной библиотеке документов

Инструмент ассистента поиска	Точность ответа	Скорость нахождения ответа	Ресурсоемкость реализации
Облачные БЯМ нового поколения (Claude 3.5 sonnet, gpt4o, Gemini 2.5 pro)	удовлетворительно	хорошо	плохо (невозможно)
Поисковая система на основе ИИ (БЯМ с надстройкой) (perplexity ai)	хорошо	хорошо	плохо (невозможно)
Локальные дистиллированные БЯМ + RAG (llama 3.1, Gemma3, deepseek, qwen)	плохо	удовлетворительно	удовлетворительно
Локальные дистиллированные БЯМ + Graph RAG (llama 3.1, Gemma3, deepseek, qwen)	плохо	удовлетворительно	удовлетворительно
Классическая поисковая система (Google, Yandex, Elasticsearch)	удовлетворительно	отлично	хорошо
Ручной контекстный поиск (оператор средней квалификации)	отлично	плохо	отлично

Рост объема библиотеки документов приводит к снижению скорости нахождения информации при ручном поиске и к снижению точности ответов при использовании ассистентов на основе искусственного интеллекта.

3. Вопросы безопасности применения различных видов поисковых ассистентов на АЭС и других объектах критической инфраструктуры

Для удовлетворения требованиям безопасности, любая система поиска данных должна быть консультативной. Это значит, что решение по результатам поиска должен принимать оператор, а система должна предоставлять не только ответ на запрос, но и подтверждение правильности ответа в виде ссылок на документ и цитат из него. Второе важное требование безопасности – это защита системы от воздействия извне. Для удовлетворения этому требованию поисковая система должна иметь возможность работать автономно, в полностью изолированной среде. Этому требованию не удовлетворяют наиболее перспективные с точки зрения эффективности современные облачные системы. Уменьшенные (дистиллированные) варианты БЯМ, с другой стороны, на данный момент плохо справляются с задачей поиска информации в локальных документах, хоть и могут работать в изолированной среде и не требуют больших аппаратных ресурсов.

4. Заключение

По результатам данного исследования можно сделать следующие выводы:

- Крупномасштабные облачные БЯМ и поисковые системы на основе ИИ дают приемлемые результаты при обработке поисковых запросов к локальной библиотеке документов, но использование их на объектах с повышенным требованием к безопасности невозможно.
- Локальные дистиллированные БЯМ с подключением локального поискового контекста (RAG, Graph RAG) на данный момент работают непредсказуемо и с низкой точностью. С точки зрения безопасности перспектив у этого варианта больше, т.к. возможна реализация в полностью изолированной безопасной среде.
- Классическая поисковая система с морфологическим анализом запроса не способна интегрировать данные из нескольких источников, фактически является усовершенствованным и более гибким контекстным поиском. Требуется повышенной квалификации, углубленного знания предметной области. По совокупности характеристик это наиболее оптимальное решение поискового ассистента на данный момент.

Литература

1. ГОСТ Р. 59276-2020. 35. Системы искусственного интеллекта. Классификация систем искусственного интеллекта. ГОСТ Р. 59277-2020. 36.
2. *Лебедев Л.С.* Обзор экспертных систем и перспективы их применения в энергетике // Вестник ИргТУ. 2014. №4 (87). URL: <https://cyberleninka.ru/article/n/obzor-ekspertnyh-sistem-i-perspektivy-ih-primeneniya-v-energetike> (дата обращения 25.06.2025).
3. *Lalli Myllyaho, Mikko Raatikainen, Tomi Männistö, Tommi Mikkonen, Jukka K. Nurminen.* Systematic literature review of validation methods for AI systems // Journal of Systems and Software, 2021. – Vol. 181. – 111050. ISSN 0164-1212. DOI:10.1016/j.jss.2021.111050.
4. *Phillips P., Hahn C., Fontana P., Yates A, Greene K., Broniatowski D. and Przybocki M.* 818 (2021), Four Principles of Explainable Artificial Intelligence, NIST Interagency/Internal Report (NISTIR), 819 National Institute of Standards and Technology, Gaithersburg, MD, [online], 820 DOI:10.6028/NIST.IR.8312, https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=933399 821 (дата обращения 07.05.2025).
5. *Flehmig N., Lundteigen M.A. and Yin S.* Implementing Artificial Intelligence in Safety-Critical Systems during Operation: Challenges and Extended Framework for a Quality Assurance Process, IECON 2024 – 50th Annual Conference of the IEEE Industrial Electronics Society, Chicago, IL, USA, 2024. – P. 1–8. DOI: 10.1109/IECON55916.2024.10906021.
6. ISO/IEC TR 5469:2024. Artificial intelligence – Functional safety and AI systems.
7. AI-Enabled Search Assistant for Operating Manuals <https://mosaicdatascience.com/2023/02/10/ai-enabled-search-assistant-for-operating-manuals/> (дата обращения 25.06.2025).
8. Intelli Manual. <https://www.intellinetsystem.com/interactive-digital-manual&> (дата обращения 25.06.2025).
9. Elasticsearch. <https://www.elastic.co/docs/solutions/search> (дата обращения 25.06.2025).
10. Introducing Claude 4 <https://www.anthropic.com/news/claude-4> (дата обращения 26.05.2025).
11. What is Claude AI? <https://www.ibm.com/think/topics/claude-ai> (дата обращения 26.05.2025).
12. Gemini 2.5: Our most intelligent AI model. <https://blog.google/technology/google-deepmind/gemini-model-thinking-updates-march-2025/> (дата обращения 04.06.2025).

13. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Scott Yih, Tim Rocktäschel, Sebastian Riedel, Douwe Kiela
<https://proceedings.neurips.cc/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf> (дата обращения 04.06.2025).
14. Welcome to GraphRAG/ <https://microsoft.github.io/graphrag/> (дата обращения 04.06.2025).
15. Perplexity lets you search your internal enterprise files and the web. <https://venturebeat.com/ai/perplexity-lets-you-search-your-internal-enterprise-files-and-the-web/> (дата обращения 04.06.2025).