Я. А. Шенцов, Т. Ю. Чернышева
Тюменский государственный университет, г. Тюмень
УДК 004.85

РАЗРАБОТКА AI-АССИСТЕНТА ДЛЯ СЕЛЬСКОГО ХОЗЯЙСТВА

Аннотация. В данной статье проводится анализ и разработка AI-ассистента на базе большой языковой модели, направленные на повышение эффективности, автоматизации и оптимизации процессов в сельском хозяйстве. Исследование включает сравнительный анализ популярных больших языковых моделей, проектирование архитектуры AI-ассистента, а также определение подхода к реализации.
Ключевые слова: AI-ассистент, большие языковые модели, RAG, искусственный интеллект.

Введение. В современном мире, в различных областях жизни становится все более широким и разнообразным применение искусственного интеллекта. Одним из перспективных направлений в данной сфере является разработка AI-ассистентов. Популярным подходом в реализации AI-ассистентов является использование больших языковых моделей (LLM) [1, 2]. Данные модели способны обрабатывать и генерировать большие последовательности естественного языка, что делает их подходящими кандидатами для использования в различных архитектурах AI-ассистентов.

Проблема исследования. Существует ряд проблем, связанных с использованием языковых моделей в качестве основы для AI-ассистентов, например:

– высокие требования к вычислительным ресурсам. При неправильном выборе архитектуры большой языковой модели и недостатке вычислительных мощностей в момент использования модели можно столкнуться с проблемой медленного инференса, что может плохо повлиять на работу всего AI-ассистента;

– большой объем данных для обучения модели. При обучении с нуля или дообучении LLM необходим большой текстовый корпус, сборка, очистка и подготовка которого может быть довольно затруднительна;

– отсутствие гарантий безопасности и конфиденциальности данных. При использовании сторонних моделей таких, как GPT, есть риск утечки данных;

– отсутствие уверенности в правильности сгенерированного моделью текста. Зачастую модели могут быть обучены на ложных данных или данных, которые не содержат знания о необходимой предметной области. Следовательно, модель не сможет ответить на поставленный вопрос или ответит на него неверно.

Исходя из проблем, сформулирована цель исследования: разработка и реализация AI-ассистента, основанного на большой языковой модели для применения в сельском хозяйстве. Для достижения цели выделены следующие задачи:

– анализ существующих LLM и выбор оптимальной;

– исследование и анализ подходов реализации LLM, а также выбор подходящих под нужды и требования реализовываемого ассистента;

– проектирование архитектуры AI-ассистента;

– реализация AI-ассистента и чат-бота для взаимодействия с пользователем.

Материалы и методы. Первым этапом реализации AI-ассистента является анализ и выбор большой языковой модели. При выборе LLM важно учитывать несколько ключевых факторов: размер модели, ее архитектуру, количество параметров, величину контекстного

окна, возможность развертывания в отсутствии больших вычислительных и графических мощностей, скорость и качество работы.

Исходя из вышеперечисленных критериев был проведен сравнительный анализ наиболее популярных больших языковых моделей, таких как GPT, Mistral, Falcon, Llama и YaGPT. Результаты сравнительного анализа приведены в табл. 1.

Таблица 1

Сравнительный анализ популярных LLM

Архитектура модели	*Количество параметров*	*Размер контекстного окна*	*Доступность*	*Возможность развертывания на локальном ПК*
GPT-3	~175 млрд	2048	Закрытая	Нет
Mistral-7B	~7 млрд	4096	Открытый код	Да
YaGPT-13B	~13 млрд	-	Закрытая	Нет
Llama2-7B	**~7 млрд**	**4096**	**Открытый код**	**Да**
Falcon-7B	~7 млрд	2048	Открытый код	Да

В ходе сравнительного анализа выявлено, что наиболее подходящей LLM моделью для реализовываемого AI-ассистента является Llama2-7B [3], так как она имеет большое контекстное окно, открытый исходный код, возможность развернуть на локальном сервере, а также имеет небольшое количество параметров, что влияет на размер модели, но также и на качество сгенерированного текста [4].

В процессе анализа различных подходов к реализации AI-ассистента на базе LLM рассмотрено несколько методов:

1. Finetuning. Метод включает в себя дополнительное обучение предварительно обученной языковой модели на специфических задачах. Использование данного метода требует больших вычислительных ресурсов и времени на обучение [5].

2. Prompt-based generation. Подход заключается в использовании конкретных текстовых подсказок и инструкций для регулирования генерации ответов от модели. Он обладает преимуществом гибкости и контроля над выходными данными, но может требовать тщательной настройки [6].

3. Retrieval Augmented Generation (RAG). Подход комбинирует генерацию текста с поиском и выбором наиболее релевантных данных из предварительно индексированной векторной базы данных. RAG позволяет модели создавать ответы на основе контекста, полученного из базы данных. Это позволяет снизить риск непредсказуемых или некорректных ответов, типичных для генеративных моделей [7, 8].

Проанализировав подходы, выбрали prompt-based generation и RAG. Отказ от использования finetuning метода аргументирован отсутствием больших вычислительных ресурсов.

Следующий этап — проектирование архитектуры разрабатываемого ассистента. Архитектура Ассистента разделена на несколько основных блоков: Агент, RAG-модуль, инструменты. Использование Агентов в LLM является подходом, набирающим популярность. Благодаря Агентам у LLM появляется возможность принимать решения, выполнять действия и воспринимать окружающую среду [9, 10]. Инструмент выполняет задачи, которые назначаются Агентом. Так как инструменты работают с внутренними базами знаний, то они также

взаимодействуют с RAG-модулем, который в свою очередь состоит из модели эмбеддинга и ретривера. Архитектура разрабатываемого AI-ассистента представлена на рис. 1.

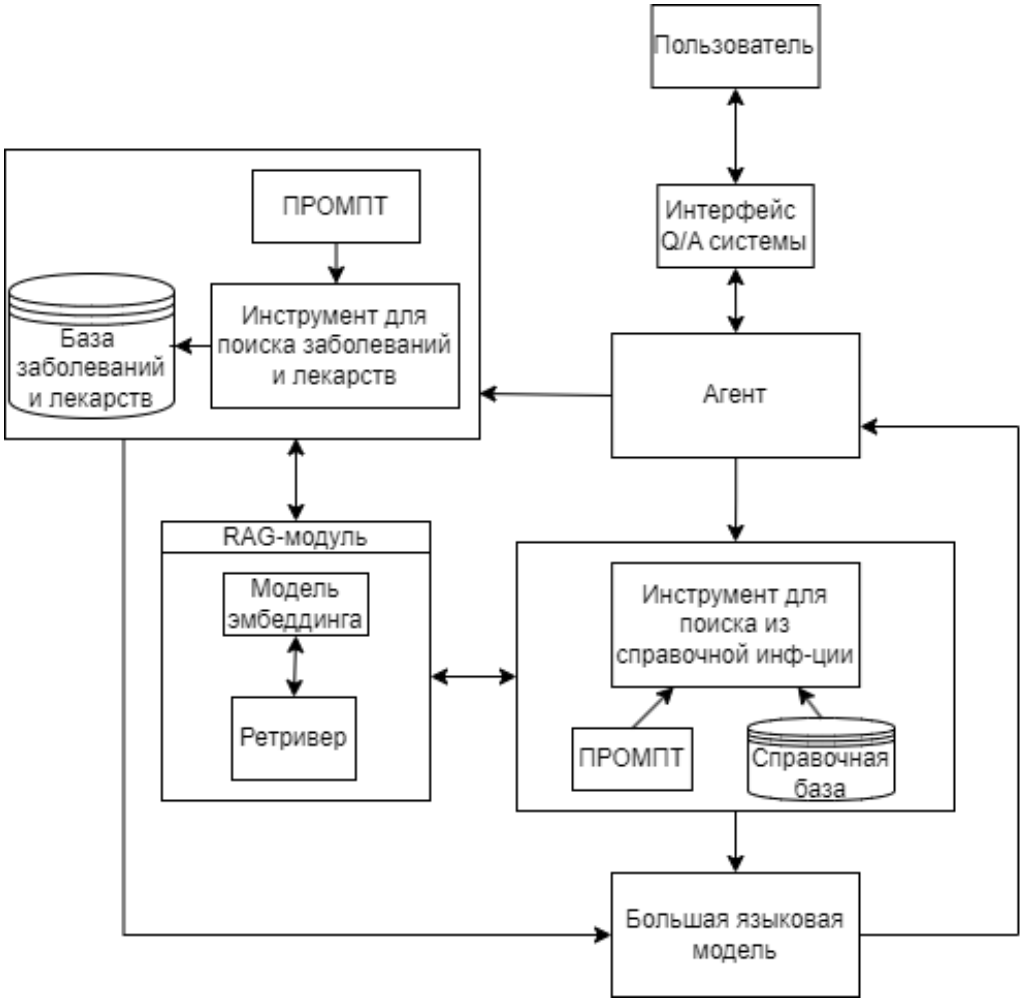

Рис. 1. Общая архитектура разрабатываемого AI-ассистента

При реализации AI-ассистента использованы следующие инструменты: язык программирования Python, библиотеки LangChain, Llama.cpp, Transformers, BM25, для векторной СУБД — Pgvector, для реализации чат-бота –Telebot.

Результаты. Результатом исследования стал реализованный AI-ассистент в виде чат-бота. Данный ассистент способен отвечать на запросы пользователя о сельскохозяйственных культурах: давать советы по уходу за растениями, информировать о запрашиваемых заболеваниях и способах их лечения, а также предоставлять рекомендации по оптимизации сельскохозяйственных процессов. AI-ассистент обладает базой знаний о различных сельскохозяйственных культурах, их особенностях, возможных болезнях и методах лечения. Пользователь может задавать вопросы о конкретных растениях, описывать наблюдаемые симптомы и получать рекомендации по уходу и лечению. Также в AI-ассистента внедрена функция анализа погодных условий и предоставления рекомендации по оптимальным моментам для полива, уборки урожая и других мероприятий. Примеры вопросов и рекомендаций Ассистента представлены на рис. 2.

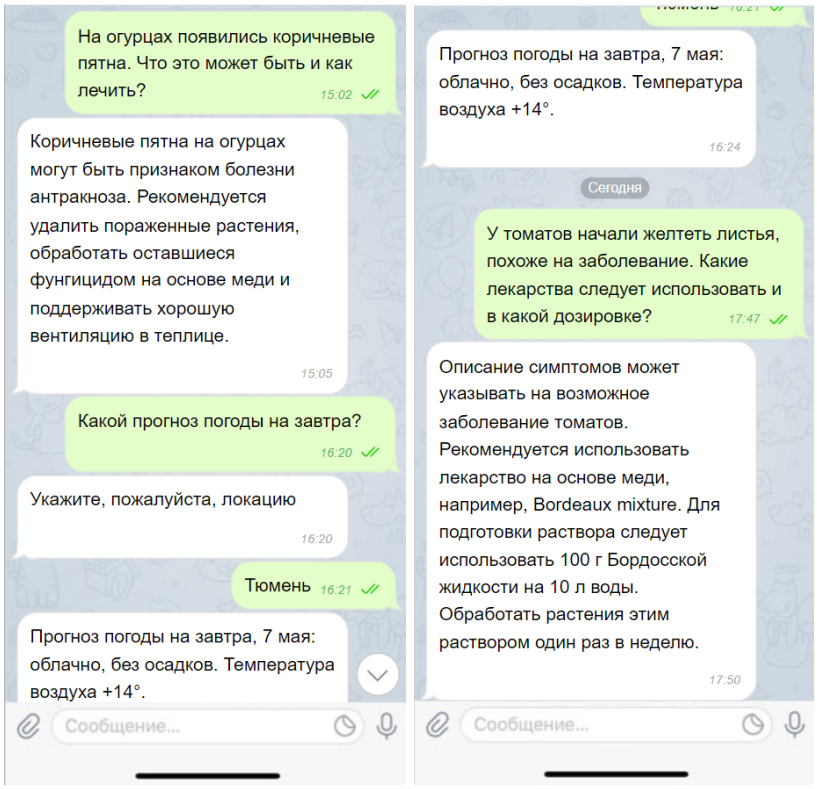

Рис. 2. Пример работы AI-ассистента

Заключение. В статье проанализирован ряд больших языковых моделей по выделенным критериям. Исследованы подходы к реализации и использованию LLM и приняты два из них. Данные подходы выбраны из-за гибкости и способности генерировать текст на основе контекста, что важно для обеспечения качественных ответов в рамках предметной области. Итогом стал разработанный AI-ассистент в виде чат-бота на платформе Telegram [11], который способен отвечать на запросы пользователей о сельскохозяйственных культурах, предоставлять советы, информировать о заболеваниях и способах их лечения, а также предлагать рекомендации по оптимизации сельскохозяйственных процессов. Планируется внедрение функции распознавания речи для ускорения взаимодействия пользователя с ассистентом [12]. Разработанный AI-ассистент существенно облегчает работу сельскохозяйственным производителям, ускоряя различные процессы, что помогает сэкономить затрачиваемое на поиск решения время.

СПИСОК ЛИТЕРАТУРЫ

1. Naveed H., Khan A.U., Qiu S. [et al.]. A comprehensive overview of large language models // arXiv preprint arXiv:2307.06435. — 2023.
2. Minaee S., Mikolov T., Nikzad N. [et al.]. Large language models: A survey // arXiv preprint arXiv:2402.06196. — 2024.
3. Touvron H., Lavril T., Izacard G. [et al.]. Llama: Open and efficient foundation language models // arXiv preprint arXiv:2302.13971. — 2023.
4. Alizadeh K., Mirzadeh I., Belenko D. [et al.]. Llm in a flash: Efficient large language model inference with limited memory // arXiv preprint arXiv:2312.11514. — 2023.

5. Zheng J., Hong H., Wang X. [et al.]. Fine-tuning Large Language Models for Domain-specific Machine Translation // arXiv preprint arXiv:2402.15061. — 2024.
6. Maity S., Deroy A., Sarkar S. Harnessing the Power of Prompt-based Techniques for Generating School-Level Questions using Large Language Models // Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation. — 2023. — С. 30-39.
7. Lin D. Revolutionizing Retrieval-Augmented Generation with Enhanced PDF Structure Recognition // arXiv preprint arXiv:2401.12599. — 2024.
8. Zhu Y., Yuan H., Wang Sh. [et al.]. Large language models for information retrieval: A survey // arXiv preprint arXiv:2308.07107. — 2023.
9. Xi Z., Chen W., Guo X. The rise and potential of large language model based agents: A survey // arXiv preprint arXiv:2309.07864. — 2023.
10. Huang X., Liu W., Chen X. Understanding the planning of LLM agents: A survey // arXiv preprint arXiv:2402.02716. — 2024.
11. Шенцов Я.А. Разработка архитектуры вопросно-ответной системы на базе большой языковой модели / Я. А. Шенцов // Прогрессивные технологии и экономика в машиностроении: сборник трудов XV Всероссийской научно-практической конференции для студентов и учащейся молодежи, Юрга, 11–13 апреля 2024 года / Национальный исследовательский Томский политехнический университет, Юргинский технологический институт. — Томск: Национальный исследовательский Томский политехнический университет, 2024. — С. 247-249.
12. Shentsov Y.A., Chernysheva T.Y., Barskaya G.B. Application of speech recognition and autoreference models for logging tasks // Third International Conference on Optics, Computer Applications, and Materials Science (CMSD-III 2023). — SPIE, 2024. — Т. 13065. — С. 7-13.