СИСТЕМНЫЙ АНАЛИЗ И УПРАВЛЕНИЕ
SYSTEM ANALYSIS AND PROCESSING OF KNOWLEDGE

УДК 004.89
DOI 10.52575/2687-0932-2024-51-3-699-709

Использование метода RAG и больших языковых моделей в интеллектуальных образовательных экосистемах

[1] Оболенский Д.М., [2] Шевченко В.И.
[1] ООО «Новые технологии», Россия, 299053, г. Севастополь, ул. Острякова, 98
[2] Севастопольский государственный университет,
Россия, 299053, г. Севастополь, ул. Университетская, 33
E-mail: denismaster@outlook.com, VIShevchenko@sevsu.ru

Аннотация. В статье рассматривается использование метода Retrieval-Augmented Generation (RAG) и больших языковых моделей в интеллектуальных образовательных экосистемах. Предложено использование больших языковых моделей для улучшения представления образовательных ресурсов, вакансий и предпочтений пользователей в рекомендательных системах. Рассмотрено применение метода RAB для дополнения знаний больших языковых моделей новыми данными без дополнительного обучения. В примере реализации в интеллектуальной образовательной экосистеме показано применение библиотеки langchain, языковой модели GigaChat и векторной СУБД Qdrant с использованием описаний вакансий и образовательных ресурсов для генерации дружелюбного для пользователя описания рынка труда в соответствии с его запросом.

Ключевые слова: RAG, LLM, интеллектуальная образовательная экосистема, большие языковые модели, python, langchain

Для цитирования: Оболенский Д.М., Шевченко В.И. 2024. Использование метода RAG и больших языковых моделей в интеллектуальных образовательных экосистемах. Экономика. Информатика, 51(3): 699–709. DOI 10.52575/2687-0932-2024-51-3-699-709

Application of Large Language Models and the RAG in Intelligent Educational Ecosystems

[1] Denis M. Obolensky, [2] Victoria I. Shevchenko
[1] New Technologies LLC
98 Ostryakova St, Sevastopol 299053, Russia
[2] Sevastopol State University,
33 Universitetskaya St, Sevastopol 299053, Russia
E-mail: denismaster@outlook.com, VIShevchenko@sevsu.ru

Abstract. The article discusses the usage of the Retrieval-Augmented Generation (RAG) algorithm and large language models in intelligent educational ecosystems. The authors demonstrate the ability of large language models to improve the representation of educational resources, vacancies and user preferences in recommendation systems. The application of the RAG algorithm to supplement the knowledge of large language models with new data without additional training is considered. The example of implementation in an intelligent educational ecosystem shows the use of the Langchain library, the GigaChat large language model and the Qdrant vector database with jobs and educational resources descriptions to generate a user-friendly description of the labor market in accordance with his request.

Keywords: RAG, LLM, intelligent educational ecosystem, large language models, python, Langchain

For citation: Obolensky D.M., Shevchenko V.I. 2024. Application of Large Language Models and the RAG in Intelligent Educational Ecosystems. Economics. Information technologies, 51(3): 699–709. DOI 10.52575/2687-0932-2024-51-3-699-709

Введение

Образовательные экосистемы, согласно исследованию Бабкина А.В и др. [Бабкин и др., 2022] – одно из наиболее многообещающих направлений развития дистанционного образования. Это комплексная среда, предоставляющая персонализированные образовательные ресурсы и услуги, которые позволяют эффективно поддерживать и совершенствовать образовательный процесс. Экосистема объединяет преподавателей, студентов, родителей, организации, работодателей и другие заинтересованные стороны в едином информационном, образовательном и технологическом пространстве.

В качестве дополнения и улучшения существующих моделей была предложена концепция интеллектуальной образовательной экосистемы (ИОЭ) [Оболенский, Шевченко, 2019; Оболенский, Шевченко, 2020]. Она представлена на рис. 1.

Рис. 1. Концептуальная модель ИОЭ
Fig. 1. Conceptual model of IEE

Предложенная ИОЭ включает в себя следующие элементы:

– существующую современную модель образовательной экосистемы, включающую системы управления обучением, возможности дистанционных курсов, образовательные материалы курсов и дополнительные ресурсы (книги, статьи, видео и т. д.);

– модель обучающегося, информацию о его интересах, возможностях и компетенциях;

– требования работодателя, представляющие текущее состояние рынка труда и запросы работодателей в виде информации о вакансиях;

– рекомендательную систему, связывающую вышеперечисленные элементы в единое образовательное пространство [Оболенский, Шевченко, 2019].

Рекомендательные системы на основе контента – это тип рекомендательной системы, которая предлагает некоторые элементы пользователям на основе характеристик других элементов, предпочитаемых пользователями [Оболенский, Шевченко, 2021]. В системе контентных рекомендаций элементы представлены в виде набора характеристик или атрибутов. Рекомендации составляются на основе метрики сходства между

характеристиками элементов и предпочтениями пользователей. Схема работы простейшей рекомендательной системы представлена на рис. 2.

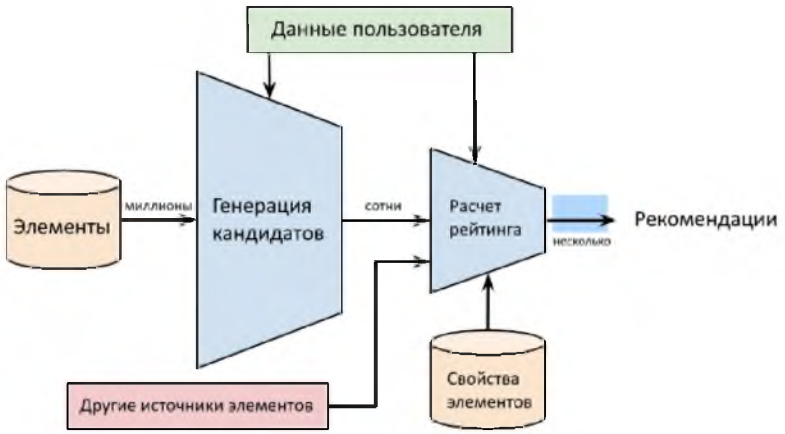

Рис. 2. Схема работы рекомендательной системы
Fig. 2. The scheme of the recommendation system

Большие языковые модели (БЯМ, Large Language Models, LLM) – это семейство мощных генеративных нейронных сетей на архитектуре «трансформер» [Vaswani et al, 2017], обученных на огромном объеме текстовых данных [Малышев, Смирнов, 2024]. Архитектура LLM представлена на рис. 3. Они используются для генерации текста, классификации и выполнения других задач обработки естественного языка. Примерами таких моделей являются BERT [Devlin, Chang, Toutanova, 2019], GPT-4 от компании OpenAI [Achiam et al, 2023], модель YandexGPT от компании Yandex [YandexGPT 3, 2024] и GigaChat от компании Сбер [Малышев, Смирнов, 2024; GigaChat, 2024].

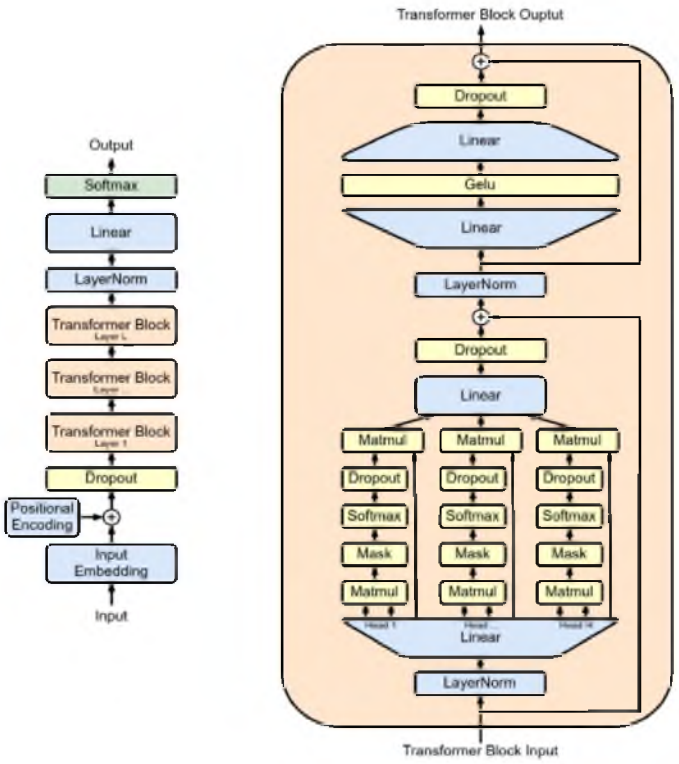

Рис. 3. Архитектура нейронной сети «Трансформер»
Fig. 3. The architecture of the neural network "Transformer"

Большие языковые модели могут использоваться в системах рекомендаций на основе контента для улучшения представления элементов и пользователей. LLM, такие как GPT-3, BERT или GigaChat также могут генерировать высококачественные векторные представления для некоторого текста с помощью специальных embedding-моделей. Эти векторные представления, также называемые embedding-векторами, улавливают семантическое значение текста, его контекст. Для представления элементов и пользователей в рекомендательных системах, например, в образовательных экосистемах, можно использовать текстовые описания элементов и предпочтений пользователя, а также embedding-вектора на их основе.

Важную роль в работе БЯМ, например, YandexGPT или GigaChat, играет prompt-сообщение [Zhou et al, 2022]. Prompt-сообщение подает на вход нейросети вместе с данными пользовательского ввода и используется моделью для генерации ответа. Prompt-сообщение непосредственно влияет на процесс генерации цепочки токенов [Zhou et al, 2022] при помощи алгоритма авторегрессии и влияет на содержание, стиль и структуру сгенерированных выходных данных.

При работе с БЯМ пользователи или разработчики системы обычно вводят некоторое системное prompt-сообщение, в котором указывается контекст, вопрос или задача, на которые они хотят, чтобы модель ответила. Prompt-сообщение может представлять собой одно предложение, абзац или даже более длинный фрагмент текста, в зависимости от сложности желаемого ответа. Модель использует информацию, предоставленную в prompt-сообщении, для генерации наиболее релевантного ответа, который соответствует заданным входным данным [Zhou et al, 2022].

С другой стороны, prompt-сообщения могут расширять контекст нейросети, дополняя ее новыми данными, на которых она не была обучена. Одним из наиболее широко используемых способов расширения контекста и знаний модели с использованием prompt-сообщений является метод Retrieval-Augmented Generation (RAG) [Gao et al, 2023; Lewis et al, 2020].

Использование метода RAG совместно с большими языковыми моделями

Метод RAG [Lewis et al, 2020] объединяет в себе поиск данных, расширение найденными данными контекста и генерацию. Данный подход был представлен исследователями в 2020 году для устранения ограничений традиционных БЯМ, таких как GPT-3, при решении сложных задач поиска информации.

При использовании метода RAG большая языковая модель, например, GigaChat, дополняется механизмом поиска, который позволяет ей получать доступ к внешним источникам знаний. Этот механизм поиска позволяет модели извлекать соответствующую информацию из базы знаний или корпуса документов перед генерированием ответа. Метод RAG основан на концепции few-shot learning [Brown et al, 2020]. Метод RAG используется в задаче улучшения способности модели генерировать релевантные, информативные и фактически точные ответы с помощью тех данных, на которых она не была обучена [Gao et al, 2023]. Схема работы алгоритма RAG представлена на рис. 4.

Важной особенностью данного подхода является отсутствие необходимости какого-либо обучения модели [Radford, 2019]. Так как большие языковые модели содержат миллиарды параметров, их обучение занимает значительное количество аппаратных, временных и финансовых ресурсов [Shoeybi et al, 2019]. При использовании метода RAG дополнительное обучение не требуется, достаточно просто дополнить модель новыми данными в стандартном prompt-сообщении.

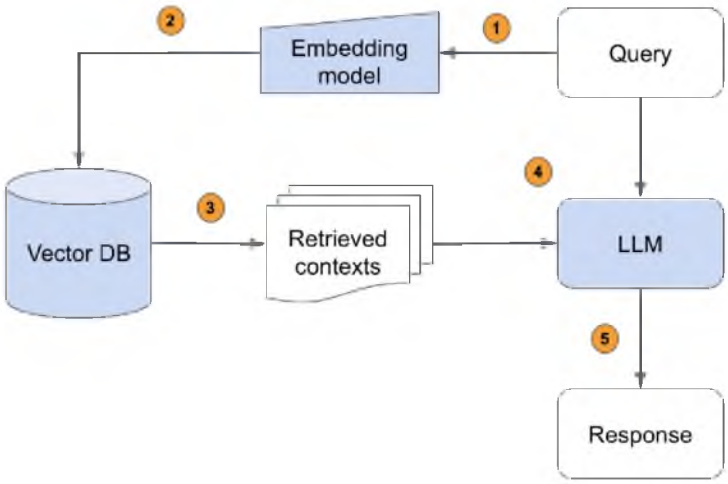

Рис. 4. Схема работы метода RAG
Fig. 4. The scheme of operation of the RAG method

Метод RAG устроен следующим образом (рис. 4):

1. пользователь вводит в систему некоторый запрос;
2. запрос пользователя конвертируется с помощью специальной embedding-модели в векторное представление (embedding-вектор) [Keraghel et al, 2024];
3. используется специальное хранилище embedding-векторов и метрики сходства для поиска наиболее схожих на запрос пользователя embedding-векторов;
4. исходный запрос пользователя, а также исходные контексты, связанные с найденными на шаге 3 embedding-векторами, используются как дополнительные prompt-сообщения в LLM, дополняя и расширяя контекст модели;
5. модель, используя дополненный новыми данными контекст, возвращает некоторый результат пользователю в виде дружелюбного текста.

Для реализации хранилища векторов часто используются дополнительные корпусы документов, базы знаний, векторные СУБД, например, Weaviate [Weaviate, 2024] или Qdrant [Qdrant, 2024], а также графовые базы данных, например, Neo4J [Neo4J, 2024].

Векторная база данных – это особый тип NoSQL-базы данных, оптимизированный для хранения и запроса многомерных числовых векторов [Qdrant, 2024]. Векторные базы данных предназначены для эффективного хранения, индексации и запроса векторных данных, что делает их хорошо подходящими для приложений, связанных с поиском сходства, кластеризацией, классификацией и другими задачами, требующими анализа многомерных данных и манипулирования ими. Эти базы данных обычно используются в таких областях, как машинное обучение, обработка естественного языка, обработка изображений, рекомендательные системы и многое другое. Каждая коллекция векторов связана с embedding-моделью и представляет собой некоторое векторное пространство [Keraghel et al, 2024] (рис. 5.).

Для реализации семантического поиска [Mikolov, 2013] векторов в векторных базах данных можно использовать различные метрики для измерения сходства [Shadab, Subhajit, 2020] между векторами, например:

1. **Косинусное расстояние**. Косинусная мера сходства измеряет косинус угла между двумя embedding-векторами. Значения данной метрики варьируются от -1 (полностью противоположное) до 1 (идентичное).
2. **Евклидово расстояние**. Евклидово расстояние вычисляет расстояние по прямой между двумя embedding-векторами в многомерном пространстве. Меньшие расстояния указывают на большее сходство [Shadab, Subhajit, 2020].
3. **Метрика L1**. Данная метрика вычисляет сумму абсолютных различий между координатами двух векторов. Данная мера сходства основана на сумме различий по каждому измерению [Mikolov, 2013].

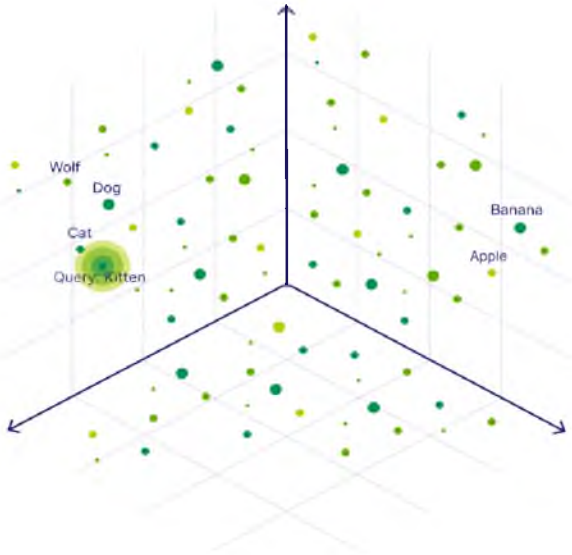

Рис. 5. Пример векторного пространства, используемого в рекомендательных системах
Fig. 5. An example of a vector space used in recommendation systems

Выбор метрики зависит от конкретных характеристик данных и желаемого поведения системы семантического поиска.

Метод RAG также может использовать и другие источники информации, например, графовые СУБД [Linyao et al, 2023; Linhao et al, 2023]. Примерами современных техник улучшения метода RAG являются генерация запросов к графовым СУБД с помощью БЯМ [Bowen et al, 2023], генерация дополнительных вопросов с помощью БЯМ для уточнения поиска, ранжирование найденных документов для оптимизации контекста и другие [Bowen et al, 2023; Linyao et al, 2023].

Использование метода RAG в интеллектуальной образовательной экосистеме

Рассмотрим пример реализации рекомендательной системы на основе контента на примере модуля рекомендаций в интеллектуальной образовательной экосистеме.

В данной системе представлены различные виды образовательного контента, например, курсы, лекции, статьи и видеозаписи, а также данные пользователей и требования работодателя. Соответствующие признаки вышеуказанных объектов могут включать различные метаданные, такие как тип контента, заголовок, описание, уровень сложности, продолжительность, формат (видео, текст), язык и любые другие атрибуты, преимущественно представленные в виде текстовых данных [Gao et al, 2023].

Использование embedding-векторов в данной предметной области помогает фиксировать семантические связи между словами и фразами, что может быть полезно для понимания контекста образовательного контента с одной стороны, или требований сферы занятости населения с другой стороны [Keraghel et al, 2024].

Атрибуты данных объектов объединяются и конвертируются в embedding-вектора, которые сохраняются в векторной СУБД Qdrant [Qdrant, 2024]. Векторное пространство для вакансий представлено на рис. 6.

Аналогичные преобразования происходят и с пользовательским интересами, целями обучения, последними освоенными компетенциями. Система сохраняет предпочтения пользователей – например, те образовательные материалы, которые пользователь посмотрел или которыми поделился в сети.

Далее происходит процесс сравнения embedding-векторов пользователей как с embedding-векторами образовательных ресурсов, формируя дальнейший список образовательных ресурсов для изучения, так и с embedding-векторами вакансий. Полученные данные образовательных ресурсов и вакансий в виде текстовых описаний поступают совместно с исходным запросом пользователя на вход LLM-модели. Выбранная

БЯМ генерирует описание рынка труда в соответствии с выбранным запросом, а также предлагает гиперссылки на наиболее подходящие вакансии и образовательные ресурсы.

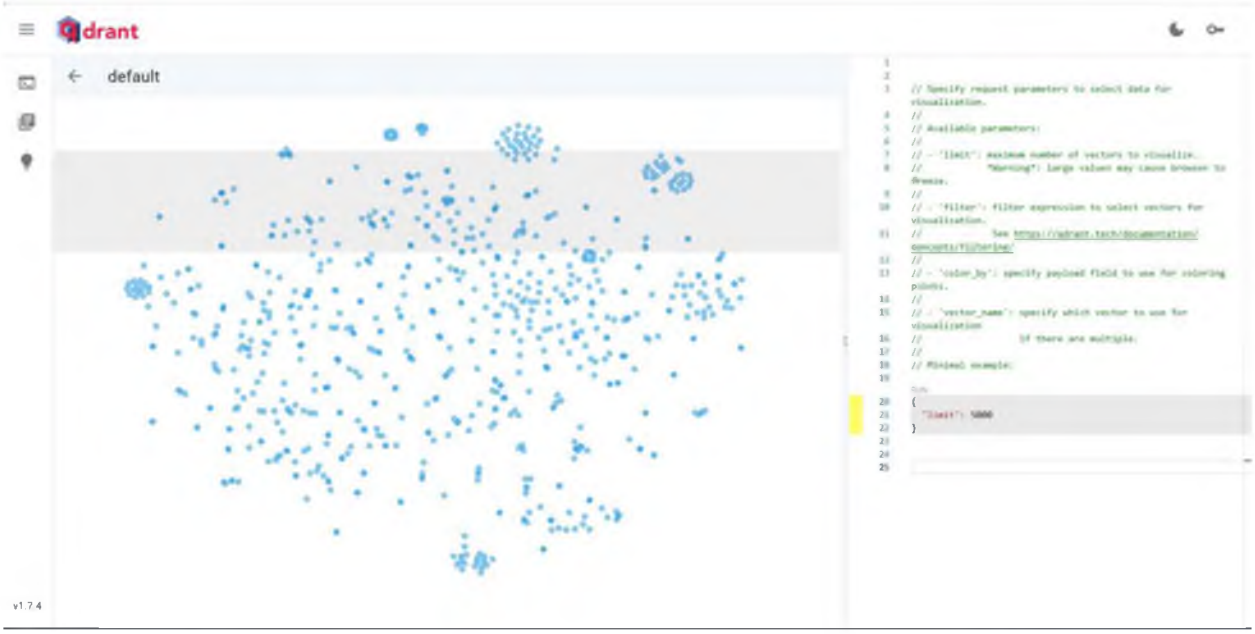

Рис. 6. Векторное пространство описаний вакансий в ИОЭ, визуализированное с помощью векторной СУБД Qdrant
Fig. 6. The vector space of job descriptions in the IOE, visualized using the Qdrant vector DBMS

Система также использует концепцию knowledge graph, представленную в виде графа компетенций [Оболенский, Шевченко, 2023], для улучшения работы большой языковой модели. При использовании поиска подходящих вакансий и описания рынка труда в соответствии с запросом пользователя, информация о его компетенциях и требуемых компетенциях вакансий в обезличенном виде также передается на вход нейросети.

Пример работы LLM-модели в ИОЭ представлен на рис. 7.

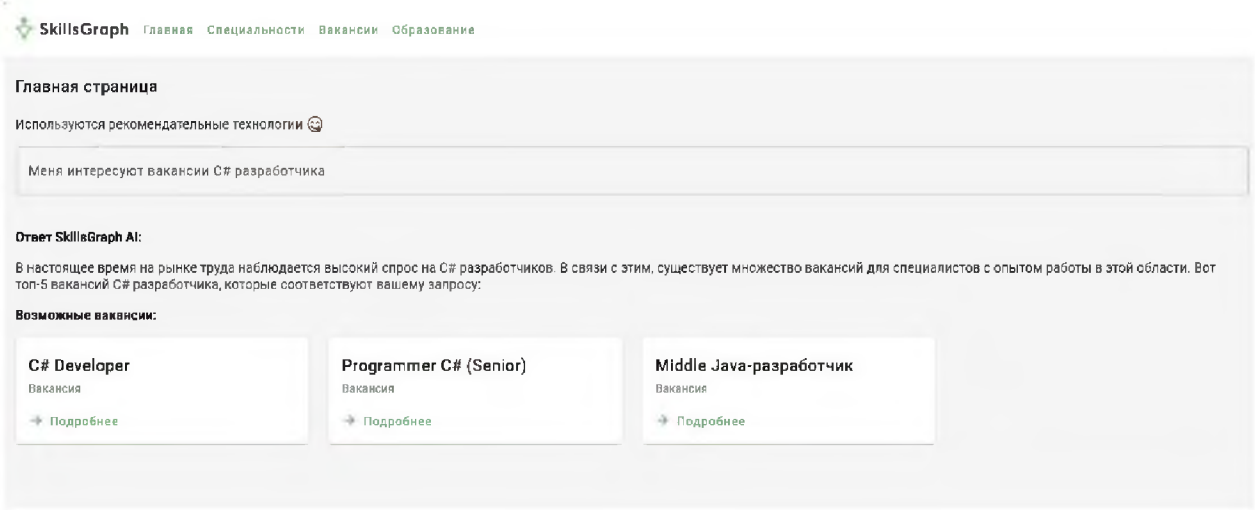

Рис. 7. Примеры работы LLM с использованием метода RAG в интеллектуальной образовательной экосистеме
Fig. 7. Examples of LLM work using the RAG method in an intelligent educational ecosystem

Для взаимодействия с большими языковыми моделями был реализован микросервис на языке Python, использующий библиотеку Langchain [Langchain, 2024]. В качестве embedding-модели использовалась модель GigaChat Embeddings, а в качестве LLM-модели – GigaChat Lite от компании Сбер [GigaChat, 2024].

Заключение

Таким образом, большие языковые модели могут быть использованы в рекомендательной системе на основе контента, лежащей в основе интеллектуальной образовательной экосистемы. Использование больших языковых моделей нейронных сетей предоставляет мощное и точное представление образовательных ресурсов, вакансий и интересов пользователя с учетом контекста. В качестве исходных данных для векторных представлений могут быть использованы заголовки, описания образовательных ресурсов, вакансий, а также другие текстовые атрибуты.

Создание и поддержка embedding-векторов объектов для большого объема образовательных ресурсов и вакансий может быть дорогостоящим с точки зрения вычислений и ресурсоемким процессом. Однако предложенный подход позволяет улучшить работу механизма извлечения признаков за счет использования больших языковых моделей и embedding-векторов.

Метод RAG позволяет дополнить БЯМ новыми знаниями, на которых она не была обучена. Использование векторных баз данных позволяет реализовать эффективный поиск embeddings-векторов для соответствующего запроса пользователя.

В рамках дальнейших исследований планируется дальнейшее развитие рекомендательной системы на основе БЯМ и метода RAG, а также сравнение различных БЯМ и embeddings-моделей в рамках данной предметной области.

Список литературы

Бабкин А.В., Корягин С.И., Либерман И.В., Клачек П.М. 2022. Индустрия 5.0: Создание интеллектуальной образовательной экосистемы. Экономика и индустрия 5.0 в условиях новой реальности (ИНПРОМ-2022), 76–79.

Малышев И.О., Смирнов А.А. 2024. Обзор современных генеративных нейросетей: отечественная и зарубежная практика. Международный журнал гуманитарных и естественных наук. №1-2(88).

Оболенский Д.М., Шевченко В.И. 2019. Интеллектуальные образовательные экосистемы. Сб. науч. тр. междунар. науч.-техн. конф. «DICTUM – FACTUM: от исследований к стратегическим решениям». Севастополь. 162–171. DOI: 10.32743/dictum-factum.2020.162-1714e4

Оболенский Д.М., Шевченко В.И. 2020. Концептуальная модель интеллектуальной образовательной экосистемы. Экономика. Информатика. 47(2): 390–401. DOI: 10.18413/2687-0932-2020-47-2-390-401.4e4e

Оболенский Д.М., Шевченко В.И. 2021. Обзор современных методов построения рекомендательных систем – на основе контента и гибридные системы. Мир компьютерных технологий: сборник статей всероссийской научно-технической конференции студентов, аспирантов и молодых ученых, Севастополь, 05–09 апреля 2021 г. Министерство науки и высшего образования РФ, Севастопольский государственный университет. Севастополь: Федеральное государственное автономное образовательное учреждение высшего образования «Севастопольский государственный университет», 151–156.

Оболенский Д.М., Шевченко В.И. 2023. Построение и анализ графа компетенций на основе данных вакансий с порталов поиска работы. Экономика. Информатика, 50(1): 191–202. https://doi.org/10.52575/2687-0932-2023-50-1-191-202

Achiam J., Adler S., Agarwal S., Ahmad L., Akkaya I., Aleman F., Almeida D., Altenschmidt J., Altman S., Anadkat S., et al. 2023. Gpt-4 technical report. arXiv preprint arXiv:2303.08774.

Bowen J., Gang L., Chi H., Meng J., Heng J., Jiawei H. 2023. Large Language Models on Graphs: A Comprehensive Survey. arXiv preprint arXiv:2312.02783.

Brown T. et al. 2020. Language models are few-shot learners. Advances in neural information processing systems 33. 1877–1901.

Devlin J., Chang M., Lee K., Toutanova K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. North American Chapter of the Association for Computational Linguistics.

Gao Y., Xiong Y., Gao X., Jia K., Pan J., Bi Y., Dai Y., Sun J., Guo Q., Wang M., Wang H. 2023. Retrieval-Augmented Generation for Large Language Models: A Survey. ArXiv, abs/2312.10997.

GigaChat. 2024. URL: https://developers.sber.ru/portal/products/gigachat

Graph Data Platform | Graph Database Management System. Neo4j. 2024. URL: https://neo4j.com/

High-Performance Vector Search at Scale. Qdrant – Vector Database – Qdrant. 2024. URL: https://qdrant.tech/

Keraghel I., Morbieu S., Nadif M. Beyond Words: A Comparative Analysis of LLM Embeddings for Effective Clustering. In: Miliou, I., Piatkowski, N., Papapetrou, P. (eds). 2024. Advances in Intelligent Data Analysis XXII. IDA 2024. Lecture Notes in Computer Science, vol 14641. Springer, Cham. https://doi.org/10.1007/978-3-031-58547-0_17

Langchain. 2024. URL: https://python.langchain.com/v0.2/docs/introduction/

Lewis P., Perez E., Piktus A., Petroni F., Karpukhin V., Goyal N., Kuttler H., Lewis M., Wen-tau Yih, Rocktaschel T., et al. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. Advances in Neural Information Processing Systems, 33:9459–9474, URL https://doi.org/10.48550/arXiv.2005.11401.

Luo L., Li Y.-F., Haffari Gh., Pan Sh. 2023. Reasoning on graphs: Faithful and interpretable large language model reasoning. arXiv preprint arXiv:2310.01061.

Mikolov T., et al. 2013. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Radford A., et al. 2019.Improving language understanding by generative pre-training.

Shadab I., Subhajit G. 2020. Efficient Ranking Framework for Information Retrieval Using Similarity Measure. DOI: 10.1007/978-3-030-37218-7_141.

Shoeybi M. et al. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. arXiv preprint arXiv:1909.08053.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser L., Polosukhin I. 2017. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

Weawiate – The AI-native database for a new generation of software. 2024. URL: https://weaviate.io/

YandexGPT 3 – Новое поколение генеративных текстовых нейросетей. 2024. YandexGPT. URL: https://ya.ru/ai/gpt-3

Yang L., Chen H., Li Zh., Ding X., Wu X. 2023. ChatGPT is not Enough: Enhancing Large Language Models with Knowledge Graphs for Fact-aware Language Modeling. arXiv preprint arXiv:2306.11489.

Zhang J., Lertvittayakumjorn P., Guo Y. 2019. Integrating Semantic Knowledge to Tackle Zero-shot Text Classification. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long and Short Papers), Minneapolis, Minnesota. Association for Computational Linguistics. 1031–1040.

Zhou D., Scharli N., Hou L., Wei J., Scales N., Wang X., Schuurmans D., Bousquet O., Le Q., Chi E.H. 2022. Least-to-Most Prompting Enables Complex Reasoning in Large Language Models. ArXiv, abs/2205.10625.

References

Babkin A.V., Koryagin S.I., Liberman I.V., Klachek P.M. 2022. Industriya 5.0: So-zdanie intellektual'noy obrazovatel'noy ekosistemy [Industry 5.0: Creating an intelligent educational ecosystem]. Ekonomika i industriya 5.0 v usloviyakh novoy real'nosti (INPROM-2022) [Economics and Industry 5.0 in the context of a new Reality (INPROM-2022)], 76–79.

Malyshev I.O., Smirnov A.A. 2024. Overview of modern generative neural networks: domestic and foreign practice. International Journal of Humanities and Natural Sciences. №1-2(88).

Obolenskiy D.M., Shevchenko V.I. 2019. Intellektual'nye obrazovatel'nye eko-sistemy [Intelligent educational ecosystems]. Sb. nauch. tr. mezhdunar. nauch.-tekhn. konf. «DICTUM – FACTUM: ot

issledova-niy k strategicheskim resheniyam» [Collection of scientific tr. international scientific and technical conf. "DICTUM – FACTUM: from research to strategic solutions"]. Sevastopol'. 162–171. DOI: 10.32743/dictum-factum.2020.162-1714e4

Obolenskiy D.M., Shevchenko V.I. 2020. A conceptual model of the intelligent educational ecosystem. Economics. Information Technologies, 47(2): 390–401. DOI: 10.18413/2687-0932-2020-47-2-390-401.4e4e

Obolenskiy D.M., Shevchenko V.I. 2021. Obzor sovremennykh metodov postroeniya rekomendatel'nykh sistem na osnove kontenta i gibridnye sistemy [An overview of modern methods of building recommendation systems – content-based and hybrid systems] Mir komp'yuternykh tekhnologiy: sbornik statey vserossiyskoy nauchno-tekhnicheskoy konferentsii studentov, aspirantov i molodykh uchenykh [The World of computer technology: a collection of articles of the All-Russian scientific and technical conference of students, postgraduates and young scientists], Sevastopol', 05-09 aprelya 2021. Ministerstvo nauki i vysshego obrazovaniya RF, Sevastopol'skiy gos-udarstvennyy universitet. Sevastopol': Federal'noe gosudarstvennoe avtonomnoe ob-razovatel'noe uchrezhdenie vysshego obrazovaniya "Sevastopol'skiy gosudarstvennyy universitet". 151–156.

Obolenskiy D.M., Shevchenko V.I. 2023. Building and Analyzing a Skills Graph Built Using Vacancy Data from Job Portals. Economics. Information Technologies. 50(1): 191–202. https://doi.org/10.52575/2687-0932-2023-50-1-191-202

Achiam J., Adler S., Agarwal S., Ahmad L., Akkaya I., Aleman F., Almeida D., Altenschmidt J., Altman S., Anadkat S., et al. 2023. Gpt-4 technical report. arXiv preprint arXiv:2303.08774.

Bowen J., Gang L., Chi H., Meng J., Heng J., Jiawei H. 2023. Large Language Models on Graphs: A Comprehensive Survey. arXiv preprint arXiv:2312.02783.

Brown T. et al. 2020. Language models are few-shot learners. Advances in neural information processing systems 33. 1877-1901.

Devlin J., Chang M., Lee K., Toutanova K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. North American Chapter of the Association for Computational Linguistics.

Gao Y., Xiong Y., Gao X., Jia K., Pan J., Bi Y., Dai Y., Sun J., Guo Q., Wang M., Wang H. 2023. Retrieval-Augmented Generation for Large Language Models: A Survey. ArXiv, abs/2312.10997.

GigaChat. 2024. URL: https://developers.sber.ru/portal/products/gigachat

Graph Data Platform | Graph Database Management System. Neo4j. 2024. URL: https://neo4j.com/

High-Performance Vector Search at Scale. Qdrant – Vector Database – Qdrant. 2024. URL: https://qdrant.tech/

Keraghel I., Morbieu S., Nadif M. Beyond Words: A Comparative Analysis of LLM Embeddings for Effective Clustering. In: Miliou, I., Piatkowski, N., Papapetrou, P. (eds). 2024. Advances in Intelligent Data Analysis XXII. IDA 2024. Lecture Notes in Computer Science, vol 14641. Springer, Cham. https://doi.org/10.1007/978-3-031-58547-0_17

Langchain. 2024. URL: https://python.langchain.com/v0.2/docs/introduction/

Lewis P., Perez E., Piktus A., Petroni F., Karpukhin V., Goyal N., Kuttler H., Lewis M., Wen-tau Yih, Rocktaschel T., et al. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. Advances in Neural Information Processing Systems, 33:9459–9474, URL https://doi.org/10.48550/arXiv.2005.11401.

Luo L., Li Y.-F., Haffari Gh., Pan Sh. 2023. Reasoning on graphs: Faithful and interpretable large language model reasoning. arXiv preprint arXiv:2310.01061.

Mikolov T., et al. 2013. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Radford A., et al. 2019.Improving language understanding by generative pre-training.

Shadab I., Subhajit G. 2020. Efficient Ranking Framework for Information Retrieval Using Similarity Measure. DOI: 10.1007/978-3-030-37218-7_141.

Shoeybi M. et al. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. arXiv preprint arXiv:1909.08053.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser L., Polosukhin I. 2017. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

Weawiate – The AI-native database for a new generation of software. 2024. URL: https://weaviate.io/

YandexGPT 3 – Новое поколение генеративных текстовых нейросетей. 2024. YandexGPT. URL: https://ya.ru/ai/gpt-3

Yang L., Chen H., Li Zh., Ding X., Wu X. 2023. ChatGPT is not Enough: Enhancing Large Language Models with Knowledge Graphs for Fact-aware Language Modeling. arXiv preprint arXiv:2306.11489.

Zhang J., Lertvittayakumjorn P., Guo Y. 2019. Integrating Semantic Knowledge to Tackle Zero-shot Text Classification. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long and Short Papers), Minneapolis, Minnesota. Association for Computational Linguistics. 1031–1040.

Zhou D., Scharli N., Hou L., Wei J., Scales N., Wang X., Schuurmans D., Bousquet O., Le Q., Chi E.H. 2022. Least-to-Most Prompting Enables Complex Reasoning in Large Language Models. ArXiv, abs/2205.10625.

Конфликт интересов: о потенциальном конфликте интересов не сообщалось.
Conflict of interest: no potential conflict of interest related to this article was reported.

Поступила в редакцию 20.06.2024
Поступила после рецензирования 31.08.2024
Принята к публикации 06.09.2024

Received June 20, 2024
Revised August 31, 2024
Accepted September 06, 2024

ИНФОРМАЦИЯ ОБ АВТОРАХ

Оболенский Денис Михайлович, инженер-программист, ООО «Новые технологии», г. Севастополь, Россия

Шевченко Виктория Игоревна, кандидат технических наук, доцент, заведующий базовой кафедрой «Корпоративные информационные системы», Севастопольский государственный университет, г. Севастополь, Россия

INFORMATION ABOUT THE AUTHORS

Denis M. Obolensky, senior software developer, New Technologies LLC, Sevastopol, Russia

Victoria I. Shevchenko, Candidate of Technical Sciences, Associate Professor, Head of the basic department "Corporate Information Systems", Sevastopol State University, Sevastopol, Russia