LLM and RAG-Based Question Answering Assistant for Enterprise Knowledge Management

Gürkan Şahin
R&D Centre
Adesso Türkiye
İstanbul, Türkiye
gurkan.sahin@adesso.com.tr

Karya Varol
R&D Centre
Adesso Türkiye
İstanbul, Türkiye
karya.varol@adesso.com.tr

Burcu Kuleli Pak
R&D Centre
Adesso Türkiye
İstanbul, Türkiye
burcu.kuleli@adesso.com.tr

2024 9th International Conference on Computer Science and Engineering (UBMK) | 979-8-3503-6588-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/UBMK63289.2024.10773564

***Abstract*— Large language models (LLM) have become integral to many natural language processing applications, particularly in the area of automatic question answering. In this study, a question answering system was developed to enable Adesso Türkiye employees to access internal company information quickly and accurately. A Retrieval Augmented Generation (RAG)-based question answering framework was constructed by utilizing multiple large language models and embedding techniques, along with content curated by experts in human resources and information security. The performance of the system was evaluated using ROUGE, BLEU, and accuracy metrics, and the results indicated high levels of success. Future work will focus on enhancing performance through the use of different language models, enriching the system with datasets from various domains, and integrating the developed system into MS Teams to ensure accessibility for employees.**

Keywords—generative artificial intelligence, natural language processing, large language models, question answering, retrieval augmented generation (RAG), GPT

I. INTRODUCTION

Question answering systems are designed to comprehend natural language queries posed by users and generate appropriate responses. These systems offer numerous benefits, including rapid information access, enhanced user experience, and increased efficiency in information retrieval across various domains. Consequently, they are now effectively employed in diverse fields such as education, healthcare, customer service, and banking. Typically, question answering systems leverage a combination of natural language processing and machine learning techniques. One prominent approach is the information retrieval-based method, which involves several stages. Initially, the question is analyzed both linguistically and semantically. Subsequently, potential answers are extracted from relevant documents using weighting techniques such as term frequency-inverse document frequency (TF-IDF). In the final stage, the most probable answer from the selected documents is presented. Another widely-used method in question answering systems is the language model-based approach. This approach harnesses the capabilities of large language models, including BERT (Bidirectional Encoder Representations from Transformers) [1] and GPT (Generative Pre-trained Transformer) [2]. These models, pre-trained on extensive datasets, can be fine-tuned for specific tasks such as question answering, yielding highly accurate results. Moreover, hybrid systems that integrate both information retrieval methods and large language models can be utilized in the development of question answering systems. In such systems, pertinent documents are retrieved through keyword-based searches, and large language models are employed to identify and extract the sentences or paragraphs containing the answers from these documents. This combined approach leverages the strengths of both methodologies to produce more robust and accurate question answering systems.

In this study, an artificial intelligence-supported question answering system was developed to enhance employee productivity by enabling Adesso Türkiye employees to access internal company information swiftly and accurately. Large language models from the GPT family were employed to enable the system to comprehend employee inquiries and retrieve the most pertinent answers with high accuracy from a database of internal company documents. The dataset utilized in this research was comprised of content prepared by the human resources and information security departments. A RAG [3] framework was implemented, leveraging large language models for the task of question answering. The effectiveness of the developed system was evaluated using pre-formulated question sets, and comprehensive analyses were conducted. The experimental results demonstrated that the proposed question answering system consistently delivered highly accurate, fast, and reliable responses to employees' inquiries. Additionally, feedback from employees indicated that the system played a crucial role in facilitating quick access to information and acted as a significant catalyst for enhancing their productivity.

II. FUNDAMENTALS

This section outlines the key principles involved in creating a question-answering application that uses a large language model and RAG. It also explains the core ideas and concepts that form the basis of these technologies.

A. Embedding Models

Embedding models in natural language processing are utilized to convert language elements such as words, sentences, and paragraphs into numerical vectors. This transformation enables the detection of semantic similarities and relationships between texts. Word2vec [4], a model developed by Google, uses two distinct algorithms, CBOW and Skip-Gram, for word representations. Another well-known word representation model is GloVe [5], created by Stanford University. For representing larger text units beyond words, Doc2vec [6] was inspired by word2vec. Additionally, FastText [7], developed by Facebook, can be employed for both word and text representations and distinguishes itself from word2vec by representing words through character n-grams. BERT, a language model from Google, is based on the transformer architecture. More recently, larger and more effective embedding models have emerged. In this study, the text-embedding-ada-002, text-embedding-3-small, and text-embedding-3-large models were utilized to extract text

representations from the company's documents. Furthermore, many open-source embedding models are available on the Hugging Face platform, in addition to those developed by OpenAI.

B. Large Language Models

Large language models are trained on extensive datasets, enabling them to grasp the intricacies of language and execute advanced natural language tasks such as summarization, question answering, text classification, and text generation. Developing large language models necessitates substantial data and significant computational power. The training process for these models involves multiple stages. Initially, there is data collection and pre-processing. The subsequent stage involves defining the model architecture. Typically, these models utilize the transformer [8] architecture, which leverages an attention mechanism to capture relationships between words. Key aspects of the model architecture include the model's size, the number of layers, and the number of parameters per layer. For instance, the GPT-3 model is a large language model containing 175 billion parameters. After training a large language model, it can be fine-tuned for specific subtasks by retraining the model on a smaller, task-specific dataset and optimizing the task-related hyperparameters. Training large language models also presents certain challenges. The requirement for vast amounts of data and specialized hardware with high computational capabilities, such as GPUs and TPUs, is significant. Furthermore, these models sometimes produce incorrect information, a phenomenon known as hallucination. Methods such as RAG can be employed to mitigate the effects of hallucination.

The following provides information about several prominent large language models in use today. BERT, developed by Google in 2018, is a transformer-based language model containing 342 million parameters. Gemini [9], which also supports Google's chatbot of the same name, can process multimodal data including images, text, audio, and video. GPT-3, released by OpenAI in 2020, is a language model with 175 billion parameters and was trained using resources such as Common Crawl, WebText2, and Wikipedia. GPT-3.5, an enhanced version of GPT-3, has been fine-tuned with human feedback, and its training data extends up to September 2021. GPT-4, released by OpenAI in 2023, is the largest model in the GPT series and is capable of multimodal data processing. GPT-4 Omni (GPT-4o) succeeds GPT-4 and offers several improvements, including faster response times and more natural human interaction compared to GPT-4-Turbo. LaMDA (Language Model for Dialogue Applications) [10] was developed by Google Brain in 2021 as a large language model focused on dialogue applications. Another significant model is LLaMA [11], an open-source model with approximately 65 billion parameters, released by Meta in 2023. Microsoft developed Orca [12], a model with 13 billion parameters that can run on personal computers. Despite having significantly fewer parameters, Orca delivers performance comparable to GPT-3.5 and GPT-4 and is built upon the 13 billion parameter LLaMA model. In this study, OpenAI's large language and embedding models were utilized to develop the question answering system.

C. Vector Databases

Once the numerical representation vectors of the texts are obtained, they must be stored in vector databases. Several options exist in this domain. Pinecone is notable for its high-speed similarity search capability, enabling efficient handling of large datasets. Weaviate, an open-source database, benefits from continuous community support and can manage vectors as well as structured and graphic data. Faiss, developed by Facebook AI, excels in large-scale vector searches and is optimized with GPU support to enhance search speed. Milvus is another open-source database designed for high-performance searches and is capable of managing large-scale and distributed datasets. Elasticsearch offers a hybrid approach, adept at both text-based and vector searches, ensuring swift and efficient operations on extensive datasets. In this study, the open-source vector database Chroma, available in the LangChain library, was utilized. The representation vectors derived from the text content were indexed in Chroma. The number of potential answers returned from a vector database query is determined by the k nearest neighbors. For instance, a k value of five indicates that the five closest answers to the query will be retrieved. However, setting a very high k value may lead to slower search results.

D. LangChain

LangChain is an open-source platform designed to facilitate the development of applications based on large language models. By abstracting the complexities associated with these models, LangChain allows for their use without the need for training or fine-tuning. Additionally, it supports the straightforward creation of RAG-based systems. Optimized for performance, LangChain is particularly favored in large-scale platforms and high-demand environments due to its efficiency in handling intensive tasks.

E. Knowledge Sources

When developing a question answering system with the RAG structure, diverse data sources are required. LangChain supports the utilization of information sources in multiple formats, including doc, docx, pdf, txt, and xlsx. Additionally, external sources such as web pages and databases can be integrated for question answering purposes. In this study, pdf documents from Adesso Türkiye's human resources and information security departments were used as data sources. Future research aims to expand the system by incorporating content from a broader range of domains.

III. RELATED WORK

In the study by Chidipothu et al. [13], RAG was used to improve the performance of large language models by refining responses from academic sources. Another study [14] utilized RAG to reduce hallucinations in the AtamQAS system, which was trained on primary historical data. Insights and recommendations for future research were provided. In [15], a CBR (Case-Based Reasoning)-RAG model was developed to enrich queries and improve the quality of responses. Zhang et al. [16] introduced a method called Retrieval Augmented Fine-tuning (RAFT) that ignores irrelevant documents, improving model performance on PubMed, HotpotQA, and Gorilla datasets. The model code was released as open source. Another study [17] presented an

adaptive question-answering framework that selects the best answering method based on query complexity, using a classifier to detect complexity levels. This approach significantly enhanced the system's efficiency and effectiveness.

In the context of Turkish large language models, several studies have been conducted. The research in [18] introduced VBART, a large language model pre-trained specifically for Turkish, using BART and mBART models. VBART showed strong performance in tasks like text summarization, heading creation, text modification, question answering, and question creation, and these models and datasets were made available on Hugging Face. In [19], cosmosGPT models were developed exclusively with Turkish corpora, presenting new fine-tune datasets and evaluation datasets. These models performed well across various tasks despite their smaller size. Kesgin et al. [20] introduced compact Turkish BERT models, trained on a diverse corpus of over 75GB. These models showed strong performance in tasks like zero-shot classification and were computationally efficient. In [21], Akyön et al. fine-tuned a multilingual T5 (mT5) transformer for question answering, question generation, and answer extraction using Turkish QA datasets. This study was the first to automate text-to-text question generation in Turkish, achieving state-of-the-art results across different datasets, including TQuADv1, TQuADv2, and XQuAD Turkish split.

Upon examining these studies, it became evident that no comprehensive question-answering system existed for both Turkish and company-specific applications. Consequently, the objective of this study is to develop a large language model-supported question answering system that allows company employees to access internal information quickly and with high accuracy. This system aims to significantly enhance the productivity and efficiency of employees by providing reliable and prompt responses to their queries.

IV. Methodology

Within the scope of this study, a RAG-based question answering system was developed utilizing large language models from the GPT family. RAG is a method employed to generate knowledge-based texts in natural language processing. This approach involves two main stages: retrieving information and generating text. In the first stage, information and documents pertinent to user queries are extracted from datasets such as knowledge bases, web pages, and textual sources within the information retrieval framework. In the subsequent stage, the text generation component uses this retrieved information to generate a coherent and meaningful answer. This two-stage approach allows language models to deliver highly accurate and detailed responses by drawing from an extensive pool of information. The RAG-based method is particularly effective and widely applied in knowledge-intensive tasks like question answering. Figure 1 illustrates the architecture and functioning of the RAG framework.

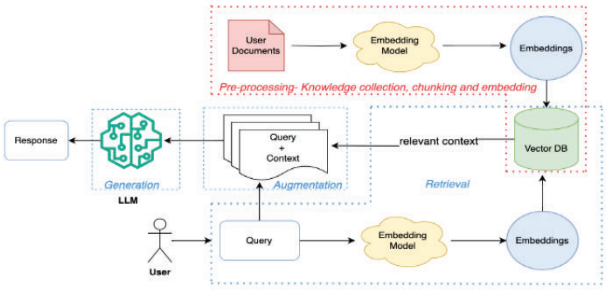

Fig. 1. RAG architecture

As illustrated in Figure 1, the RAG architecture encompasses several distinct steps. These steps include employing information sources, utilizing large language models, processing input, generating answers with the large language model, and conducting the search and retrieval of relevant answer segments. In the question answering system developed for this study, internal documents serve as the information sources. Embedding methods were applied to obtain digital representations of the texts, while text-chunking methods were used to break down large texts into manageable fragments. Vector databases were utilized to store the text representations and retrieve pertinent answer segments. The information retrieved from the vector database was synthesized into a coherent response using large language models. All these operations were seamlessly integrated and executed using the LangChain framework.

In this study, text documents produced by Adesso Türkiye's human resources and information security departments were employed as the dataset. The dataset consists of approximately one hundred separate documents. Detailed information about the dataset is provided in Table I.

TABLE I. Dataset Information

Domain	Document type	Number of documents	Number of tokens	Number of words
Human Resources	pdf	55	120 K	40 K
Information Security	pdf	41	60 K	20 K

Numerical vector representations of the texts were obtained by inputting document content into embedding models, including text-embedding-ada-002, text-embedding-3-large, and text-embedding-3-small, through the Azure OpenAI Service. If the text exceeded the embedding model's token capacity, it was divided into smaller subtexts using the RecursiveCharacterTextSplitter method in LangChain, with a chunk_size of 2000 and chunk_overlap of 100. These digital vectors were indexed in Chroma, an open-source vector database. The RetrievalQA framework in LangChain was then used to preprocess these vectors for question answering, utilizing the Chroma database to retrieve answers. Various GPT models synthesized these answer parts into a coherent response. The number of possible answers retrieved from the vector database was set to four, using cosine similarity as the metric. Responses included the answer text, the document name, and the page number, enabling users to easily find the original source.

During the development of the question answering system, Python was utilized as the programming language and the FastAPI library was employed to create the web application. For the user interface design, HTML, CSS, and Bootstrap

were used. LangChain libraries were selected to facilitate communication between the language models and the RAG architecture. Resources provided by OpenAI were accessed through the Azure OpenAI Service via Microsoft Azure. Both the web application with the user interface and the REST API endpoints were implemented and thoroughly tested. To evaluate the performance of the question answering system, question-answer datasets for both human resources and information security domains were created. The accuracy, BLEU, and ROUGE metrics were used to measure the correctness, quality, and reliability of the system's responses.

The accuracy, BLEU, and ROUGE metrics are essential tools for evaluating the performance of a RAG question answering system. Accuracy measures the proportion of correct answers, ensuring the system's reliability. BLEU evaluates the degree of overlap between the generated answers and reference texts, assessing the fluency and relevance of the responses. ROUGE metrics, including ROUGE-1, ROUGE-2, ROUGE-3, and ROUGE-L, focus on recall and provide a detailed analysis of the generated text. ROUGE-1 measures the overlap of unigrams, ROUGE-2 of bigrams, ROUGE-3 of trigrams, and ROUGE-L evaluates the longest common subsequence between the generated and reference texts. These metrics assess the completeness and informativeness of the answers. Together, accuracy, BLEU, and ROUGE metrics provide a comprehensive evaluation of the RAG system's correctness, quality, and ability to generate responses that closely resemble human-provided answers, ensuring a robust and effective question answering system.

V. Experiments, Results, And Discussion

The performance of the developed question-answering system was measured using datasets prepared by expert personnel. For the human resources domain, 80 question-answer pairs were created, while 90 question-answer pairs were developed for the information security domain. The responses provided by the system were reviewed by three different experts and assessed by majority vote. In addition to evaluating the answer texts, accuracy measurements also considered the document and page numbers where the correct answers were located. The accuracy results obtained from these question sets are presented in Table II. These results were derived using the GPT-4o LLM and the text-embedding-3-large embedding model. Although different combinations of LLMs and embedding models were utilized, the human evaluation of the results was time-consuming. Therefore, the accuracy values provided are specifically for the GPT-4o and text-embedding-3-large combination. The results of other model combinations were assessed using BLEU and ROUGE scores.

TABLE II. Answer Accuracies (GPT-4o & Text-Embedding-3-Large)

Domain	Answer accuracy	Document accuracy	Page number accuracy
Human Resources	0.8	0.67	0.62
Information Security	0.92	0.88	0.84

When Table II is examined, higher accuracy values are evident for the information security field compared to the human resources domain. A detailed look at the human resources documents shows that the same information can appear in different documents. This redundancy makes it harder to find the correct answer, as the same information is found in multiple sources, leading to potentially incorrect or incomplete responses. In contrast, the information security documents are more distinct, with each document covering a unique topic. This separation likely results in higher accuracy and precision for the information security responses. Furthermore, as shown in Table II, the accuracy for identifying the correct document and page number is lower than the accuracy of the answer text. This means that while the answer text itself may be correct, there can be mistakes in pinpointing the exact document and page number from which the answer was taken.

TABLE III. BLEU And Rouge Metrics For Human Resource Dataset Across Different Model Combinations (P: Precision, R: Recall, F1: F1 Score)

LLM & Embedding	BLEU	ROUGE-1			ROUGE-2			ROUGE-3			ROUGE-L		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
gpt-4o & text-embedding-3-large	**0.67**	0.8	0.79	**0.79**	0.73	0.74	**0.74**	0.69	0.67	**0.68**	0.76	0.75	**0.75**
gpt-4o & text-embedding-3-small	0.57	0.71	0.7	0.71	0.6	0.6	0.6	0.55	0.55	0.55	0.67	0.66	0.67
gpt-4o & text-embedding-ada-002	0.53	0.7	0.68	0.69	0.58	0.57	0.57	0.53	0.52	0.52	0.65	0.63	0.64
gpt-4-turbo & text-embedding-3-large	0.55	0.84	0.69	0.76	0.73	0.61	0.67	0.66	0.56	0.61	0.8	0.66	0.72
gpt-4-turbo & text-embedding-3-small	0.45	0.73	0.59	0.65	0.59	0.48	0.53	0.52	0.43	0.47	0.69	0.56	0.62
gpt-4-turbo & text-embedding-ada-002	0.43	0.71	0.56	0.63	0.56	0.45	0.5	0.49	0.4	0.44	0.66	0.53	0.59
gpt-35-turbo & text-embedding-3-large	0.47	0.71	0.65	0.68	0.6	0.56	0.58	0.53	0.5	0.51	0.66	0.61	0.63
gpt-35-turbo & text-embedding-3-small	0.37	0.62	0.56	0.59	0.47	0.43	0.45	0.4	0.38	0.39	0.57	0.52	0.54
gpt-35-turbo & text-embedding-ada-002	0.39	0.66	0.55	0.6	0.51	0.44	0.47	0.44	0.38	0.41	0.61	0.52	0.56

Table III presents the performance of various LLM and embedding model combinations on the human resources dataset. The models were evaluated using BLEU and ROUGE metrics, which include precision (P), recall (R), and F1 scores for ROUGE-1, ROUGE-2, ROUGE-3, and ROUGE-L. The combination of gpt-4o & text-embedding-3-large achieved the highest performance across most metrics, notably achieving high scores in BLEU and all ROUGE variants. This indicates that this combination provides highly accurate and comprehensive answers. The gpt-4o & text-embedding-3-

small and gpt-4o & text-embedding-ada-002 combinations also performed well, showing high BLEU scores and strong ROUGE metrics, though slightly lower than the best combination. These pairs are effective but less precise compared to gpt-4o & text-embedding-3-large. gpt-4-turbo & text-embedding-3-large and gpt-4-turbo & text-embedding-3-small combinations demonstrated moderate performance, with lower BLEU and ROUGE scores compared to gpt-4o pairings. gpt-35-turbo combinations showed the lowest performance across the metrics, indicating a noticeable difference in the quality of responses.

In summary, the results highlight the importance of choosing the right combination of large language models and embedding models. The superior performance of gpt-4o & text-embedding-3-large underscores the value of combining robust models and embeddings to achieve high accuracy and effectiveness in question-answering systems.

TABLE IV. BLEU AND ROUGE METRICS FOR INFORMATION SECURITY DATASET ACROSS DIFFERENT MODEL COMBINATIONS (P: PRECISION, R: RECALL, F1: F1 SCORE)

LLM & Embedding	BLEU	ROUGE-1			ROUGE-2			ROUGE-3			ROUGE-L		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
gpt-4o & text-embedding-3-large	**0.58**	0.76	0.79	**0.77**	0.65	0.64	**0.65**	0.56	0.57	**0.56**	0.68	0.68	**0.68**
gpt-4o & text-embedding-3-small	0.55	0.74	0.75	0.74	0.61	0.61	0.61	0.53	0.54	0.53	0.64	0.65	0.65
gpt-4o & text-embedding-ada-002	0.53	0.74	0.73	0.73	0.59	0.58	0.58	0.52	0.51	0.52	0.64	0.62	0.63
gpt-4-turbo & text-embedding-3-large	0.34	0.76	0.56	0.65	0.54	0.4	0.46	0.43	0.32	0.37	0.61	0.45	0.52
gpt-4-turbo & text-embedding-3-small	0.31	0.74	0.53	0.62	0.51	0.37	0.43	0.41	0.29	0.34	0.58	0.42	0.49
gpt-4-turbo & text-embedding-ada-002	0.28	0.73	0.5	0.59	0.49	0.34	0.4	0.38	0.27	0.32	0.57	0.39	0.47
gpt-35-turbo & text-embedding-3-large	0.30	0.74	0.54	0.62	0.53	0.39	0.45	0.43	0.32	0.37	0.61	0.44	0.51
gpt-35-turbo & text-embedding-3-small	0.27	0.71	0.5	0.59	0.48	0.35	0.41	0.38	0.29	0.33	0.56	0.41	0.47
gpt-35-turbo & text-embedding-ada-002	0.28	0.71	0.51	0.59	0.49	0.36	0.42	0.39	0.29	0.33	0.56	0.41	0.47

Table IV shows the performance of different LLM and embedding model combinations on the information security dataset, evaluated using BLEU and ROUGE metrics. The metrics include precision (P), recall (R), and F1 scores for ROUGE-1, ROUGE-2, ROUGE-3, and ROUGE-L. The combination of gpt-4o & text-embedding-3-large achieved the highest performance overall, with strong scores in BLEU and all ROUGE metrics. This indicates that this combination provides the most accurate and comprehensive answers. The combinations gpt-4o & text-embedding-3-small and gpt-4o & text-embedding-ada-002 also performed well, showing high BLEU scores and strong ROUGE metrics, but not quite matching the top combination. These pairings are effective but slightly less precise. The gpt-4-turbo and gpt-35-turbo combinations showed moderate performance. They had lower BLEU and ROUGE scores compared to the gpt-4o combinations. This suggests that while these models are still effective, they do not perform as well as the top combination.

In conclusion, the results highlight the importance of choosing the right combination of large language models and embedding models. The best performance was achieved by gpt-4o & text-embedding-3-large, showing the value of robust models and embeddings for achieving high accuracy in question-answering systems.

TABLE V. SAMPLE QUESTIONS AND ANSWERS

Question	Domain	Answer
What social activity clubs are available at the company?	Human Resources	The social activity clubs at adesso Türkiye are as follows: adesso Running Team, adesso Volunteers, adesso Football Team, adesso Gaming Team, adesso Diving Team, adesso Extreme Team, adesso Tennis Team, and adesso Jukebox. 'page': 5, 'source': ' adesso+Türkiye+Internal+Communication+Guide.pdf'
How much annual paid leave do I have in my first year?	Human Resources	In your first year, your annual paid leave will be deducted from the 14-day period, provided you have worked for at least one year. This means that the annual paid leave you are entitled to in your first year will be equivalent to the 14-day period allocated for your second year. However, if you take leave that you have not yet earned, it should not exceed a total of 7 working days per year. 'page': 8, 'source': 'adesso+Türkiye+Principles+and+Rules+of+Ethical+Conduct.pdf'
What is ransomware?	Information Security	Ransomware is a malicious software that encrypts your computer or files, blocking access and later demanding money from you. It has become widespread because it is highly profitable for criminals. Ransomware typically infects your computer when you open a virus-laden attachment or click on a link in a malicious email. The name "ransomware" comes from the fact that criminals demand money from you. If you don't pay the ransom, you may face the threat of your files being published for everyone to see. 'page': 1, 'source': 'adesso+Türkiye+Ransomware.pdf'
What should be done after a GDPR violation is detected?	Information Security	After a GDPR violation is detected, companies are required to report this violation to the relevant regulatory authority within 72 hours. Additionally, a report should be prepared regarding the causes and effects of the violation, and necessary measures should be taken. These measures should be aimed at preventing similar violations from recurring and minimizing the impact of the violation. Furthermore, the rights of individuals affected by the violation should be respected, and compensation should be provided where necessary. 'page': 1, 'source': 'adesso+Türkiye+GDPR.pdf'

VI. CONCLUSION

In this study, a question answering system was developed for Adesso Türkiye employees to provide quick and accurate answers to daily questions, thereby making their work easier. Content documents prepared by the human resources and information security departments were utilized, and a large language model-supported question answering assistant using the RAG structure was created. The text-embedding-ada-002, text-embedding-3-large, and text-embedding-3-small embedding models were used to obtain text vectors. Additionally, the GPT-4o, GPT-4-Turbo, and GPT-35-Turbo large language models were utilized to convert the answers into text. Question and answer pairs for each domain were prepared by experts, and the developed system was tested on this dataset. Each LLM and embedding model combination's performance was evaluated using BLEU, ROUGE-1, ROUGE-2, ROUGE-3, ROUGE-L, and accuracy metrics. It was observed that the GPT-4o & text-embedding-3-large combination achieved the best performance for both datasets. As a result of these evaluations, it was found that the system could produce answers with high BLEU and ROUGE scores. The initial evaluations indicated that this system would enhance employee efficiency in accessing company information.

In future studies, different text embedding methods and the latest large language models will be used to further improve answer quality. Additionally, content from other company domains will be added to the question answering system as information sources. Future developments include UI enhancements, integrating an admin panel and login mechanism, and incorporating the question answering system into Microsoft Teams to make it available to all employees. This expanded implementation aims to further increase usability and accessibility throughout the company.

REFERENCES

[1] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.

[2] G. Yenduri, M. Ramalingam, G. C. Selvi, Y. Supriya, G. Srivastava, P. K. R. Maddikunta, et al., "Generative pre-trained transformer: A comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions," arXiv preprint arXiv:2305.10435, 2023.

[3] W. Fan, Y. Ding, L. Ning, S. Wang, H. Li, D. Yin, et al., "A survey on RAG Meeting LLMs: Towards retrieval-augmented large language models," in Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 6491-6501, Aug. 2024.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv preprint arXiv:1301.3781, 2013.

[5] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532-1543, Oct. 2014.

[6] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in Proc. International Conference on Machine Learning, pp. 1188-1196, Jun. 2014.

[7] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," Trans. Assoc. Comput. Linguistics, vol. 5, pp. 135-146, 2017.

[8] S. Jeong, J. Baek, S. Cho, S. J. Hwang, and J. C. Park, "Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity," arXiv preprint arXiv:2403.14403, 2024.

[9] G. Team, R. Anil, S. Borgeaud, Y. Wu, J. B. Alayrac, J. Yu, et al., "Gemini: a family of highly capable multimodal models," arXiv preprint arXiv:2312.11805, 2023.

[10] R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H. T. Cheng, et al., "LaMDA: Language models for dialog applications," arXiv preprint arXiv:2201.08239, 2022.

[11] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M. A. Lachaux, T. Lacroix, et al., "LLaMA: Open and efficient foundation language models," arXiv preprint arXiv:2302.13971, 2023.

[12] S. Mukherjee, A. Mitra, G. Jawahar, S. Agarwal, H. Palangi, and A. Awadallah, "Orca: Progressive learning from complex explanation traces of GPT-4," arXiv preprint arXiv:2306.02707, 2023.

[13] N. Chidipothu, R. Anderson, and M. N. Hoque, "Enhancing the performance of large language models (LLMs) in academic knowledge mining through retrieval augmented generation (RAG),".

[14] E. Karaarslan, A. Y. Alan, S. Kumbalı, and Ö. Aydın, "Towards a reliable and sustainable question answering system for better understanding of Atatürk's principles: AtamQAS," Available at SSRN 4789423, 2024.

[15] N. Wiratunga, R. Abeyratne, L. Jayawardena, K. Martin, S. Massie, I. Nkisi-Orji, et al., "CBR-RAG: Case-based reasoning for retrieval augmented generation in LLMs for legal question answering," in Proc. International Conference on Case-Based Reasoning, Cham, Switzerland: Springer Nature, pp. 445-460, Jun. 2024.

[16] T. Zhang, S. G. Patil, N. Jain, S. Shen, M. Zaharia, I. Stoica, and J. E. Gonzalez, "Raft: Adapting language model to domain specific RAG," arXiv preprint arXiv:2403.10131, 2024.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., "Attention is all you need," in Advances in Neural Information Processing Systems, vol. 30, 2017.

[18] M. Turker, M. E. Ari, and A. Han, "VBART: The Turkish LLM," arXiv preprint arXiv:2403.01308, 2024.

[19] H. T. Kesgin, M. K. Yuce, E. Dogan, M. E. Uzun, A. Uz, H. E. Seyrek, et al., "Introducing cosmosGPT: Monolingual training for Turkish language models," arXiv preprint arXiv:2404.17336, 2024.

[20] H. T. Kesgin, M. K. Yuce, and M. F. Amasyali, "Developing and evaluating tiny to medium-sized Turkish BERT models," arXiv preprint arXiv:2307.14134, 2023.

[21] F. Ç. Akyön, A. D. E. ÇAVUŞOĞLU, C. Cengiz, S. O. Altinuç, and A. Temizel, "Automated question generation and question answering from Turkish texts," Turkish Journal of Electrical Engineering and Computer Sciences, vol. 30, no. 5, pp. 1931-1940, 2022.