International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Context-Aware Multi-Source Intelligent Knowledge Retrieval System

Prof. Anamika Shukla[1], Mansi Jadhav[2], Harsh Hule[3], Glenn Joseph[4]

[1]Assistant Professor, Department of Computer Science, Ajeenkya's DY Patil University, Lohegaon, Airport Rd, Charoli Budruk, Pune, Maharashtra
[2]B.Tech Student, Department of Computer Science, Ajeenkya's DY Patil University, Lohegaon, Airport Rd, Charoli Budruk, Pune, Maharashtra

ABSTRACT
Enterprise organisation tackles large volumes of knowledge, which is not centralised but distributed across multiple heterogeneous sources. Such heterogeneous sources are structured datasets, web pages and documents. Traditional keyword- based systems failed to retrieve contextually relevant information, which produced incomplete or inaccurate responses. Even large language models (LLMs) generated hallucinated outputs. The reason for hallucinated outputs was nothing but operating without reliable knowledge grounding. Eventually, to overcome these challenges, this paper proposes a context-aware multi-source intelligent knowledge retrieval system based on the retrieval augmented generation (RAG) framework. This system integrates multiple data sources, including PDFs, websites, and structured data repositories. Converting them into vector embeddings for efficient semantic retrieval. The system improves knowledge accessibility by providing accurate context- aware answers grounded in enterprise data. The system's mechanism of experimental evaluation demonstrates improved retrieval relevance and response quality compared to the traditional search mechanism. Henceforth, the proposed framework contributes to the development of scalable enterprise knowledge assistants capable of delivering reliable, explainable and contextually accurate information retrieval in a real-time environment.

KEYWORDS: Retrieval- Augmented Generation, Enterprise Knowledge Management, Large Language Models, Context- Aware Systems, Multi-Source Retrieval

INTRODUCTION

There is a critical question that concerns modern enterprises as they rely heavily on data-driven decision making; well, access to accurate and relevant information is essential. However, Enterprise data is often distributed across multiple sources, such as databases, documents, APIs, and internal knowledge bases. This fragmentation makes it difficult to retrieve meaningful insights efficiently. Large Language Models (LLMs) have demonstrated strong capabilities in generating human-like responses and understanding natural language queries. Despite their effectiveness, these models rely primarily on pre-trained knowledge and may produce outdated or incorrect information when dealing with domain-specific enterprise data.

To overcome these limitations, Retrieval-Augmented Generation (RAG) has emerged as a promising approach that combines information retrieval with text generation. By retrieving relevant data from external sources and incorporating it into the generation process, RAG systems improve accuracy and

contextual relevance. This system proposes a multi-source, context-aware RAG system tailored for enterprise environments. The system integrates the data from various sources and dynamically adapts to user queries, ensuring accurate and context-rich responses.

1.1 Background of the Study

In recent years, Natural Language Processing (NLP) has improved extensively. The work done here introduced a transform architecture used to understand relationships between words in a sequence [1]. Later discovered that large-scale models can perform multiple tasks on the basis of training [2].

1.2 Problem Description

Still, there were limitations that needed to be overcome as the large language models generated incorrect or misleading information, especially when the model was not trained with the same knowledge. Whereas,organisations deal with large amount of data gathered from various formats such as documents, databases, and web content. The existing systems either depended on keyword-based search or used static models, which had no access to updated data. Hence, this created a hurdle to retrieve context and enterprise (relevant) based responses. Even the lack of integration between structured and unstructured data added to the challenges faced by users of the existing system.

1.3 Importance of research

Focusing on how information can be retrieved and used in an enterprise system, combining the retrieval mechanism with language models. Useful for customer support systems, enterprise knowledge management, decision making support and, importantly, internal document search.

LITERATURE REVIEW

The architecture mentioned here has formed the backbone of modern large language models (LLMs) [1]. Built on the previous, demonstrated that large-scale language models can perform diverse tasks using minimal examples through few learning [2]. And further to over this limitations, proposed Retrieval-Augmented Generation (RAG), that integrates knowledge retrieval (external) with text generation [3].

Recent advancements in Retrieval-Augmented Generation (RAG) have led to a growing body of research focusing on improving system efficiency, scalability, and reliability. A comprehensive survey highlights the evolution of RAG architectures from basic retrieval-generation pipelines to more modular and advanced frameworks [7]. The study categorises RAG into naive, advanced, and modular types, emphasising the importance of integrating retrieval, augmentation, and generation components effectively. Further research provides a detailed overview of the current landscape of RAG systems, focusing on their evolution and real-world application [8].

Their approaches demonstrated that retrieval could scale efficiently without increasing model size [5]. Making it suitable for large-scale applications.

The authors do discuss limitations of large language models, such as hallucination and outdated knowledge, by incorporating dynamic external data sources. They identified key challenges which included scalability, bias and ethical concerns in deployment.

Another important contribution explores Retrieval-Augmented Language Models (RALMs), covering both generation and understanding paradigms [9]. This study provides a structured taxonomy of RAG systems and emphasises the importance of robust retrieval mechanisms and efficient integration with language models. It also highlights that retrieval quality directly impacts the overall performance of the system.

Compared to traditional keyword-based search, DPR significantly improves retrieval performance in the open-domain question answering task [4].
The recent research has focused on the evaluation and benchmarking of RAG systems. A study presented a review of evaluation techniques for RAG systems, addressing challenges related to measuring factual accuracy, relevance, and efficiency [10]. The paper emphasises that traditional evaluation metrics are insufficient for hybrid systems like RAG and proposes more advanced evaluation frameworks.
Recent experimental research demonstrates practical advanced diversifications and knowledge adaptation techniques [12].

RESEARCH GAP

The transform-based models and Retrieval-Augmented Generation (RAG) systems have improved NLP performance. Still, several limitations exist in current approaches. Early models demonstrated strong contextual understanding [1]. Later models showed impressive generalisation abilities [2]. RAG-based approaches solved this by integrating retrieval with generation [3]. Models still struggled with heterogeneous data formats, context switching through multiple documents and domain-specific terminology [4]. Making clear that we need a system for integrating multi source fata effectively, maintaining context awareness across retrieval documents and ultimately providing accurate and reliable responses in enterprise settings.

PROPOSED METHODOLOGY

The proposed system follows a retrieval augmented pipeline that integrates data preprocessing, embedding-based retrieval and language model generation.
Firstly, the model begins to collect data from multiple enterprise sources. The collected data is cleaned and transformed into a uniform textual format. After preprocessing, documents are divided into smaller chunks to improve processing efficiency.
Each chunk is then converted to a vector embedding using a pretrained model. These help us to capture semantic meaning and are stored in a vector database.
When the model receives a query, it converts it into vector embeddings and then compares it with stored vectors to gain relevant information. Finally, the retrieved context is passed on to a language model to generate the final response.
This approach is built upon earlier RAG frameworks [3], but extending them in a way to incorporate multi-source integration and improved context handling.

A. System architecture:

The design system effectively accesses the appropriate knowledge across a variety of diverse data and produces the correct response. The architecture is divided into many interrelated modules that include data ingestion, document preprocessing, embedding generation, Vector storage, retrieval mechanism and response Generation.

4.1 Data ingestion layer:

This module gathers data from multiple enterprise sources like PDF files, web pages and structured data. The sources possess good organisational information which requires to be digested and indexed to be retrieved easily. The system will accept a variety of inputs and will transform them into a homogeneous textile representation to enable additional processing.

4.2 Commentary and preparation of documents and chunks:

Following ingestion of the data, the documents are subjected to preprocessing to clean and format the text. The preprocessing process consists of text extraction, elimination of unneeded symbols, normalization and splitting of large documents into smaller parts. Another reason is that document chunking is a crucial step or procedure since large language models work better with smaller and contextually relevant text segments.

4.3 Embedding generation:

The process texture is then mapped to a vector embedding with a pre-trained embedding model. On these days, an embedding means the semantic meaning numerically. This allows similarity comparison of the query and document chunks efficiently. The system takes advantage of these embeddings in extracting contextual relationships between words and phrases, which enhances the capability of systems to generate relevant information.

4.4 Vector database storage:

Now, the resulting embeddings are placed in a database of vectors, which can be used to perform a similarity search.

4.5 Retrieval Mechanism

When a user enters a query, it also converts it into an embedding. The system then compares it with stored embeddings to find the closest matches.

Only the most relevant chunks are selected. This step ensures that unnecessary data is not passed to the model.

4.6 Response Generation

Finally, the selected information is combined with the user's query and is sent to the language model.

The model generates a response based on both the query and the retrieved data. This helps in reducing incorrect answers and improving the quality of output.

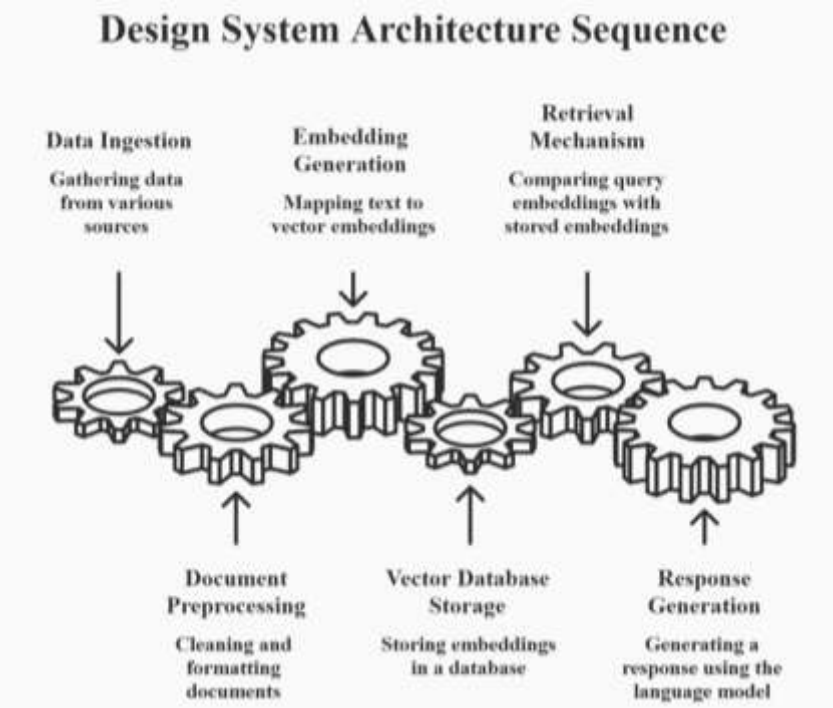


Figure 1: Design System Architecture.

DESIGN OF RESEARCH

The research is designed in an experimental approach and follows a system design. It's Applied research where the system was designed, implemented and tested. The results were generated on the input of real-world enterprise data. Focused on system development and evaluation. Improving retrieval accuracy and response quality was the main motive for this research.

TOOLS AND INSTRUMENTS

The model was designed based on several tools:
Programming language: Python
Framework: Fast API (for backend development)
Embedding models: Pretrained embedding models (OpenAI)
Vector Database: FAISS
Language Model: Large Language Model (LLM)
Libraries: NLP and data processing libraries

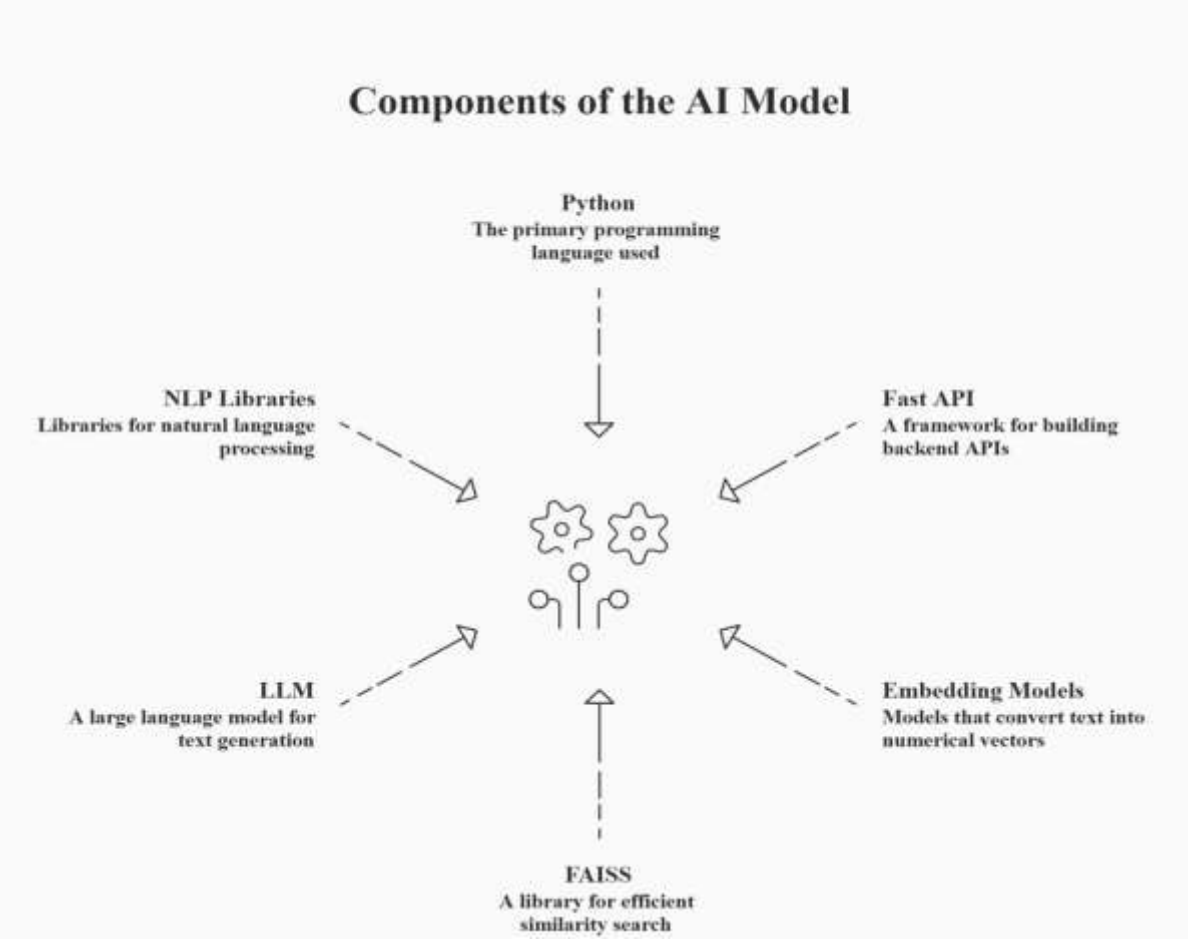


Figure 2: Components of the AI Model

MORAL ASPECTS

Ethical considerations for such an enterprise-relevant model, where data can be sensitive, are important and such AI-based systems.

The system is designed to avoid misuse of sensitive enterprise data, ensure data privacy and confidentiality and provide transparent and explainable responses.

ANTICIPATED RESULTS

The expected result of the system includes better retrieval accuracy, better understanding of queries, more context-aware responses and fewer chances of hallucination in generated outputs.

RESULTS AND FINDINGS

The designed output indicated a noticeable improvement in the way information was retrieved when compared to traditional keyword-based search methods. One primary observation is that the system can understand the semantic meaning of users' queries rather being depended on the exact keyword matches. This allows it to retrieve more relevant, more contextual and appropriate information, even when the query is phrased differently from the stored data. This resulted in eventual improved responses that align with findings in retrieval-based systems. Embedding techniques improved search quality by focusing on meaning rather than surface-level text.

Another key finding is the system's ability to effectively handl multi source data. As a result, users can receive more comprehensive answers that combine knowledge from various inputs. Thus, increasing the overall usefulness of the system also in complex query scenarios.

The use of Retrieved-Augmented Generation also contributes to reducing incorrect or misleading responses. By grounding the generated output in retrieved data, the system minimises the chances of hallucination, which is a common issue in language models [3].

While the result highlights certain dependencies. The performance of the system is closely tied to the quality of embeddings and the effectiveness of the retrieval process. Selecting the appropriate number of retrieved documents (top k) plays a crucial role in balancing accuracy and efficiency.

Overall, the model has improved at a good scale and generates accurate and contextually relevant responses. The system derives promising results for real-world enterprise applications where accurate and meaningful information access is essential.

CONCLUSION

In this research, a multi-source, context-aware enterprise RAG system has been proposed and analysed aim was to improve knowledge retrieval and response generation study highlights the limitations of traditional search systems (keyword-based), particularly in handling dynamic, large-scale and heterogeneous enterprise data.

By combining the proposed approach, the system addresses these limitations in a practical and effective manner.

The quality of outputs compared to systems that rely on either retrieval or generation was much more impressive. One of the key contributions of this work is the emphasis on context awareness across multiple data sources.

Instead of treating each source independently, the system brings together information from various formats and uses it collectively to answer users' queries. Thus, make it worthwhile for enterprise environments.

The results demonstrate that the system can improve accuracy, reduce hallucination and provide more meaningful responses. Also, maintaining flexibility and scalability. While the system still has certain

limitations that remain like dependency on embedding quality and computational cost, the overall approaches show strong potential.

In conclusion, this research contributes to the growing field of retrieval augmented systems by presenting a solution that is both practical and adaptable. The system builds a foundation for further advancements in enterprise knowledge management and intelligent information retrieval systems.

FUTURE SCOPE

1. Expansion of the System to Incorporate Real-Time Data Streams

The system can be further improved by expanding its scope to cover live APIs, dashboards, and continuously updated enterprise databases. As a result, it will respond in real time, minimising the possibility of responding with outdated and obsolete information.

2. Enhancement of Context Retention in Multi-Turn Conversations

Currently, the system is designed to process single questions.

Future research may be dedicated to improving the conversation context retention. In other words, future development will provide the opportunity to develop multi-turn conversation capabilities.

3. Fine-Tuning of the Model with Domain-Specific Data

Further improvement of the system's performance is likely to involve fine-tuning of the language model based on industry data, thereby increasing knowledge of the language model on specific domains.

4. Advancements in Retrieval Optimisation Methods

For future optimisation, there is room for researching advanced retrieval techniques, namely, hybrid methods that combine semantic search and keyword search. Dynamic top-k selections, re-ranking, and other strategies may also benefit the performance.

5. Scalability for Larger-Scale Deployment in an Enterprise Environment

As enterprise data continues to grow in amount, scalability may become a crucial challenge to address through further development. The development should focus on distributed vector databases, efficient indexing strategies, and performance optimisation.

6. Development of Explainable AI Features

In order to enhance explainability and increase users' confidence in the system, future development could involve building explainable AI features showing reasoning and/or sources behind the response.

7. Supporting Different Types of Input Data

The system can be further developed by supporting different types of data input (images, voice, video). It may significantly improve the processing of enterprise data by expanding retrieval and generation capabilities in multimodal contexts.

REFERENCES

1. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, “Attention Is All You Need”, Advances in Neural Information Processing Systems, 2017.
2. Tom B.B., Benjamin M., Nick R., Melanie S., Jared K., Greg B., Sam M., Alec R., Mark C., Christopher H., “Language Models are Few-Shot Learners”, Advances in Neural Information Processing Systems, 2020.
3. Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, Douwe Kiela,

"Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks", Advances in Neural Information Processing Systems, 2020.

4. Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, Wen-tau Yih, "Dense Passage Retrieval for Open-Domain Question Answering", Conference on Empirical Methods in Natural Language Processing, 2020.
5. Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Tomás Pfaff, Sebastian Farquhar, David K. Misra, et al., "Improving Language Models by Retrieving from Trillions of Tokens", International Conference on Machine Learning, 2022.
6. Yunfan Gao, Yun Xiong, Xinyu Gao, Kang Zhou, Jiawei Chen, Yuxuan Li, Jiarui Sun, Zhiyuan Liu, Maosong Sun, "Retrieval-Augmented Generation for Large Language Models: A Survey", arXiv preprint, 2023.
7. S. Gupta, A. Gupta, S. Jain, "A Comprehensive Survey of Retrieval-Augmented Generation: Evolution, Current Landscape and Future Directions", arXiv preprint, 2024.
8. Y. Hu, Y. Lu, "RAG and RAU: A Survey on Retrieval-Augmented Language Models", arXiv preprint, 2024.
9. A. Gan, X. Chen, Z. Liu, "Retrieval-Augmented Generation Evaluation in the Era of Large Language Models", arXiv preprint, 2025.
10. M. Cheng, L. Wang, Y. Zhao, X. Li, "A Survey on Knowledge-Oriented Retrieval-Augmented Generation", arXiv preprint, 2025.
11. J. Keller, M. Stein, T. Gollub, "CIR at TREC 2024 RAG: Augmented Generation with Diversified Segments", Text Retrieval Conference, 2024.