International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 03 | March - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Langchain-Chat with My PDF

M. Deepak[1], A. Anusha[2], P. Phanivighnesh[3], Dr. G. Sreenivasulu[4]

[1]Department of CSE, J.B. Institute of Engineering and Technology, Hyderabad
[2]Department of CSE, J.B. Institute of Engineering and Technology, Hyderabad
[3]Department of CSE, J.B. Institute of Engineering and Technology, Hyderabad
[4]Head of Department of CSE, J.B. Institute of Engineering and Technology, Hyderabad

---

Abstract – This paper presents a state-of-the-art system that is intended to facilitate natural language interaction with PDF documents. Leveraging the powerful Retrieval-Augmented Generation (RAG) algorithm, the solution seamlessly integrates information retrieval and generative language models to generate precise and context-sensitive responses.

The operation starts when a user uploads a PDF file. The system proceeds to process the file, breaking it down into bite-sized text chunks that are kept organized for easy retrieval. Upon the submission of a query by a user, the RAG algorithm locates the most applicable parts of the document and uses a generative language model to build an understandable and accurate response.

By combining the LangChain platform with the RAG approach, this paper introduces an effective tool for extracting precise information from long PDF documents efficiently. Its capacity to provide relevant and accurate responses makes it particularly valuable in education, research, and documentation professions where immediate access to accurate information is paramount.

Key Words: Retrieval-Augmented Generation (RAG), LangChain Framework, PDF Document Querying, Information Retrieval, Generative Language Models

1.INTRODUCTION

This paper presents LangChain-Chat with My PDF, a system designed to revolutionize document interaction by enabling natural language question-answering directly from PDF files. Users often face challenges navigating extensive digital documents in fields such as law, academia, healthcare, and corporate business. Traditional search methods, limited to keyword matching, frequently yield incomplete or irrelevant results. This system addresses these limitations by employing Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs) to provide precise, contextually relevant answers through conversational queries.

By leveraging the LangChain framework, the system integrates LLMs, vector databases, and natural language processing (NLP) utilities to enable efficient information retrieval. Key technologies include models like GPT and BERT for natural language understanding, and vector databases such as Pinecone or FAISS for fast similarity-based search. The RAG approach enhances accuracy by retrieving relevant text snippets before generating coherent responses.

This solution offers broad applications across multiple domains. Legal professionals can quickly access key clauses and case details, researchers can extract insights from academic papers, and healthcare practitioners can locate critical patient information or clinical guidelines. Additionally, corporate teams can efficiently retrieve policy details, financial records, or training material, improving overall productivity.

By combining advanced retrieval and generation techniques, this system significantly enhances information accessibility, making complex documents easier to navigate and reducing the time spent searching for crucial details.

In this chapter Section 2 reviews Working Mechanism. Section 3 Architecture. Section 4 Technologies. Section 5 Result. Section 6 Conclusions.

2. WORKING MECHANISM

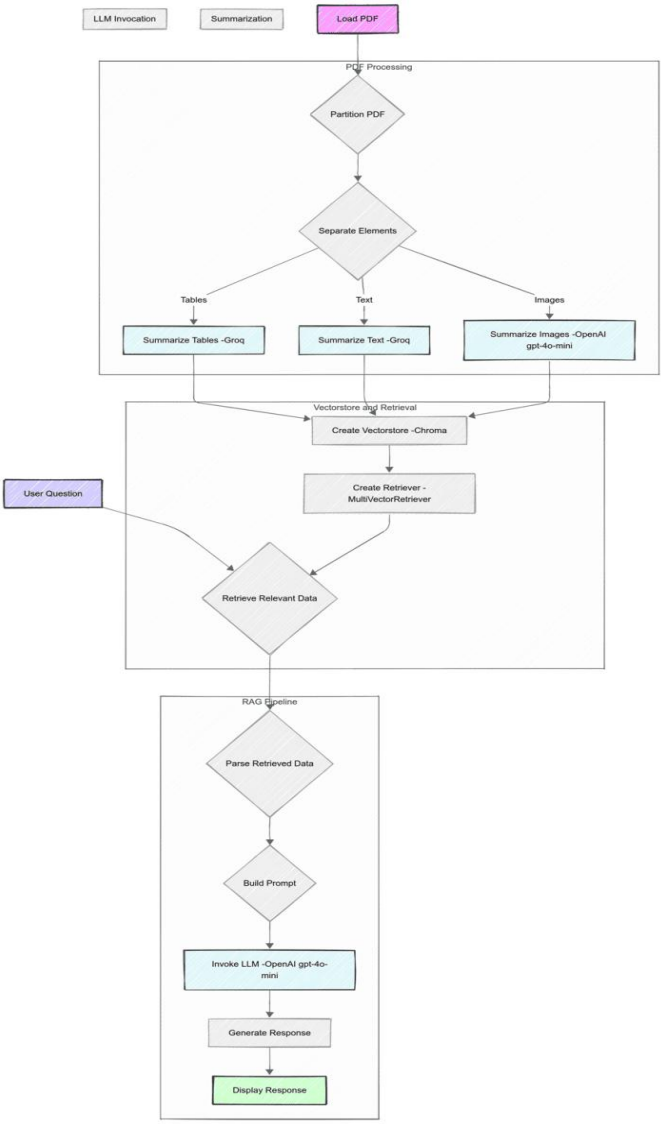
This paper presents LangChain-Chat with My PDF, a system designed to revolutionize document interaction by enabling natural language question-answering directly from PDF files. Users often face challenges navigating extensive digital documents in fields such as law, academia, healthcare, and corporate business. Traditional search methods, limited to keyword matching, frequently yield incomplete or irrelevant results. This system addresses these limitations by employing Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs) to provide precise, contextually relevant answers through conversational queries.

By leveraging the LangChain framework, the system integrates LLMs, vector databases, and natural language processing (NLP) utilities to enable efficient information retrieval. Key technologies include models like GPT and BERT for natural language understanding, and vector databases such as Pinecone or FAISS for fast similarity-based search. The RAG approach enhances accuracy by retrieving relevant text snippets before generating coherent responses.

This solution offers broad applications across multiple domains. Legal professionals can quickly access key clauses and case details, researchers can extract insights from academic papers, and healthcare practitioners can locate critical patient information or clinical guidelines. Additionally, corporate teams can efficiently retrieve policy details, financial records, or training material, improving overall productivity.

By combining advanced retrieval and generation techniques, this system significantly enhances information accessibility, making complex documents easier to navigate and reducing the time spent searching for crucial details.

3. ARCHITECTURE



The proposed system presents an efficient Retrieval-Augmented Generation (RAG) pipeline for processing PDFs and answering user queries using a Large Language Model (LLM). The workflow begins with loading a PDF, which is then partitioned into its constituent elements, including text, tables, and images. Each element undergoes specialized summarization: tables and text are summarized using Groq, while images are processed using OpenAI's gpt-4o-mini. The extracted and summarized data is then stored in a vector database (Chroma), allowing for efficient retrieval. A MultiVectorRetriever is employed to fetch relevant information based on a user's query, ensuring accurate and contextually relevant responses. The retrieved data is then parsed, formatted into a structured prompt, and processed by the OpenAI gpt-4o-mini model to generate a natural language response. This response is subsequently displayed to the user. By integrating advanced LLM capabilities, vector databases, and retrieval mechanisms, the system provides an intelligent and scalable approach to document processing and question-answering, making it highly useful for research, automation, and enterprise applications requiring efficient knowledge extraction from large documents.

4. TECHNOLOGIES

4.1 Langchain

LangChain is a strong platform created to supercharge Large Language Models (LLMs) by incorporating them with outside information sources, APIs, vector databases, and retrieval processes. LangChain makes it possible for developers to create context-sensitive, intelligent applications that move past plain text generation through the ability of models to interact with documents, databases, and real-world systems.

4.2 Groq

Groq is a next-generation AI computing platform for ultra-fast inference of Large Language Models (LLMs). Unlike conventional GPU-based architectures, Groq employs a new class of AI accelerator that dramatically boosts the speed and efficiency of AI model processing. It is optimized for low-latency and high-performance AI applications and is best suited for real-time interactions.

4.3 Streamlit

Streamlit is an open-source Python library used by developers to create interactive web applications easily and rapidly, particularly for data science and machine learning (ML) projects. Streamlit is extensively utilized to develop user-friendly applications for AI, such as chatbots, data visualizers, and document processors.

4.4 LLMs

Large Language Models (LLMs) are sophisticated artificial intelligence models that are trained on gigantic datasets to read, write, and process human language. Deep learning methods, specifically transformers, are employed by these models to read text, respond to queries, summarize, and even converse in context.

5. RESULT

Our envisioned system is an AI assistant that is built with the help of LangChain and Large Language Models (LLMs) to allow for effective information extraction and understanding from PDF files. The system uses a Retrieval-Augmented Generation (RAG) algorithm, which allows for querying the document to get relevant content and providing precise, contextually sound responses to user questions. The process includes PDF analysis, segmentation of large chunks of its content, and extracting both structured and unstructured data in the form of text, tables, and images. The extracted data is then saved in a vector database (Chroma), enabling quick retrieval. When a query is entered by a user, the system extracts the most pertinent parts of the document, parses the text that has been retrieved, and builds a structured query for the LLM (OpenAI gpt-4o-mini). The model then produces a compact, informative answer, with emphasis on clarity,

accuracy, and direct referencing to the content of the document. The system also offers support for ambiguity resolution and interpretation of intricate technical information, and thus it is an effective machine for automated document analysis, research, and knowledge extraction for use in different academic and business applications.

6. CONCLUSIONS

The "LangChain-Chat With My PDF" initiative enriches interaction with PDFs to make retrieving information more intelligent and efficient. This system breaks down a document into its text, tables, and images and uses LLAMA and OpenAI LLMs to summarize each element. It integrates the Retrieval-Augmented Generation (RAG) framework to retrieve the most pertinent content efficiently to answer queries from users with contextual accuracy.

Built using LangChain, the system seamlessly integrates multiple APIs to facilitate query processing. When a user submits a question, the system retrieves the most relevant information from the summarized data and sends it to OpenAI's API, which generates a coherent, context-aware response. The generated output is then presented in a user-accessible format, such as a terminal display or potential future interfaces.

This project closes the gap between static reports and active, real-time interaction and highly enhances exploration of documents, extraction of knowledge, and automated summarization. Enhancements in the future, such as a graphical user interface (GUI), support for multiple documents, and better retrieval accuracy, may further enlarge accessibility, scalability, and efficiency, making it an effective tool for academic research, business applications, and document intelligence solutions.

ACKNOWLEDGEMENT

We would like to extend our heartfelt thanks to the LangChain community and the developers for a strong framework that facilitated easy integration of Large Language Models (LLMs) to facilitate smart document interaction. Our thanks also go to OpenAI and LLAMA for their highly capable language models, which significantly contributed to query response accuracy and efficiency.

We also recognize the work of the Retrieval-Augmented Generation (RAG) framework, which greatly enhanced the performance of the system to retrieve useful information from PDFs. Many thanks to the open-source community and researchers whose relentless innovations in Natural Language Processing (NLP) and document processing have shaped this effort.

Lastly, we would like to express our deepest gratitude to our peers, mentors, and fellow colleagues for the useful feedback, encouragement, and support during the course of preparing this project. Their advice has been crucial in sharpening and developing this study.

REFERENCES

[1] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N & Riedel, S. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS).
[2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All You Need. In Advances in Neural Information ProcessingSystems(NeurIPS).
3] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805.
[4] Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W.-T. (2020). Dense Passage Retrieval for Open-Domain Question Answering. arXiv preprint arXiv:2004.04906.
[5] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., ... & Amodei, D. (2020). Language Models are Few-Shot Learners. arXiv preprint arXiv:2005.14165.
[6] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv preprint arXiv:1301.3781.
[7] Pinecone Documentation. (n.d.). Vector Database for Machine Learning and NLP Applications. Pinecone.io.
[8] Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving Language Understanding by Generative Pre-Training. OpenAI.
[9] Google AI Blog. (2022). Understanding Semantic Search with Dense Vectors.
[10] Schuster, T., Gupta, V., Lewis, P., & Riedel, S. (2021). SLING: A Modular and Extensible Framework for Creating Information Retrieval and NLP Applications. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval.
[11] LangChain Documentation. (n.d.). LangChain - Building Applications with Language Models.
[12] OpenAI Documentation. (n.d.). Using GPT-3 and GPT-4 for NLP Applications. OpenAI.