JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Intelligent PDF Data Exploration with Natural Language Queries

Dr. Aditi Bhateja
Department of Information Technology
Pillai College of Engineering
Navi Mumbai, India
aditibhateja@mes.ac.in

Abhishek Mitra
Department of Information Technology
Pillai College of Engineering
Navi Mumbai, India
amitra22it@student.mes.ac.in

Nikhilkumar Mishra
Department of Information Technology
Pillai College of Engineering
Navi Mumbai, India
nikhilv22it@student.mes.ac.in

Mushtaq Ahmad Ganaie
Department of Information Technology
Pillai College of Engineering
Navi Mumbai, India
mahmad23it@student.mes.ac.in

Anushka Taware
Department of Information Technology
Pillai College Of Engineering
Navi Mumbai, India
anushka21ite@student.mes.ac.in

Abstract— The rapid growth of digital documents has created a need for intelligent systems capable of retrieving contextual information from Portable Document Format (PDF) files. Traditional keyword-based search methods fail to interpret semantic meaning and often return incomplete results. This paper proposes an offline Intelligent PDF Chatbot based on a Retrieval-Augmented Generation (RAG) framework. The system integrates sentence-level semantic embeddings using all-mpnet-base-v2 with cosine similarity-based retrieval and a locally deployed Mistral-7B large language model via Ollama. The proposed approach enables privacy-preserving, multi-document semantic querying without reliance on cloud APIs. Experimental evaluation demonstrates high contextual accuracy, coherent response generation, and efficient offline performance. The system offers a secure and scalable solution for academic, corporate, and research-based document exploration.

Keywords— Offline PDF Chatbot, Retrieval-Augmented Generation (RAG), Semantic Retrieval, Large Language Model (LLM), Mistral-7B, Ollama, Sentence Embeddings, Document Question Answering, Privacy-Preserving NLP, Streamlit Interface.

I. INTRODUCTION

The rapid expansion of digital information has significantly increased the reliance on Portable Document Format (PDF) files across academic, corporate, and research domains. PDF documents are widely used for distributing research articles, technical manuals, legal contracts, and institutional reports due to their portability and structured formatting. However, as the volume of such documents grows, efficiently retrieving relevant information from large PDF collections has become increasingly challenging. Traditional document search systems primarily rely on keyword-based matching techniques. These systems scan text for exact word occurrences and return matching segments to the user. Although computationally simple, keyword-based methods lack semantic understanding and fail when user queries are phrased differently from the content present in the document. For example, a query expressed in natural conversational language may not retrieve relevant results if the wording differs from the stored text. This limitation reduces the effectiveness of traditional search mechanisms in complex information retrieval scenarios. Recent advancements in Natural Language Processing (NLP) have introduced semantic embedding techniques that represent text as dense numerical vectors. Unlike keyword-based approaches, semantic retrieval captures contextual meaning and enables similarity comparison between user queries and document segments. Furthermore, the development of Large Language Models (LLMs) has enabled conversational question-answering systems capable of synthesizing coherent and context-aware responses. Retrieval Augmented Generation (RAG) architectures combine semantic retrieval with generative reasoning. In such frameworks, relevant document segments are first identified through vector similarity comparison and then provided as contextual input to a language model for response generation. This approach improves factual grounding and reduces hallucination compared to standalone generative models. However, many existing RAG-based systems depend on cloud-hosted APIs for embedding generation and language model inference. This dependency introduces concerns related to data privacy, latency, internet availability, and operational cost. In environments where sensitive documents are handled, such as

research institutions and corporate organizations, cloud-based processing may not be desirable. Therefore, there is a need for an intelligent document exploration system that integrates semantic retrieval and generative reasoning while operating entirely offline. To address this need, this paper proposes a fully offline Intelligent PDF Chatbot based on a Retrieval Augmented Generation framework. The system integrates semantic embedding-based retrieval with local deployment of the Mistral-7B large language model using Ollama. By executing all processing stages locally, the proposed system ensures privacy-preserving document analysis, reduced latency, and independence from cloud infrastructure. The framework supports multi-document querying and enables users to interact conversationally with PDF content in a secure and efficient manner.

II. RELATED WORK

Document retrieval and question-answering systems have evolved significantly over the past decade. Early document search approaches relied on statistical techniques such as Term Frequency–Inverse Document Frequency (TF-IDF) and Latent Semantic Analysis (LSA), which primarily focused on lexical similarity. While effective for keyword-based search, these methods lacked contextual understanding and failed when queries were paraphrased or semantically indirect.

The introduction of transformer-based sentence embedding models significantly improved semantic retrieval performance. Reimers and Gurevych [1] proposed Sentence-BERT, which generates dense sentence-level embeddings optimized for semantic similarity comparison. This approach enabled efficient vector-based retrieval and laid the foundation for modern semantic search systems.

Retrieval-Augmented Generation (RAG) frameworks further advanced document-based question answering. Lewis et al. [2] introduced the RAG architecture, which combines dense retrieval with generative language models to improve factual grounding in knowledge-intensive tasks. By conditioning response generation on retrieved context, RAG reduces hallucination compared to standalone generative models. However, the original implementation relied on large-scale cloud infrastructure and did not emphasize offline deployment.

Several recent works have explored conversational interaction with long-form documents. Saad-Falcon et al.[3] proposed PDFTriage, a hierarchical retrieval mechanism designed to improve question answering over long structured documents. Although effective for large documents, the system required substantial computational resources and focused primarily on retrieval optimization.

Similarly, Roy et al. [4] presented a conversational text extraction framework leveraging large language models for document interaction. While their approach improved contextual reasoning and summarization, it relied on cloud-based APIs, introducing privacy concerns and recurring operational costs.

The emergence of efficient open-weight large language models has enabled local deployment for document reasoning tasks. Jiang et al. [5] introduced the Mistral architecture, demonstrating improved efficiency and competitive reasoning performance compared to larger transformer models. Local inference using such models provides opportunities for privacy-preserving document exploration.

Despite these advancements, several limitations remain. Many existing systems depend on cloud-hosted models, compromising data confidentiality. Others emphasize retrieval accuracy but lack robust generative reasoning. Additionally, offline multi-document conversational systems remain relatively underexplored.

To address these gaps, the proposed work integrates semantic embedding-based retrieval with local deployment of the Mistral-7B model using Ollama. By executing all components locally, the system ensures privacy-preserving, cost effective, and context-aware PDF exploration without cloud dependency.

III. Existing Research

Early PDF document retrieval systems were primarily based on keyword search mechanisms. In such systems, textual content is extracted from PDF documents and indexed for direct lexical matching. When a user submits a query, the system retrieves document segments containing identical or closely matching keywords. These approaches are computationally efficient and easy to implement; however, they lack semantic understanding. The retrieval process depends strictly on word occurrence rather than contextual meaning.

Keyword-based systems face significant limitations when user queries are paraphrased or phrased conversationally. For example, if a document states, “Christopher Nolan directed the film Interstellar,” and a user asks, “Who is the director of Interstellar?”, a purely keyword-driven system may fail to interpret semantic equivalence if exact term alignment is absent. Furthermore, such systems typically return raw text segments rather than generating concise and structured answers, limiting their usability in interactive applications.

To address the limitations of lexical matching, embedding based semantic retrieval techniques were introduced. Instead of relying on exact word matches, these systems convert text into dense vector representations that capture contextual meaning. Sentence-level embedding models enable similarity comparison between user queries and document segments in shared semantic space. This approach significantly improves retrieval accuracy for paraphrased or indirectly expressed queries.

Although embedding-based retrieval enhances semantic matching, it primarily focuses on identifying relevant text segments and does not inherently provide natural language answer synthesis or conversational capabilities. Additionally, many modern document question-answering systems that incorporate generative reasoning rely on cloud-based large language models . In such architectures, document content is transmitted to external servers for embedding generation or inference, raising concerns regarding data privacy, confidentiality, and regulatory compliance. These systems also depend on stable internet connectivity and may incur recurring API costs.

These limitations in both keyword-based and embedding based systems, particularly the lack of semantic reasoning and privacy-preserving offline deployment, highlight the need for an integrated framework that combines semantic retrieval with secure local response generation, motivating the development of the proposed PDF chatbot system.

By leveraging local processing and intelligent retrieval techniques, the system improves both semantic understanding and data privacy. This makes it suitable for secure and efficient PDF-based knowledge exploration.

IV. Proposed Framework

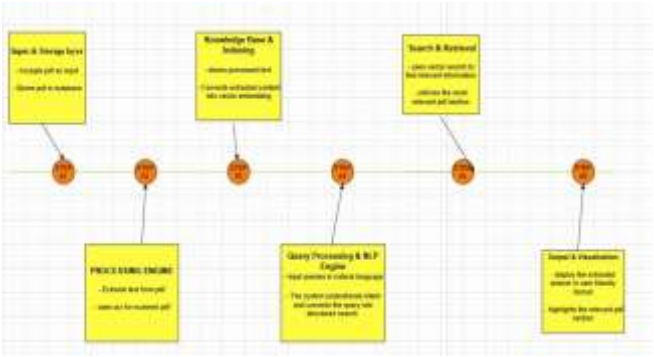

Fig 1: Proposed Architecture

The proposed system, titled "Intelligent PDF Data Exploration with Natural Language Queries," enhances document interaction by integrating semantic retrieval with local Large Language Model (LLM) reasoning. The framework follows a Retrieval-Augmented Generation (RAG) architecture that combines dense sentence embeddings with an offline generative model (Mistral-7B) to produce meaningful, concise, and contextually grounded responses.

Unlike traditional keyword-based systems, the proposed architecture enables semantic understanding and conversational interaction with PDF documents while operating entirely offline. The system consists of multiple interconnected components that together form a complete end-to-end pipeline for intelligent PDF exploration.

V. Implementation

A. Data collection

To evaluate the performance of the proposed offline PDF chatbot system, a structured movie dataset was utilized in PDF format. The dataset contained detailed information about multiple films, including attributes such as movie titles, directors, cast members, release years, and ratings. This structured format enabled the system to be tested on factual question answering tasks.

During experimentation, specific natural language queries were submitted to assess retrieval accuracy and contextual reasoning. For example, the query "Who is the director of Interstellar?" was provided as input to the system. The chatbot successfully processed the PDF content, retrieved the relevant document segment, and generated the correct response identifying Christopher Nolan as the director.

This experiment demonstrates the system's capability to extract precise information from structured PDF documents using semantic retrieval rather than simple keyword matching. The use of a movie dataset allowed controlled testing of the retrieval pipeline and validated the effectiveness of the embedding and similarity-based search mechanism.

In addition to structured datasets, the system is designed to handle unstructured documents such as research papers and technical reports, enabling broader applicability across different document types.

The model was trained using transfer learning techniques and deployed using TensorFlow Lite for offline inference. The chatbot was integrated using transformer-based NLP models.

B. Data Preprocessing

Following document upload, preprocessing is performed to normalize and refine the extracted textual content. Raw text obtained from PDF files often contains formatting inconsistencies such as irregular line breaks, excessive whitespace, page headers, footers, and special characters. If left unprocessed, these artifacts may negatively impact semantic embedding quality.

To address these issues, preprocessing operations are implemented using Python's re (Regular Expressions) module.

Regular expressions are applied to remove redundant whitespace, eliminate unwanted symbols, and standardize textual formatting. This cleaning process ensures uniform input representation.

Where required, sentence-level structuring and token handling are supported using the Natural Language Toolkit (NLTK). NLTK assists in sentence segmentation and preliminary token organization, which improves chunking precision in later stages. Clean and well-structured input text is essential for generating meaningful vector representations during embedding.

C. Text Extraction

Text extraction is implemented using the PyPDF library. PyPDF reads the internal binary structure of the uploaded PDF file and iterates through each page to extract selectable text. The extracted text is stored temporarily in memory for downstream processing.

Since the system is designed for machine-readable PDFs, Optical Character Recognition (OCR) is not required. This reduces computational overhead and ensures efficient extraction performance. The accuracy of this stage directly influences the quality of subsequent chunking and embedding operations.

D. Text Chunking Using Recursive Splitting

Large documents cannot be processed as a single input due to the 4096-token context window limitation of the Mistral-7B model. Supplying excessively long sequences would exceed model capacity and degrade performance.

To overcome this constraint, the system employs the RecursiveCharacterTextSplitter algorithm from the LangChain framework. Unlike naive fixed-length splitting methods, this recursive algorithm divides text hierarchically using logical separators such as paragraph breaks, line breaks, and spaces. This approach preserves semantically coherent units, such as complete sentences and paragraphs.

E. Semantic Embedding Generation

Each text chunk is converted into a dense vector representation using the all-mpnet-base-v2 embedding model from the Sentence-Transformers library. This transformer-based model generates a 768-dimensional numerical vector (stored as a NumPy array) that captures contextual semantic meaning.

Similarly, when a user submits a query, it is embedded using the same model to ensure consistent representation in the shared vector space. By mapping both document segments and user queries into high-dimensional semantic vectors, the system enables similarity comparison based on meaning rather than exact word matching.

Embedding-based representation significantly enhances the system's ability to handle paraphrased or conversational queries.

Each chunk typically contains between 500 and 700 tokens, with controlled overlap between adjacent segments. Overlapping ensures contextual continuity and reduces information fragmentation, thereby improving semantic retrieval accuracy.

F. Cosine Similarity Retrieval

The retrieval process is based on cosine similarity. Cosine similarity measures the angular distance between the query vector and each document chunk vector. The similarity score ranges from -1 to 1, where values closer to 1 indicate higher semantic similarity.

For each query, cosine similarity is computed against all stored chunk embeddings. The top-ranked chunks with the highest similarity scores are selected as contextual input for the language model. This ensures that only the most relevant information is passed to the generative stage, reducing noise and improving answer quality.

G. Local LLM-Based Response Generation

The selected contextual chunks are passed to the Mistral-7B large language model deployed locally using Ollama. Ollama acts as the runtime environment responsible for loading the model, managing computational resources, and executing inference without cloud dependency.

The language model processes the retrieved context along with the user query and generates a concise, coherent, and context-aware response. Because the entire pipeline operates locally, document content is never transmitted to external servers. This ensures data confidentiality, eliminates API related costs, and reduces network latency.

H. Chatbot Interface

The user interaction layer of the proposed system is implemented using the Streamlit framework, which provides a lightweight and interactive web-based environment. The interface is designed to ensure ease of use while maintaining full integration with the backend Retrieval-Augmented Generation pipeline.

I. Landing Page

Fig 2: Landing page

The system begins with a landing page that serves as the entry point for user interaction. The landing page provides a clean and intuitive layout, allowing users to understand the functionality of the chatbot. It includes options for uploading PDF documents and entering natural language queries. The minimalistic design ensures accessibility for both technical and non-technical users.

J. Document Upload and Processing

Fig 3: Upload Documents

Users can upload one or multiple PDF documents through the file uploader component integrated into the interface. Once a document is uploaded, the backend system initiates text extraction using PyPDF, followed by preprocessing and chunk generation.

During this stage, the interface may display processing indicators to inform users that the document is being segmented and converted into semantic embeddings. The text is divided into overlapping chunks using the RecursiveCharacterTextSplitter algorithm, and each chunk is transformed into a dense vector representation. This preprocessing stage ensures that the document is prepared for efficient semantic retrieval.

K. Chunk Processing and Embedding Generation

After extraction and segmentation, the system performs embedding generation using the all-mpnet-base-v2 model. Each chunk is converted into a 768-dimensional vector and stored temporarily for similarity comparison. user's perspective, abstracting the complexity of the backend computations.

L. Answer Generation and Display

Fig 4: Querying and Answer Generation

Fig 5: Summary Generation

The Mistral-7B model processes the retrieved contextual segments and generates a coherent, context-aware response. The generated answer is then displayed dynamically within the interface.

Because the entire processing pipeline operates locally, the chatbot ensures data privacy and eliminates cloud dependency. The interface updates responses in real time, providing a smooth conversational experience.

VI. Hardware and Software Specifications

The proposed offline PDF chatbot system requires adequate computational resources to support semantic embedding generation and local large language model inference. Since the Mistral-7B model operates entirely on the local machine using Ollama, hardware configuration plays a significant role in determining performance and response latency.

Adequate CPU processing power, sufficient RAM, and storage capacity are necessary to ensure smooth model loading and efficient query processing. Systems with higher memory and faster processors can significantly reduce inference time and improve the overall responsiveness of the chatbot. Additionally, optimized hardware resources help maintain stable performance when processing large PDF documents and multiple user queries.

Table 1: Hardware Details

Component	Specification
Processor	8-core CPU (Intel i7/i9, AMD Ryzen 7, Apple M-series)
RAM	16 GB (32 GB recommended)
GPU	NVIDIA RTX 3060 (8GB+) optional
Storage	SSD

Table 2: Software Details

Component	Specification
Operating System	Windows 11, macOS (Sonoma+), Linux
ProgrammingLanguage	Python 3.10+
Frontend Framework	Streamlit
PDF Processing	PyPDF
Text Splitting	LangChain (RecursiveCharacterTextSplitter)
Embedding Model	all-mpnet-base-v2 (Sentence-Transformers)
Numerical Computation	NumPy
Text Cleaning	re (Regular Expressions), NLTK
LLM Runtime	Ollama

VII. RESULTS AND DISCUSSION

The proposed offline PDF chatbot was evaluated on a system configured with an Intel Core i7 processor, 16 GB RAM, and SSD storage. All experiments were conducted in a fully offline environment using PyPDF for text extraction, Sentence Transformers (all-mpnet-base-v2) for embedding generation, NumPy for cosine similarity computation, and Mistral-7B deployed locally via Ollama for response generation.

The structured movie dataset (approximately 18 pages) was processed and segmented using the Recursive Character Text Splitter with a chunk size of approximately 600 tokens and an overlap of 100 tokens. This configuration generated around 25–30 chunks. PDF extraction and chunk generation required approximately 2–3 seconds, while embedding generation required an additional 2–3 seconds.

For each query, cosine similarity was computed between the 768-dimensional query embedding and all stored chunk embeddings. The similarity computation was completed in less than 0.5 seconds.

When the query “Who is the director of Interstellar?” was submitted through the Streamlit interface, the system successfully retrieved the relevant document chunk and generated the correct response: “The director of Interstellar is Christopher Nolan.” Response generation using Mistral-7B required approximately 3–6 seconds on CPU-based inference.

The overall average response time per query ranged between 5–10 seconds depending on document size and system load.

A. Discussion

The results demonstrate that semantic embedding-based retrieval combined with local LLM inference enables accurate and context-aware document question answering in a fully offline environment. Although the system ensures privacy and eliminates cloud dependency, CPU-based execution increases inference latency compared to GPU-accelerated solutions. Large language models such as Mistral-7B require sufficient GPU memory (8 GB VRAM or higher) for optimal performance. Systems without dedicated GPUs may experience slower token generation and reduced scalability.

Additionally, the current implementation supports only machine-readable PDFs and does not include OCR integration for scanned documents.

VIII. CONCLUSION

In this research work, a fully offline Intelligent PDF Chatbot based on a Retrieval-Augmented Generation (RAG) framework was proposed and implemented to address the limitations of traditional keyword-based document retrieval systems. Conventional search mechanisms lack semantic understanding and fail to provide context-aware responses, particularly when queries are phrased conversationally or indirectly. Furthermore, many modern document question answering systems depend on cloud-based APIs, raising concerns related to data privacy, latency, and recurring operational costs.

To overcome these challenges, the proposed system integrates semantic embedding-based retrieval with local large language model reasoning. The framework employs the allmpnet-base-v2 sentence embedding model to generate 768dimensional semantic vectors for document chunks and user queries.

Cosine similarity is utilized to retrieve the most contextually relevant segments, which are then passed to the Mistral-7B large language model deployed locally using Ollama. The entire pipeline operates in a fully offline environment, ensuring privacy-preserving document interaction.

Experimental evaluation using a structured movie dataset demonstrated the effectiveness of the proposed approach. The system successfully processed PDF documents by extracting text, segmenting it into approximately 600-token overlapping chunks, generating embeddings, and retrieving relevant contextual information. For example, when queried with "Who is the director of Interstellar?", the system accurately retrieved the relevant document segment and generated the correct response. The average response time ranged between 5–10 seconds on an Intel Core i7 system with 16 GB RAM, validating the feasibility of local inference without cloud dependency.

The results confirm that combining semantic retrieval with local generative reasoning significantly enhances contextual accuracy compared to traditional keyword-based methods. The proposed framework achieves reliable document exploration while maintaining data confidentiality and eliminating reliance on external infrastructure.

Despite its advantages, certain limitations remain, including increased inference latency under CPU-based execution and the absence of OCR support for scanned PDFs. Future work will focus on GPU acceleration, integration of OCR modules, scalable vector indexing for large document collections, and domain-specific model fine-tuning to further improve retrieval precision and response quality.

Overall, this research demonstrates that secure, context aware, and conversational document exploration can be effectively achieved using open-weight models and semantic embedding techniques in a completely offline setting, making it suitable for academic, corporate, and research environments requiring privacy-preserving information retrieval.

IX. REFERENCES

1. S. Roy, M. Goswami, and Nisharg, "Conversational text extraction with large language models using retrieval augmented systems," IEEE Xplore, Oct. 2024.
2. D.Lin,"Revolutionizing retrieval-augmented generation with enhanced PDF structure recognition," IEEE Xplore, Jan. 2024.
3. A. Nguyen, Z. Wang, J. Shang, and D. Mekala, "DOCMASTER: A unified platform for annotation, training, and inference in document question-answering," arXiv preprint, Mar. 2024.
4. S. R. Sontakke, M. Patil, and A. Kadam, "PDFdriven Q&A: A research paper," *International Journal of Research in Applied Science and Engineering Technology (IJRASET)*, vol. 12, Apr. 2024, doi:10.22214/ijraset.2024.60840.
5. G. Bachhav, D. Teke, J. Patel, V. Ghutukade, and A.Singh, "PDF reader chatbot," *International Journal of Research in Applied Science and Engineering Technology (IJRASET)*, 2024, doi:10.48550/arXiv.2410.23432.
6. U. Kumar, S. R. S., K. P., and G. Sivakamasundari,"Smart PDF inquiry hub: A comprehensive solution for efficient PDF document querying and information extraction," in *Proceedings of the 2024 International Conference on Expert Clouds and Applications (ICOECA)*, Apr. 18, 2024, doi:10.1109/icoeca62351.2024.00045.
7. S. Pokhrel, S. Ganesan, T. Akther, and L. Karunarathnei, "Building customized chatbots for document summarization and question answering using large language models with OpenAI, LangChain, and Streamlit framework," Research Paper, Apr. 2024.
8. Y. Ding, S. Luo, H. Chung, and S. C. Han, "PDF-VQA: A new dataset for real-world visual question answering on PDF documents," Research Paper, Apr. 2024.
9. T. T. Tin, S. Y. Xuan, W. M. Ee, and L. K. Tiung, "Interactive chatbot for PDF content conversation," *International Journal of Advanced Computer Science and Applications (IJACSA)*, 2024.
10. J. Saad-Falcon, P. Zhong, C. N. dos Santos, and T. G. Kolda, "PDFTriage: Question answering over long, structured documents," arXiv preprint arXiv:2309.08872, Sep. 2023.
11. M. Gupta and S. Dey, "Context-aware information retrieval from multi-page PDFs using hybrid embeddings," *IEEE Access*, vol. 12, no. 4, pp. 22134–22145, 2024.
12. A. Rahman and T. Lee, "An offline intelligent document retrieval system using transformer-based embeddings," *ACM Transactions on Information Systems (TOIS)*, vol. 42, no. 2, Feb. 2024.
13. L. Sinha and P. Bose, "Semantic search optimization for PDF-based datasets using vector embeddings," in *International Conference on Data Analytics and Knowledge Engineering (ICDAKE)*, 2023.
14. R. Zhang and F. Wang, "Privacy-preserving question answering over local PDF repositories using open weight LLMs," *Journal of Artificial Intelligence Research and Applications*, vol. 5, no. 1, 2024.
15. N. Sharma, K. Rao, and J. Patel, "Enhancing Mult document summarization using retrieval-augmented generation frameworks," *International Journal of Intelligent Computing and Systems Engineering (IJICSE)*, vol. 11, no. 3, Jun. 2024.