STM JOURNALS

International Journal of
Computer Science Languages

ISSN: 3048-944X
Volume 2, Issue 2, 2024
July–December
DOI (Journal): 10.37591/IJCSL

https://journals.stmjournals.com/ijcsl

Review IJCSL

Advanced Document Query System

Jyoti Jayesh Chavhan[1,*], Shoaib Hafiz Shaikh[2], Suyash Kiran Ayachit[2], Sheshasai Arjun Dusa[2], Nivedh Vijay Anakakkil[2]

Abstract

For knowledge-intensive businesses, large document libraries contain a plethora of information. Massive, chaotic, collections of documents that are unstructured have emerged from this rapid increase. Even if accessing or storing these documents has become easier, finding the required critical information in these vast document collections has become more difficult. NLP (natural language processing) is one of the techniques used to retrieve the information that is required from a huge chunk of data. The proposed system uses cutting-edge methods for document comprehension and NLP to enable users to upload PDFs, ask questions, and get timely, accurate answers. NLP techniques use the subtle layers of context and semantics, going beyond simple keyword extraction. The interface facilitates easy interaction with complex textual material by providing users with access to a multitude of information. Regardless of your experience level, level of diligence, or role as a multi-disciplinary expert, the platform meets a wide range of needs and fluidly adjusts to meet each user's specific needs. This initiative stands out as a pathfinder as we traverse the digital world, where information is plentiful but frequently difficult to extract. It not only makes it easier to interact with textual content, but it also creates an ecosystem where insights are easily available to everyone. It is evidence of how innovation can create a knowledge environment that is inclusive of all people and productive, paving the way for a genuinely inclusive digital age.

Keywords: NLP, user-friendly, innovation, information retrieval, extraction

INTRODUCTION

Wide-ranging document libraries contain a wealth of important data in a complex world of knowledge-intensive businesses. Rapid data growth has resulted in enormous chaotic collections of unstructured documents, which pose a significant obstacle to effective information retrieval. Although improvements in terms of accessibility and storage have been achieved, the intricacy of searching through these vast archives of documents has increased, making it increasingly difficult to find important information. The abundance of data has fundamentally changed the character of document collection, even though it has made access and storage easier. We will explore many facets of information retrieval in this project, such as search algorithms, indexing, querying, and user interfaces. The main task of the proposed system is to find the necessary information efficiently and rapidly from a variety of data sources. Giving users access to techniques and resources that help them quickly and accurately locate the data they need is the goal. By leveraging the capabilities of cutting-edge artificial intelligence, the system searches for a solution that enables users to extract valuable insights from PDFs effortlessly through simple queries. Through the interface, users can upload PDF documents and quickly ask any queries they may have about the content. The underlying artificial intelligence (AI) engine carefully examines the uploaded PDF,

***Author for Correspondence**
Jyoti Jayesh Chavhan
E-mail: jyotichavhan293@gmail.com

[1]Assistant Professor, Department of Artificial Intelligence Machine Learning, SIES Graduate School of Technology, Nerul, Navi-Mumbai, Maharashtra, India
[2]Student, Department of Artificial Intelligence Machine Learning, SIES Graduate School of Technology, Nerul, Navi-Mumbai, Maharashtra, India

Received Date: June 27, 2024
Accepted Date: August 08, 2024
Published Date: October 07, 2024

Citation: Jyoti Jayesh Chavhan, Shoaib Hafiz Shaikh, Suyash Kiran Ayachit, Sheshasai Arjun Dusa, Nivedh Vijay Anakakkil. Advanced Document Query System. International Journal of Computer Science Languages. 2024; 2(2): 26–35p.

grasps its details, and generates precise answers to the questions asked, thereby providing users with answers that are relevant to their situation. This technology enables users to delve into difficult texts without requiring manual reading and interpretation and streamlines the process of retrieving information from thick and complex papers. Our initiative bridges the gap between human queries and document comprehension, making it a dependable and time-saving tool for professionals, students, researchers, and knowledge seekers in various fields.

Through the utilization of sophisticated natural language processing (NLP) and document understanding methods, the platform opens new avenues for people to study and comprehend difficult documents efficiently and easily. The proposed system seeks to transform how humans engage with textual material by emphasizing efficiency and accessibility, which ultimately leads to a smoother and more fruitful experience in the search for knowledge. The proposed system seeks to transform how humans engage with textual material by emphasizing efficiency and accessibility, which ultimately leads to a smoother and more fruitful experience in the search for knowledge. Its vision is not limited to technological innovation. It aims to create a setting in which engaging with textual material goes beyond the ordinary and transforms into a fulfilling and immersive experience. In summary, the system utilizes its unrelenting dedication to efficiency and accessibility and hopes to lead the way in revolutionizing how we engage with textual information.

LITERATURE REVIEW

Jurafsky and Martin (2020) [1] provide an in-depth examination of basic concepts and more complex subjects in computational linguistics and NLP.

Jurafsky and Martin have contributed significantly to the development of a strong theoretical foundation for NLP, which is essential to our project, "AI Chat-Bot For Your Data." It has provided priceless insights into the complexities of information retrieval, which is a crucial component of our project centered on information extraction from PDF documents.

The underlying understanding of algorithms and approaches applicable to language processing jobs is where the principles of "Speech and Language Processing" are utilized most clearly.

As we examine the literature, Jurafsky and Martin's theoretical frameworks are crucial in determining the course of our investigation. Additionally, our project documentation acknowledges the direct effect on the implementation of algorithms and approaches relevant to our AI Chat-Bot's information extraction capabilities by referencing particular parts and chapters from the book where appropriate.

Bird et al. (2009) [2] focus on real-world NLP applications that use the Natural Language Toolkit (NLTK) and Python programming language. Text processing, language analysis, and machine learning approaches for NLP are only a few of the many areas covered. Bird et al. provided invaluable insights into our project. In particular, the book's advice on Python text processing and linguistic analysis and its discussion of core NLP ideas influenced the creation of our platform. Our AI-powered knowledge extraction platform is more reliable and efficient because we used NLTK for some parts, which is in accordance with the ideas and methods presented in this important study.

Manning et al. (2008) [3] is a seminal work in the field of information retrieval. The book, published by the Cambridge University Press, methodically goes over important ideas, including indexing, retrieval models, and document representation. The theoretical foundations of our project were greatly impacted by this. In our implementation, we used the findings of Manning et al. to improve the efficiency and accuracy of information extraction from PDF documents. The book's ideas on retrieval models have influenced the design of our AI algorithms, resulting in a reliable and effective knowledge-sharing ecosystem. Our project aligns with the concepts established in this authoritative work, including proven approaches from information retrieval research, to provide users with a comprehensive platform for efficient data extraction and analysis.

Rahman et al. (2022) [4] provide a comprehensive overview of the components, tools, and applications of NLP. The authors comprehensively explored the underlying parts and technology that form NLP, providing useful insights into the numerous uses of this topic. Our project is inspired by Rahman and Haque's evaluation, as it incorporates crucial NLP components and technologies to improve user experience. This paper's analysis of applications influences our approach to creating a powerful conversational AI system. By aligning with the ideas offered in this study, our effort incorporates known concepts from a larger area of NLP, helping to enhance information extraction methods from complicated textual data sources.

Irfan et al. (2017) [5] surveyed the use of evolutionary computation techniques to optimize information retrieval processes. This study investigated numerous techniques and methodologies in the field of evolutionary computing to improve the efficiency and efficacy of information retrieval systems. Our project builds on Irfan and Ghosh's survey findings, using evolutionary strategies to improve knowledge retrieval. The study's assessment of ways to improve retrieval procedures influences our project's approach to utilizing sophisticated AI algorithms. Our system intends to offer users accurate and timely replies by combining the evolutionary computation concepts discussed in this study, thereby contributing to the continual evolution of information retrieval platforms.

Ren et al. (2020) [6] present a case study of information retrieval methods that use knowledge-enhanced word embeddings in a dialogue context. This study investigated the use of knowledge augmentation techniques inside word embeddings to improve the precision and relevance of information retrieval.

Our project was inspired by Ren et al.'s case study, especially in the area of knowledge-enhanced word embeddings. Our platform uses powerful AI algorithms to enhance user searches and to increase the accuracy of information extraction from PDF documents. The case study investigation of knowledge-enhanced embeddings supports our project's focus on context-aware replies, resulting in a more efficient and dynamic user experience for information retrieval via conversation.

Dörpinghaus and Stefan (2020) [7] focus on improving retrieval methods in large-scale knowledge networks. While the material supplied is brief, the issue of optimization in retrieval techniques is consistent with our project, "AI Chat-Bot For Your Own Data," which employs advanced AI algorithms to efficiently extract information from PDF documents. Although the specific techniques or algorithms mentioned in this paper are not specified, the emphasis on optimization may provide insights into tactics or approaches relevant to our project's objectives. Further investigation of the paper's content is required to make more exact linkages between its findings and the optimization methodologies used in our AI chatbot. Specific data regarding the optimization methodologies presented in this study would help make a more targeted comparison.

Pan and Zhen (2020) [8] investigate the optimization of information retrieval algorithms in digital libraries, with a particular emphasis on semantic search engines. This paper's examination of information retrieval method improvement is conceptually similar to our project, which focuses on information extraction from PDF documents. While the offered material does not go into detail on specific optimization approaches, the emphasis on semantic search engines may provide insights into ways to improve the efficiency and accuracy of information retrieval. To establish a more specific link, this study must be further examined. Specific data on the semantic search engine's function and optimization tactics would allow for a more complete comparison with the methodologies used in our AI chatbot.

Ceri et al. (2013) [9] dig into the methodology, strategies, and problems of collecting information from internet sources. This scholarly effort provides a starting point for understanding web-based information retrieval systems. This book is especially important, given our ongoing efforts. The insights provided in this book, which include subjects such as indexing, search techniques, and the display of web-derived information, are expected to help our project's goal of extracting information from PDF

documents. Furthermore, the description of user interactions with web-based information systems offers useful insights that can help develop a user-friendly interface for our AI-powered platform.

PROPOSED MODEL

In a world where there is a lot of difficult information in documents, obtaining important details from PDFs is difficult. The systems we currently have are not easy to set up and are expensive, making it difficult for different people to acquire the knowledge they need. Our new system plans to change how people use PDFs by addressing these issues.

Our system is easy for anyone to use and uses smart technology to help extract information quickly. It is made for small businesses and regular people. We were concerned about making it affordable and simple. Our goal was to provide everyone with access to important information, making it easier for everyone to understand documents. Our new system aims to become the best way for people to obtain information, changing how we find and understand things in documents for everyone.

Explaining the components of the system: In this section, we will delve into the core components that constitute the "Advanced Document Query Method" with a focus on how "Vector Embedding, large language models (LLM), and Semantic Search" play a vital role in extracting the required information.

The Document Upload Feature

The document upload feature allows users to upload multiple PDFs that are subsequently processed to extract their textual content. The text retrieved from these PDFs was segmented into smaller units or chunks, forming a large string of text. These individual text chunks were then converted into embedding. This method involves the aggregation of text from uploaded PDFs, transforming it into a unified string, and further segmenting it into smaller analyzable components.

This process can be easily implemented using Python. For the PDF reading function, our system relied on the PyPDF2 Python library. This library offers an 'extract_text()' method that enables the extraction of content from individual pages within a PDF. By iterating through each page and applying this method, we accumulated the extracted text into a variable, creating a comprehensive unified string of text.

To segment this unified text into smaller chunks, our system integrates LangChain, an open-source framework designed for applications utilizing LLMs. Specifically, our implementation utilizes the 'CharacterTextSplitter' method provided by LangChain. This method facilitates the generation of smaller text segments, enabling a more granular analysis of content.

The Chunks Overlapping Concept

The 'CharacterTextSplitter' method encompasses various parameters, and among those utilized in our implementation are 'separator,' 'chunk_size,' and 'chunk_overlap.' The 'chunk_size' denotes the character count within each generated segment, determining the size of individual chunks. On the other hand, 'chunk_overlap' maintains sentence integrity across chunks. This ensures that a new chunk does not begin midway through a sentence by incorporating a repetition of characters from the preceding chunk to kick-start the subsequent one. This repetition size was specified by the 'chunk_overlap' parameter.

This strategy safeguards the continuity of sentences and preserves their semantic meaning. Consequently, it significantly aids the efficacy of the semantic search process by ensuring meaningful chunking without disrupting sentence structures.

Vector Embedding

In our process, after obtaining the segmented text chunks, the subsequent step involves generating an embedding. Embedding refers to the numerical representation of text chunks, enabling storage within vector databases. These embeddings serve as a transformative technique for converting textual data into

numerical forms that encapsulate semantic meanings and relational aspects. They essentially map data entities as coordinates within a multidimensional space and position similar data points closer together. These embeddings are crafted using machine learning algorithms and act as organized representations of unstructured textual data.

Mathematically, vectors are embedded in a high-dimensional space. They are derived using complex algorithms that involve neural networks to capture semantic relationships between words, sentences, or larger textual units. The distance or proximity between points in this high-dimensional space reflects the similarity or dissimilarity between underlying data elements.

Within our system, to create these embeddings, we utilized OpenAI's ADA V2 embedding model. Another commonly used model for this purpose is 'word2vec.' These models employ sophisticated algorithms to convert text chunks into numerical representations.

The generated vector representations, also known as embeddings, are stored in a specialized database referred to as a vector database. In our system architecture, the FAISS (Facebook AI Similarity Search) serves as the chosen vector database. FAISS employs a specialized algorithm, the inverted multi-index, enabling fast and efficient similarity searches across the vast embedding space. This method organizes embedding in a way that enhances search performance by indexing and clustering similar embeddings, facilitating speedy retrieval based on their proximity in the high-dimensional space.

Overall, the process involves converting text chunks into embedding, which are numerical representations that preserve semantic relationships, as shown in Figure 1. These embeddings find storage within a vector database, allowing efficient retrieval and similarity assessments based on their spatial arrangements in the embedding space.

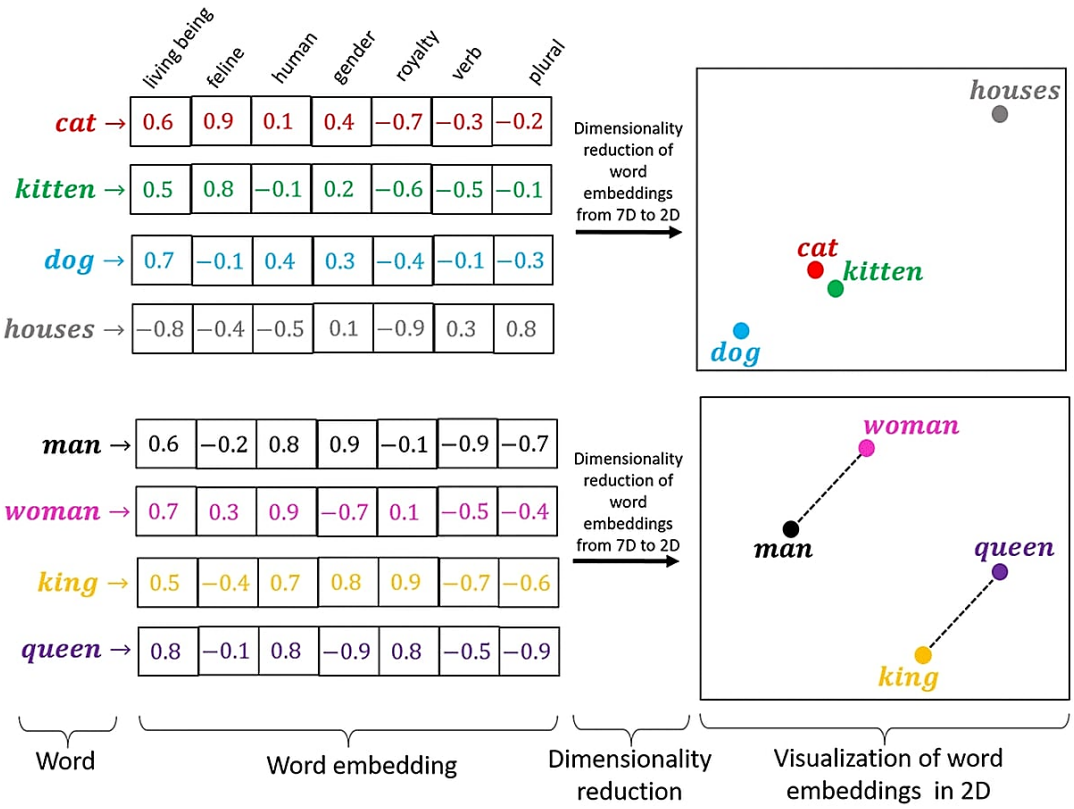

Figure 1. Word embedding.
Source: Gautam H. (2020). Word Embedding: Basics - Hariom Gautam - Medium. [online] Medium. Available from: https://medium.com/@hari4om/word-embedding-d816f643140

The Query System

Within our system's query functionality, users are empowered to pose questions relevant to the content stored in the uploaded PDF documents. These user queries undergo the same embedding conversion process employed earlier to translate text into numerical representations.

To facilitate user interaction, our system features a straightforward chatbot interface that enables document upload and question submission. For user interface (UI) development, we leveraged Python in combination with Streamlit, an open-source Python library renowned for simplifying the creation and distribution of visually appealing, customized web applications tailored specifically for machine learning and data science purposes.

This chatbot interface streamlines user engagement by providing an intuitive platform where users can seamlessly upload documents, ask questions, and receive relevant answers derived from the PDF document content. By integrating Streamlit's capabilities, we aim to enhance user accessibility and interaction, while ensuring an efficient and user-friendly experience within our system.

Similarity Search

Following the user's query transformation into a vector representation, the system initiates a vector similarity search. This process involves comparing the vector representation of a user's question with the stored vector representations within the vector database. The aim is to retrieve pertinent information relevant to a user query.

Vector similarity search operates based on the principle of measuring the similarity or closeness between vectors in a high-dimensional space. In our context, each vector represents the essence of the text and captures the semantic relationships among the chunks of text obtained from the documents.

The technique primarily utilizes similarity metrics, such as cosine similarity or Euclidean distance, to ascertain the resemblance between the query's vector representation and those stored in the database. Cosine similarity measures the cosine of the angle between two vectors and yields a value between -1 and 1, where higher values signify greater similarity. Euclidean distance computes the straight-line distance between vectors, with smaller distances indicating higher similarity.

An example demonstrating this process involves two vectors in a simplified two-dimensional space. Consider vectors A (3, 4) and B (1, 2), as shown below.

The Euclidean distance between these vectors can be calculated as:

$$\text{Distance} = \sqrt{(3-1)^2 + (4-2)^2} = \sqrt{2^2 + 2^2} = \sqrt{8} \approx 2.83$$

For cosine similarity, it is computed as:

$$\text{Cosine Similarity} = \frac{A.B}{||A||||B||} = \frac{(3\times1)+(4\times2)}{\sqrt{3^2+4^2}\times\sqrt{1^2+2^2}} = \frac{11}{\sqrt{25}+\sqrt{5}} \approx 0.97$$

In our system, the user query is converted into a vector representation using the same embedding technique, as shown in Figure 2. This query vector is then compared to all vectors stored in the vector database (representing chunks of text from the PDFs).

Similarity metrics (cosine similarity or Euclidean distance) were computed between the query vector and each stored vector in the database. Vectors with higher similarity scores indicate a closer semantic match to the user query, as shown in Figures 3–5). Finally, the system retrieves the most relevant chunks of text (vectors) from the database, based on the highest similarity scores. These chunks contain information deemed most pertinent to the user query, as shown in Figure 2. This process enables an efficient method to retrieve and present relevant information from the vector database, facilitating accurate responses to user queries based on the similarity of vector representations.

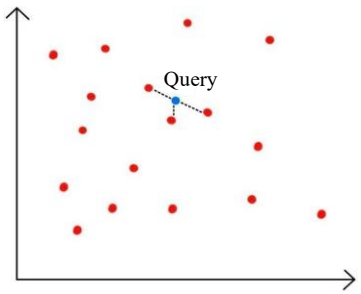

Figure 2. Three nearest neighbors for a query in a vector space.
Source: Tripathi R. (2023). What is Similarity Search? [online] Pinecone.io. Available at: https://www.pinecone.io/learn/what-is-similarity-search/

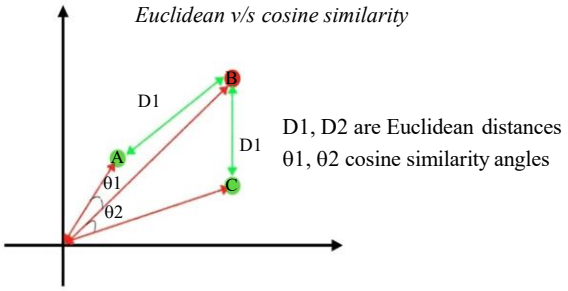

Figure 3. Euclidean versus cosine similarity.

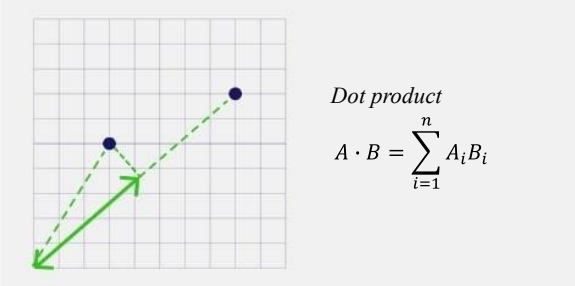

Figure 4. Dot product formula in cosine similarity.

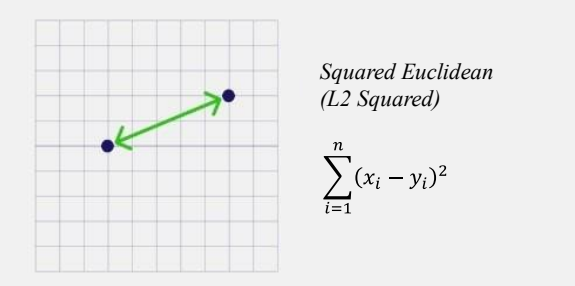

Figure 5. Squared Euclidean distance formula.

Answer Generation

Upon retrieving the relevant text chunks, our system analyzed them to identify the most pertinent information concerning the user's query. This analysis was facilitated by leveraging the capabilities of LLMs to craft concise and informative responses grounded in the context provided by the retrieved text chunks.

LLMs represent a category of AI systems proficient in executing diverse NLP tasks. These models are trained extensively on extensive datasets, hence the term "large." They operate based on machine learning principles, particularly utilizing a neural network architecture known as the transformer model.

LLMs are colossal in size, often spanning tens of gigabytes, owing to their ability to train on colossal amounts of textual data. For example, a one-gigabyte text file can accommodate approximately 178 million words. Upon receiving retrieved text chunks, LLMs harness their learned knowledge, context comprehension, and language generation abilities to generate precise answers relevant to the user's query. This process involves:

- *Contextual understanding:* LLMs grasp the context from retrieved text chunks and assimilate the information to understand the underlying context related to the user query.
- *Language generation:* Using this contextual understanding, LLMs generate responses that are not just syntactically correct, but also contextually appropriate and informative.

For instance, when a user asks a question, LLM processes the relevant text chunks to extract information related to the context of the query. It then employs the learned linguistic patterns, context comprehension, and knowledge representation to generate an accurate and informative answer aligned with the user's query. This process shows how LLMs serve as sophisticated language models capable of understanding context, interpreting information, and generating meaningful responses based on retrieved textual content.

RESULTS AND DISCUSSIONS

The results of our project indicate significant advancement in simplifying the process of information extraction from PDF documents. By integrating advanced AI algorithms and a user-friendly interface, we successfully achieved our goal of streamlining the retrieval of specific insights from complex textual data.

Users have reported a substantial reduction in the time and effort required to extract information compared with existing systems. This efficiency stems from the project's ability to comprehensively analyze PDF documents and provide precise responses to user queries. Moreover, the accessibility and affordability of the project have addressed the limitations of existing systems, making them more user-friendly and cost-effective for a broader user base.

Continuous testing and user feedback are instrumental in fine-tuning the platform, resulting in improved accuracy and user satisfaction. Overall, the results of this project demonstrate the potential of AI to transform the way we interact with textual information, offering an accessible and efficient solution for knowledge extraction from PDF documents, as shown in Figures 6–8.

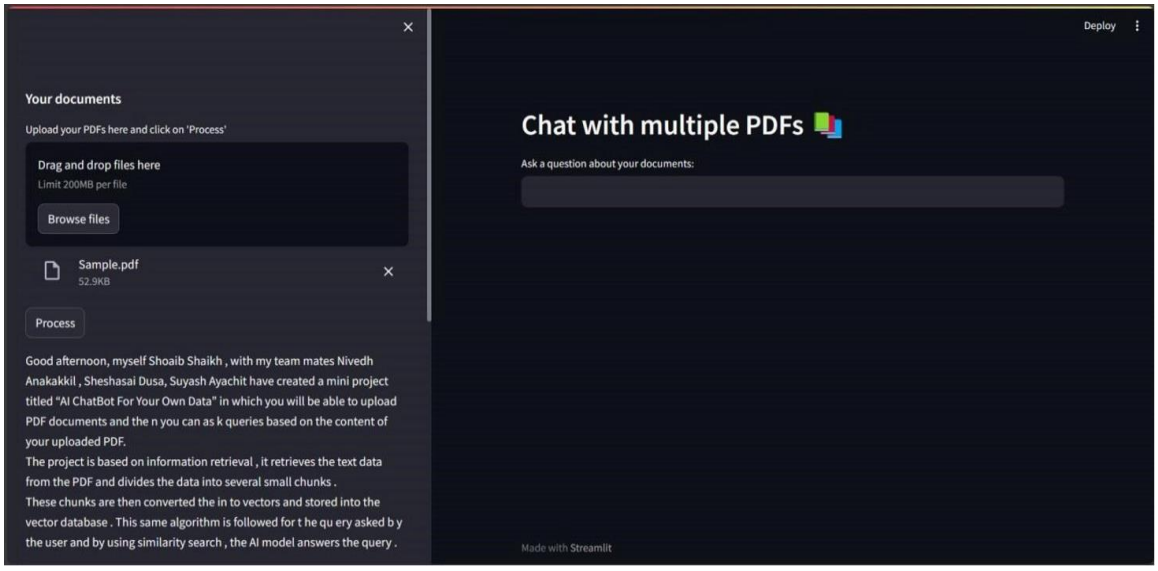

Figure 6. User uploaded a pdf named 'sample.pdf' and the content of the pdf is displayed.

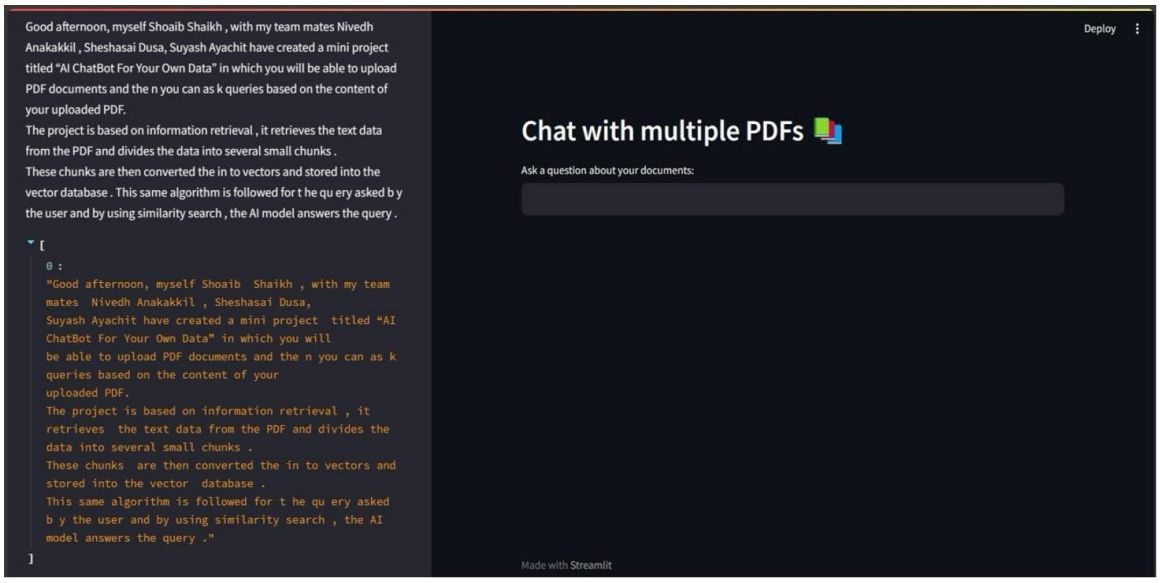

Figure 7. The content is divided into chunks.

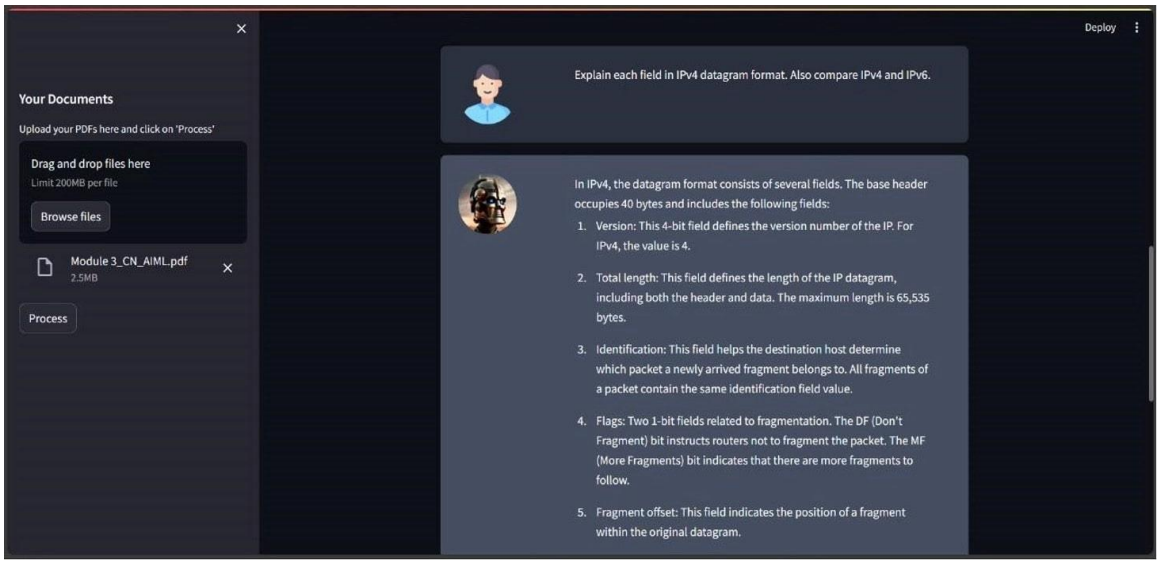

Figure 8. User has asked a question based on the document and the answer is generated by the system.

CONCLUSION

In summary, information extraction from PDF documents has advanced significantly owing to our AI-powered platform. It is positioned as a transformational tool with the potential to change the user experience with complicated textual data because of its powerful AI capabilities and user-friendly interface. This platform can completely change how people engage with and obtain insights from complex texts, including professionals, students, and academics.

The primary strength of our platform is its capacity to close the gap between the comprehension of documents and human questions. This software guarantees a smooth and effective approach for consumers looking for particular information from PDFs by incorporating cutting-edge AI algorithms. Imagine a situation where a researcher must quickly retrieve pertinent information from a long study report. Our technology simplifies this procedure by enabling the user to upload documents, ask questions using natural language, and obtain precise answers. This significantly reduced the time and effort required for such tasks.

Prospective avenues exist for augmenting and broadening the functionality of our platform. First, there is potential for ongoing AI growth, with an emphasis on refining response accuracy and the system's comprehension of intricate inquiries. The platform will remain at the forefront of information extraction technologies owing to this development.

The potential of the platform may also be increased by adding support to document formats other than PDFs. Consider how convenient it would be for users who deal with Word documents or other text-based formats in addition to PDF formats. Increasing interoperability increases the platform's usefulness and accommodates a larger spectrum of user demands, including those from various academic or professional backgrounds. Multilingual support is an additional development area. Global accessibility can be improved by allowing the system to extract information from texts in several languages.

REFERENCES

1. Jurafsky D, Martin JH. Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition with language models. 3rd ed. [Online]. Draft of August 20, 2024. Stanford University; University of Colorado at Boulder; 2024. Available from: https://web.stanford.edu/~jurafsky/slp3/ed3bookaug20_2024.pdf.
2. Bird S, Klein E, Loper E. Natural Language Processing with Python. Sebastopol (CA): O'Reilly Media, Inc.; 2009.

3. Manning CD, Raghavan P, Schütze H. Introduction to Information Retrieval. New York: Cambridge University Press; 2008.
4. Rahman N, Mozer R, McHugh RK, Rockett IRH, Chow CM, Vaughan G. Using natural language processing to improve suicide classification requires consideration of race. Suicide Life Threat Behav. 2022;52:782–91. DOI: 10.1111/sltb.12862. PubMed: 35384040.
5. Irfan S, Ghosh S. Optimization of information retrieval using evolutionary computation: A survey. 2017 International Conference on Computing, Communication and Automation (ICCCA). IEEE; 2017. p. 328–33. DOI: 10.1109/CCAA.2017.8229837.
6. Ren J, Wang H, Liu T. Information retrieval based on knowledge-enhanced word embedding through dialog: A case study. Int J Comput Intell Syst. 2020;13:275–90. DOI: 10.2991/ijcis.d.200310.002.
7. Dörpinghaus J, Stefan A. Optimization of retrieval algorithms on large scale knowledge graphs. 2020 15th Conference on Computer Science and Information Systems (FedCSIS), Sofia, Bulgaria. 2020. pp. 227–36. DOI: 10.15439/2020F2.
8. Pan Z. Optimization of information retrieval algorithm for digital library based on semantic search engine. 2020 International Conference on Computer Engineering and Application (ICCEA), Guangzhou, China. 2020. pp. 364–7. DOI: 10.1109/ICCEA50009.2020.00085.
9. Ceri S, Bozzon A, Brambilla M, Della Valle E, Fraternali P, Quarteroni S. Web Information Retrieval. Germany: Springer Berlin Heidelberg; 2013. DOI: 10.1007/978-3-642-39314-3.

