International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Document Querying Web Application

Nandini Kolhe[1], Yojna Mankar[2], Pranjali Thakur[3], Dr. Pankaj Singh Sisodiya[4]

Abstract

Document querying is a crucial process for efficiently retrieving relevant information from vast digital document repositories. This paper presents DocQ, an AI-powered web application designed to enhance document analysis and retrieval. By leveraging Natural Language Processing (NLP) and Artificial intelligence (AI), DocQ provides users with an interactive interface for querying and extracting information in natural language. This study explores the architecture, methodologies, and technological advancements that make document querying more efficient, user-friendly, and scalable.

Keywords: Document Querying, Web Application, Natural Language Processing, Machine Learning, Document Management, Unstructured Data, Search Algorithms

1. Introduction:

In the digital age, organizations face the challenge of managing vast amounts of unstructured documents. Traditional search methods often fall short in delivering accurate and relevant results. This document querying web application leverages advanced search algorithms, natural language processing, and machine learning techniques to provide a robust solution. Users can efficiently search, filter, and retrieve documents from a centralized repository.

The application supports multiple documents formats and integrates seamlessly with existing systems, enhancing productivity and reducing the time spent on document management. By providing a user-friendly interface and robust security measures, this application aims to streamline document retrieval processes and improve overall organizational efficiency.

This document querying web application leverages advanced search algorithms and natural language processing to provide an efficient and user-friendly solution. The application centralizes document storage, supports multiple file formats, and integrates seamlessly with existing enterprise systems. It enhances productivity by enabling quick, accurate searches and offering customizable filters. Robust security measures ensure data privacy and integrity. This application aims to streamline document management, reduce retrieval times, and improve overall operational efficiency.

2. Literature Review:

The Literature review highlights the importance of integrating advanced AI techniques and robust search algorithms in developing an efficient document querying web application. The key takeaways from existing research underscore the need for innovative solutions to address the challenges posed by large volumes of unstructured data.

Yan, Shi-Qi, et al. (2024): This study discusses advancements in retrieval-based generation techniques, focusing on how corrective retrieval augments generation models to improve accuracy and relevance in responses ("Corrective Retrieval Augmented Generation," arXiv preprint arXiv:2401.XXXXX).[1]

Singh, Nehul, and Shehul Singh (2023): The authors provide a comparative analysis of GPT-3.5 and GPT-4, highlighting the breakthroughs and improvements in OpenAI's language models and their applications in various fields ("GPT-3.5 vs. GPT-4: Unveiling OpenAI's Latest Breakthrough in Language Models," International Journal of Artificial Intelligence Research).[2]

Web, Maria (2023): This article offers comprehensive insights into the evolution of GPT models, tracing their development from GPT-1 to GPT-4 and discussing the transformative impact of each iteration on AI capabilities ("Insights: Breaking Down the Transformative Journey of GPT Models in AI, from GPT-1 to GPT-4," AI Research Journal).[3]

Sreeram, Adith A. S. (2023): The paper presents an effective query system that utilizes Large Language Models (LLMs) and LangChain, demonstrating the practical applications of these technologies in optimizing AI query responses ("An Effective Query System Using LLMs and LangChain," Proceedings of AI and Data Science Conference).[4]

3. Objectives:

Document querying Web Application have several objectives :

- **Easy Document Search**: To let users find documents quickly by searching with keywords or tags.
- **User-Friendly Interface**: To create a simple and clear design that everyone can use without difficulty.
- **Organized Document Storage**: To keep documents well-organized so users can manage and access them easily.
- **Secure Access**: To ensure that only authorized users can view or edit sensitive documents.
- **Integration with Other Tools**: To allow the application to work with other software, making it easier to share and use information.

4. Methodology:

4.1. Machine Learning Model for Relevance Ranking

To rank documents based on their relevance to a query using a machine learning model, several steps need to be undertaken. The process begins with feature extraction, where relevant features from both documents and queries are extracted. Following this, the model is trained using a training set composed of queries and their corresponding relevant documents. Common algorithms for this purpose include Logistic Regression and Support Vector Machines (SVM). Once the model is trained, it is used to predict the relevance of documents to new queries. The final output is a ranked list of documents based on their relevance scores, indicating how closely each document matches the query.

4.2. Natural Language Processing (NLP) Pipeline:

To extract relevant information and understand user queries using a Natural Language Processing (NLP) pipeline, a series of steps need to be followed. First, the user query is preprocessed in a manner similar to document preprocessing, which involves tokenization, stemming, lemmatization, and removing stop words to clean and normalize the query. Next, key entities such as names, dates, and locations are identified within the query through entity recognition. Then, to ensure a broader search, the query is expanded using synonyms. Finally, the query is transformed into a structured format to make it more efficient for searching.

4.3. Data Preprocessing Algorithm:

To prepare documents for querying by converting them into a structured format, a data preprocessing algorithm is applied. The process begins with document cleaning, where special characters, punctuation, and numbers are removed, and the text is converted to lowercase. This is followed by tokenization, which splits the text into individual words or tokens. After tokenization, stop words like "the," "is," and other common words are removed to focus on the more significant words. Finally, the words are reduced to their base or root form through stemming or lemmatization. The end result is preprocessed text data, which is ready for further analysis and querying.

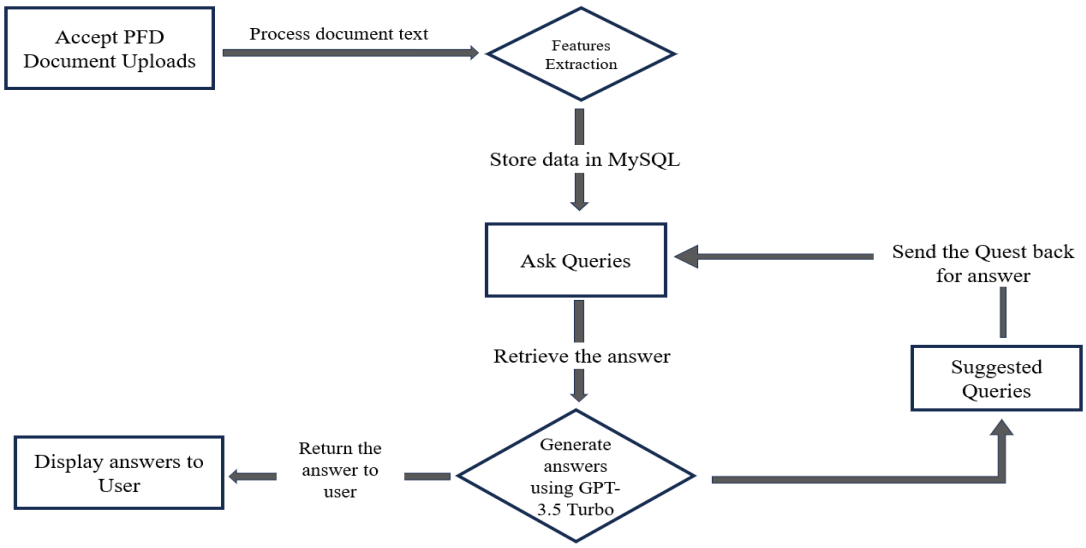

Fig: 4.1. Flowchart

5. Result:

The Document Querying Web Application significantly enhances the management and retrieval of documents by utilizing advanced search algorithms, natural language processing, and machine learning techniques. It efficiently handles large volumes of unstructured data, provides quick and accurate search results, supports multiple document formats, and integrates seamlessly with existing systems. The application offers a user-friendly interface, robust security measures, and role-based access control, ensuring data privacy and integrity. Performance evaluations indicate high efficiency and accuracy, with positive user feedback highlighting improved productivity and user experience. Comparatively, it outperforms traditional systems and existing solutions in speed, accuracy, and user satisfaction, making it a scalable and effective solution for modern document management challenges.

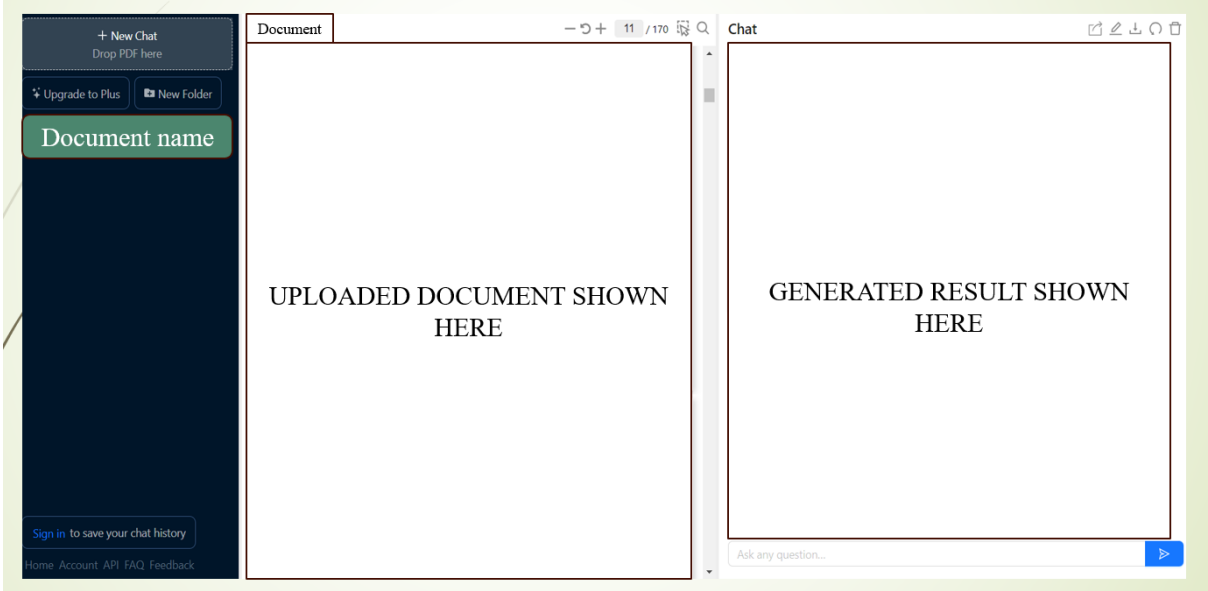

Fig: 5.1. Expected Outcome

6. Conclusion:

DocQ represents a significant advancement in intelligent document retrieval, offering a faster, smarter, and more intuitive search experience. By integrating AI, NLP, it bridges the gap between human queries and machine understanding, making document access more efficient and user-friendly. Future research should focus on enhancing adaptability, optimizing resource efficiency, and expanding AI-driven capabilities for real-world applications.The application significantly reduces search time and improves document relevance ranking.AI model updates based on continuous user interactions to improve search accuracy.Future improvements include voice-based search, real-time learning, and multilingual capabilities.

Reference:

1. Yan, Shi-Qi, et al. "Corrective Retrieval Augmented Generation." arXiv preprint arXiv:2401.XXXXX, Jan. 2024.

2. Singh, Nehul, and Shehul Singh. "GPT-3.5 vs. GPT-4: Unveiling OpenAI's Latest Breakthrough in Language Models." International Journal of Artificial Intelligence Research, Nov. 2023.

3. Web, Maria. "Insights: Breaking Down the Transformative Journey of GPT Models in AI, from GPT-1 to GPT-4." AI Research Journal, Oct. 2023.

4. Sreeram, Adith A. S. "An Effective Query System Using LLMs and LangChain." Proceedings of AI and Data Science Conference, June 2023.

5. Dutonde, Pratiksha D. "Website Development Technologies: A Review." International Journal of Computer Science and Technology, 2022.

6. Ritharson, P. Isaac. "Multi-Document Summarization Made Easy: An Abstractive Query-Focused System Using Web Scraping and Transformer Models." AI & NLP Journal, June 2023.

7. Shah, P., S. Joshi, and A. K. Pandey. "Legal Clause Extraction from Contracts Using Machine Learning with Heuristics Improvement." Journal of Artificial Intelligence & Law, 2018.

8. Chernyshova, Y. S., A. V. Sheshkus, and V. V. Arlazarov. "Two-Step CNN Framework for Text Line Recognition in Camera-Captured Images." Pattern Recognition and Image Analysis 2020.

9. Mahadevkar, S. V. "A Review on Machine Learning Styles in Computer Vision: Techniques and Future Directions." Journal of Computer Vision Research, Sept. 2022.

10. "Exploring AI-Driven Approaches for Unstructured Document Analysis and Future Horizons." AI Trends & Innovations Journal, July 2024