Inspira-Journal of Commerce, Economics & Computer Science (JCECS)
(A National bi-lingual Quarterly Double Blind Open Access Peer Reviewed Refereed Journal)
ISSN : 2395-7069(Print), Impact Factor: (Cosmos 7.650 & I2OR 5.120), Volume 12, No. 04, October-December, 2026, pp. 166-175

FileFlow: Intelligent Document Interaction Powered by Natural Language Processing

Kirti Dahiya*

Assistant Professor, Computer Science and Engineering, Maharaja Surajmal Institute of Technology, New Delhi, India.

*Corresponding Author: kirtidahiya@msit.in

Citation: *Dahiya, K. (2026). FileFlow: Intelligent Document Interaction Powered by Natural Language Processing. Journal of Commerce, Economics & Computer Science, 12(04), 166–175.*

Abstract

In an age when there is a vast amount of different types of data available, effective management and processing of document-based data is essential for many industries. This paper presents a novel AI-based chatbot known as FileFlow, which streamlines interactions with a variety of documents (DOC, PPT, CSV, PDF, web pages, etc.). While our key goal is to provide means for meaningful interactions with other users, we also want to build the capability to work with a large diversity of document types. This synthesis is about creating an AI-powered tool that will help users communicate and work with different types of documents at once. Our goal is to alter the way people work with documents through artificial intelligence. We are going to use innovative technologies, such as natural language processing and machine learning to achieve our goal. Using advanced natural language processing methods, FileFlow decodes the meaning of uploaded documents and answers users' questions accurately. FileFlow uses a combination of document parsing algorithms, machine learning models for understanding and inference, and a user-friendly interface to facilitate human-machine interactions. Evaluations show that FileFlow greatly increases the effectiveness of document-based information retrieval. Through this research, apart from highlighting the functionality of FileFlow, there is an exploration of the applications of the software towards enhancing productivity in academic, corporate, and research environment. Through the integration of these methods, we are developing an efficient system which will enable quick analysis and understanding of various documents. Our aim is to provide simple and useful suggestions for the users.

Keywords: *AI Chatbot, Document Management, Natural Language Processing, Information Retrieval, Document Parsing, User Interface, Semantic Analysis, Query Processing.*

Introduction

The purpose of our project is the creation of a user-friendly utility named FileFlow, which will make interaction with diverse file formats easy, including documents, presentations, spreadsheets, and others. In today's world, it is necessary to have a good system for document management, especially when there is plenty of data all around us. Conventional approaches do not meet requirements and call for something different. AI technologies, specifically those related to natural language processing and machine learning, appear to be a perfect tool for transformation in the area of document interaction. They provide an opportunity to automate routine work, improve searching, and gain valuable insight from documents. The incorporation of NLP and ML technologies will ensure that we have a robust platform that can extract meaning from different documents in an efficient manner.

Copyright © 2026 by Author's and Licensed by Inspira. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work properly cited.

Traditional tools for managing documents are usually based on keyword searching techniques which fail when a user is looking for answers based on context in the presence of a variety of documents. This is because the traditional systems are not able to interpret the human language and the semantics of the texts, hence providing less-than-perfect results. The key challenge here is the creation of a smarter tool that not only works with different documents but can also interpret their contents.

- FileFlow is an innovative AI-powered chatbot which will transform the way in which users can interact with their data contained in documents. The objectives for this study include the following:
- To create a comprehensive system that makes use of sophisticated natural language processing technology to understand the semantic meaning of the data in the document.
- To help the chatbot give accurate, contextual responses to user queries to enhance the user experience.
- To incorporate an adaptive learning algorithm in the chatbot to help increase its efficiency in responding and understanding user queries over time.
- To explore the effects of such a system in practical applications in other domains, including education, businesses, and research.
- To look at ways to increase user engagement and use feedback from the user to make constant improvements to the AI assistant to ensure user satisfaction.

The importance of FileFlow is not limited to just technological advancement; the technology is set to greatly improve the efficiency of information searches through a mechanism of smart analysis and understanding of data in different kinds of documents. It is going to have a great impact on the organization's workflow, on academic researches, and can be beneficial for any kind of knowledge-based activity. Moreover, the development of such kind of technology is an important step in the development of AI and NLP.

Literature Survey

Natural Language Processing is an area of research within Artificial Intelligence that is concerned with giving computers the ability to understand natural language (spoken and written) in the same way a human could [1]. Knowledge of computational linguistics (rule-based modelling of human language), statistics, machine learning and deep learning are used either individually or combined to achieve the aforementioned goal [2]. Few of the researched tasks of NLP are Automatic Summarization (Automatic summarization produces an understandable summary of a set of text and provides summaries or detailed information of text of a known type), Co-Reference Resolution (Co-reference resolution refers to a sentence or larger set of text that determines all words which refer to the same object), Discourse Analysis (Discourse analysis refers to the task of identifying the discourse structure of connected text i.e. the study of text in relation to social context)[3]. Many researchers worked on NLP, building tools and systems which makes NLP what it is today. Tools like Sentiment Analyser, Parts of Speech (POS) Taggers, Chunking, Named Entity Recognitions (NER), Emotion detection, Semantic Role Labelling have a huge contribution made to NLP.

A chatbot is an artificially intelligent creature which can converse with humans. This could be text-based, or a spoken conversation (in case of voice-based queries). Chat bots are basically used for information acquisition. It can run on the local PCs and mobile phones, though most of the time it is accessed through the internet. It is a conversational agent which interacts with users in a certain domain or on a particular topic with input in natural language sentences. Mainly a chatbot works by a user asking some question or initiating a new topic of discussion.[5] a paper by Tarun Lalwani, Shashank Bhalotia conveyed Chat bots can be referred as software agents that pretend as human entity. These are the agents with AI embedded and using NLP they can answer to user questions. Predefined knowledge base helps develop a response to the query [6].

Chatbot design [7] typically relies on parsing techniques, pattern matching strategies and Natural Language Processing (NLP) to represent the chatbot knowledge. The latter is the dominant technique thanks to the popularization of libraries and cloud-based services such as DialogFlow [8] or IBM Watson Assistant [9], which rely on Machine 4 Learning (ML) techniques to understand the user input (based on a

set of training sentences provided as part of the chatbot definition) and provide user-friendly interfaces to design the conversational flow.

Document structure analysis can be regarded as a syntactic analysis problem. The order and containment relations among the physical or logical components of a document page can be described by an ordered tree structure and can be modelled by a tree grammar which describes the page at the component level in terms of regions or blocks.[10] Document structure analysis can be regarded as a syntactic analysis problem. The order and containment relations among the physical or logical components of a document page can be described by an ordered tree structure and can be modelled by a tree grammar which describes the page at the component level in terms of regions or blocks.

[11]. The global chatbot market is projected to reach 2 billion dollars by 2024, growing at a CAGR (compound annual growth rate) of 29.7%.2 This widespread interest and demand for chatbot applications has emphasized the need to be able to quickly build complex chatbot applications supporting AI-based natural language processing developed to minimize these efforts of managing playlists. However, the proposed existing algorithms in use are computationally slow and less accurate. This proposed system based on facial expression extracted will generate a playlist automatically thereby reducing the effort and time involved in rendering the process manually. Facial expressions are given using inbuilt camera. The image is captured using camera and that image is passed under different stages to detect the mood or emotion of the user. We will study about how to automatically detect the mood of the user and present him a playlist of songs which is suitable for his current mood. Proposed paper has used Viola-Jones algorithm and multiclass SVM (Support Vector Machine) for face detection and emotion detection respectively.

Materials and Methods

- **Proposed System**

The FileFlow is a sophisticated conversational chatbot powered by artificial intelligence technology aimed at improving the search and use of information from various file formats such as DOC, PPT, Excel, PDF files, and websites. The tool is equipped with NLP technologies and user-friendly interface through which the user can upload files and get answers in context to his/her questions. This section provides information about the architecture and functionality of the tool and the components that make it work.

- **Benefits of Proposed System**

With the new system there will be a better and improved accessibility of information. The new system saves much time and effort to access and utilize information from a variety of documents. This is especially important in situations where making decisions within a short period based on difficult information is very important. The new system uses advanced NLP technologies and therefore is able to provide answers which are not only accurate but also relevant to the questions asked by users. The fact that the system can manage various kinds of document formats makes it useful in many different industries.

In addition to this, the new system is able to learn from user feedback and improve its performance.

- **Hardware Requirements**
  - Most common resources used by the computer to execute the application include hardware. Since the system proposed will be a web application, then not many resources will be required to run the application. The hardware specifications required include:
  - Any quad core processor with at least 2.0 GHz clock speed. This will be required to run the application and process the document.
  - Minimum 4GB of hard disk space for installing all the required python modules. This will be needed for running the application on the local machine to install Python and its related modules and resources of the application.
  - Minimum 4GB of RAM. This will be required to ensure that the application runs smoothly on the user's end without any delays.

- **Software Requirements**
  - Software Requirements are the basic requirements that have to be installed on the computer in order to make the application work properly. The software requirements for this project are:
  - Programming Languages: Python
  - Frameworks & Libraries: OpenAI, Gradio, LLaMA, Poetry, BeautifulSoup, python-docx, PyPDF2, NumPy, Pandas, Pillow, PySpark, python-pptx
- **Database Management: Docker**
  - Anaconda is an environment in which all the libraries required for machine learning are installed. It also offers the option of creating an isolated environment.

System Design

The design of the FileFlow is designed to ensure maximum efficiency in handling documents and searching for information using a multi-tiered architecture that combines several different technologies. This part describes the technical design of the sstem..

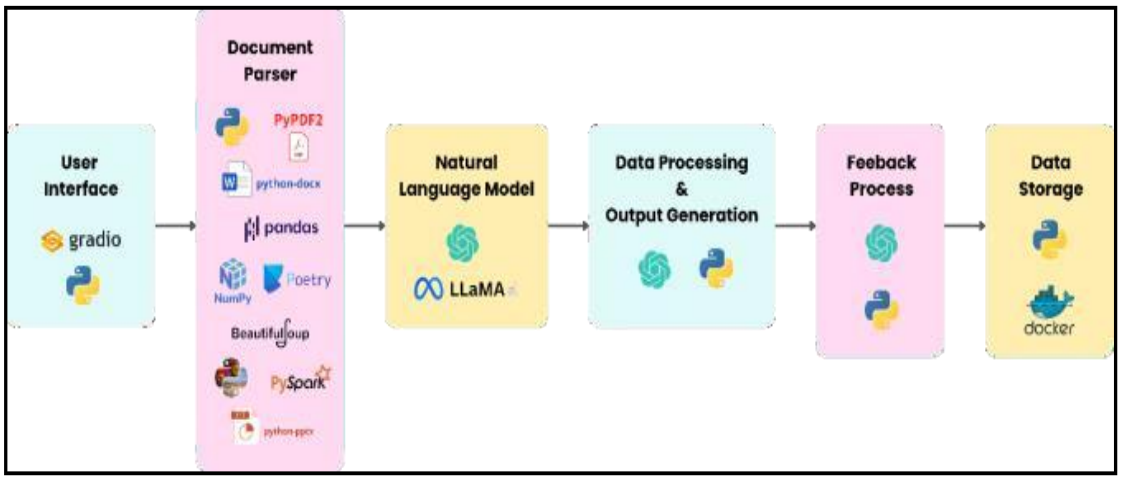


Figure 1: System Architecture of FileFlow

- **Document Input and Parsing Module**
  - Functionality: This module represents the first stop for all documents to be processed. The function of this module is to ingest documents submitted by users in different formats, for example, DOC, PPT, CSV, PDF, and web documents.
  - Process Flow: After uploading the document, it will be transformed into plain text, which is then passed to the pre-processing stage, where the process of noise removal, for instance, elimination of special symbols, is done.
  - Data Processing: After extracting text data, pre-processing is done by normalizing text (making it consistent case wise), removing unnecessary whitespaces and non-text content, for instance, images and charts, except when the images or charts are explicitly marked as being relevant to the context of text.
  - Customization and Enhancement: Custom parsers are used for those document types which have to be parsed differently from the regular narrative text, for instance, spreadsheet documents where formulas and cell references are involved.
- **Natural Language Processing Engine**
  - Functionality: The NLP Engine forms the crux of the system and uses the processed text for performing the semantic analysis of the text. Semantic Analysis of the text would help in identifying themes, contexts, and queries posed by the user.

- Process Flow: The processed text is divided into tokens like sentences and words. Each token is processed for its part of speech and then normalized for tenses, plurality etc. using lemmatization. Named entities such as dates, locations etc. are identified for tagging and quick access.
- Language Models: In addition to the normal NLP processing, customized language models are developed using a combination of supervised and unsupervised machine learning techniques.

- **Query Processing Unit**
  - Functionality: The user's natural language queries are managed by this module. This module not only identifies the user query intent but also retrieves the most relevant information from the processed documents.
  - Matching Data: Makes use of an advanced searching algorithm where both keyword and semantic searching capabilities are integrated to retrieve the most relevant document portions.
  - Optimization Methods: Implements the caching approach to improve performance.
- **Response Generation Module**
  - Functionality: This module provides the ultimate answers to the user queries. This module guarantees that not only the answers provided are correct but also well presented.
  - Process Flow: According to the information collected by the Query Processing Unit, the Response Generation Module develops an answer. If the query is straightforward, a template answer can be utilized. However, if the question is complicated, the system generates dynamic answers which include context details.
  - Feedback Loop: Includes a system of feedback in which user actions are recorded and analyzed to keep on improving the level of accuracy and relevance of responses. User corrections and suggestions are used to train the NLP and response models.
- **User Interface (UI)**
  - Functionality: It has been designed such that it is easy to use and intuitive for the users, thereby facilitating an easy uploading of documents, posing questions, and answering them.
  - Process Flow: The document is uploaded by the user through the UI. After the document has been processed and saved, the user is able to query the system. The responses are given in an easy-to-understand manner.
  - Security Measures: It implements very strong security procedures such that user data privacy and protection is guaranteed.

Methodology

- **Research Objectives and Literature Review**

First, in the early stages of our project, we formulated the research objectives, which are to develop an AI-based document interaction system. The research objectives represent a set of guiding principles for the research that is yet to be conducted. At the same time, in order to get more information regarding the field of conversational AI and document interaction, we performed a thorough literature review related to previous research in the area of conversational AI, NLP and document parsing.

Literature review provided us with a wealth of information and helped us realize where exactly lies the gap in the current research, as well as what can and should be done to advance the field further. By analyzing previous research on the topic, we managed to figure out what problems are not solved sufficiently well by currently available solutions. With the help of the information acquired during the literature review, we will try to contribute to advancing the field further.

- **System Development**

The requirement gathering phase of our project started with active user involvement to elicit and gather requirements for the intended system. Through interviews, surveys, and workshops, we tried to get information about the detailed requirements of the users when it comes to document management

and interaction. By creating a direct interaction with the users, we were able to collect information about the issues and pain points faced by them in the course of document interaction.

Following the requirement gathering process, we utilized the information collected in order to specify the use cases,

functionalities, and performance requirements of the system. This information was helpful in providing guidance for developing the intended system. In addition, we went ahead with the process of collecting different types of datasets in order to train the AI models being used in the system. This careful dataset collection process ensured that the trained AI models would be robust, accurate and capable of handling various kinds of document interaction tasks. modules. Implement error handling and logging mechanisms for system robustness.

- **Model Selection and System Design**

The task of evaluation and selection of appropriate pre-trained language and document parsing models became the key stage of our work. We carefully analyzed the effectiveness and suitability of different models to our project and chose those which will fit best to our requirements. Based on the information obtained during the requirement analysis stage, we determined the requirements of our system and selected the models according to them. Thus, we focused on models that proved their high performance in processing documents of various types and natural language queries.

After the stage of model selection, we moved to the design of the system's architecture and components. Considering the requirements of our project, we designed the architecture and defined workflows for document parsing, query processing, response generation, and feedback processing. We believe that by designing the architecture and workflow, we have provided a foundation for further development of our system and made sure that all the requirements specified in the project are met.

- **Implementation and Testing**

With the design process having been finalized, we moved onto the implementation process where the emphasis was on building and integrating the components of the system in order to form a coherent platform. Using our skills in programming languages and frameworks, we carefully developed the components of the system such that they would be compatible and synergistic. The components built were algorithms, APIs, and interfaces for the processes of document parsing, conversational AI, user interaction, and feedback management which were in line with the specifications of the project.

In addition, we carried out iterations in the development and testing phases in order to ensure the correctness and efficiency of the components of the system. Through the iterative method, we were able to identify possible issues that could arise in the development process. Tests for the components were performed through a variety of testing methods.

In this way, through an iterative process of developing and testing, we improved the system so that we ended up with a dependable and efficient system.

- **Testing, Evaluation, Documentation, and Reporting**

The evaluation and testing phase was an important part of the project, whereby we carried out intensive evaluations on the functionality, precision, and performance of the developed system. Using an extensive testing approach, we carried out unit testings, integration testings, and system testings to check the interactions and behavior of the components of the system. Evaluations were done based on simulated scenarios and actual scenarios and this helped in evaluating the performance of the system in the tasks of parsing documents, understanding the queries of the users, formulating responses, and receiving feedback from the users.

Moreover, we ensured that there was documentation and reporting of the entire process in order to document the methodology and the findings. We used research papers and presentations in documenting the entire process and the findings. Additionally, we provided documentation of the system that we have developed. This includes user manual, technical specifications, and developer guide among others. The idea of documentation of the entire process and findings of the project was intended to share the knowledge that we have gained in the course of the project.

Implementation and Result Analysis

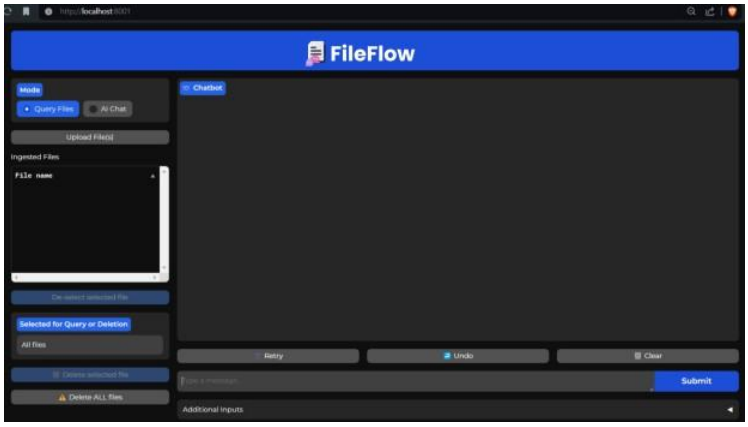


Figure 2: User Interface

The image shows the home screen for FileFlow web app.

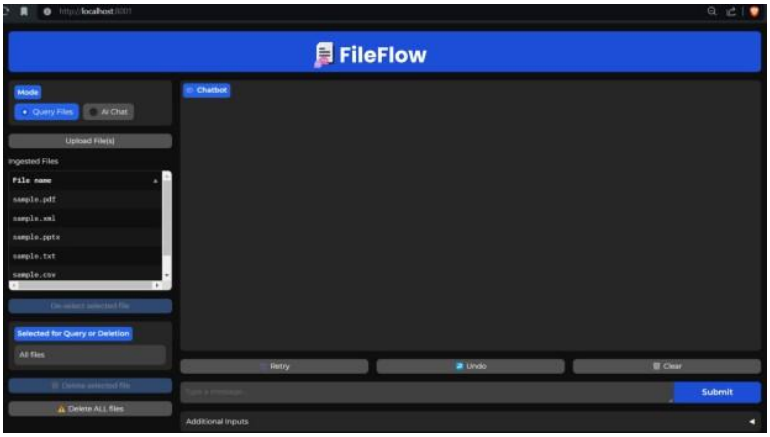


Figure 3: Screen when documents are uploaded

The image shows the screen of the web application when the user has uploaded the documents.

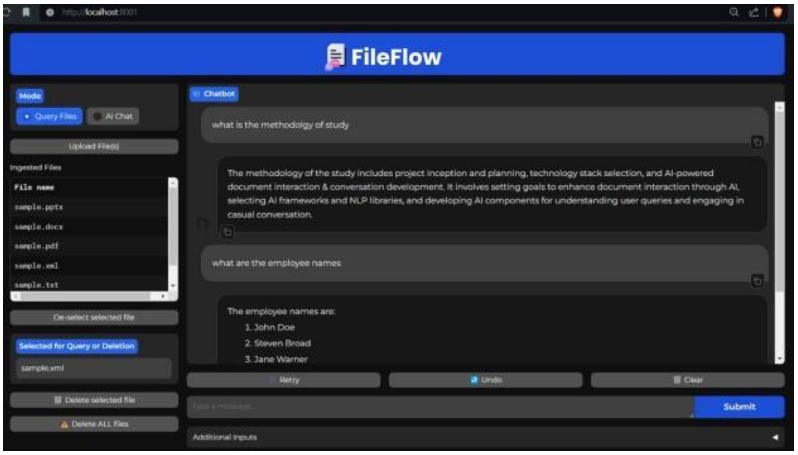


Figure 4: Screen while Querying Documents

The image shows the screen when the user performs desired queries on the document.

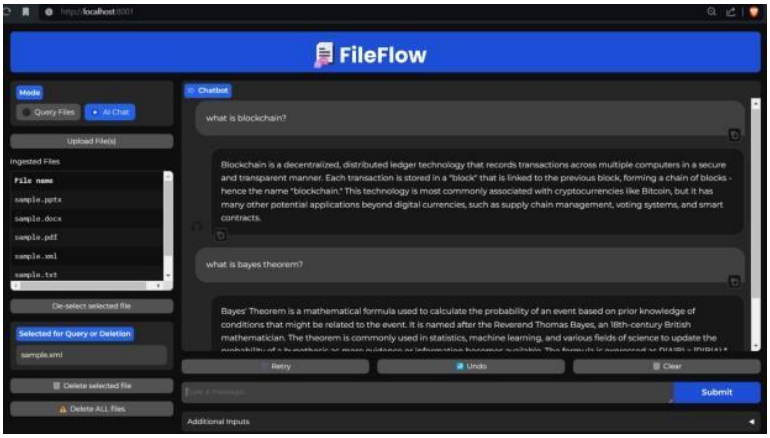


Figure 5: Screen while interacting with AI chat

The image shows the screen when the user interacts with the AI chat assistant about queries unrelated to the uploaded documents.

Accuracy of the Model

This section discusses the efficiency and accuracy of the AI models present in FileFlow by considering how effectively the system understands and responds to the inquiries made. This part is very important since the accuracy test plays a vital role in making sure that the chatbot is efficient.

Performance Metrics

- Precision: Determines the proportion of instances that are relevant out of all instances that were returned by the system. The higher the value of precision, the fewer false positives will there be.
- Recall: Evaluates the proportion of relevant instances out of the total number of relevant instances. The higher the value of recall, the fewer false negatives will there be.
- F1 Score: Harmonic mean of precision and recall metrics, giving the balance between these two metrics and showing their values through one figure. Useful in case of uneven distribution of classes labels.
- Accuracy: Proportion of correct instances out of the total number of analyzed cases.
-
- Response Time: Evaluates the time taken by the system to give an answer. This metric is important for assessing the effectiveness of user interaction with the system in real time.
- Testing Procedures:
- Dataset Preparation: The test dataset was curated in such a way that it consisted of various types and levels of complexity of documents to achieve a good coverage of all cases. The dataset contained documents from the training dataset as well as completely new documents not encountered by the model before.
- Query Generation: Various queries were designed to cover all facets and entities contained in the documents. These queries spanned from simple factual queries to more complex ones, needing semantic understanding.
- k-Fold Cross-Validation: Cross-validation was conducted using k-fold cross-validation technique.

Table I: Gpt-3.5-Turbo Accuracy

Combined Experiments Count	Sensitivity	Specificity
1	0.919	0.446
2	0.919	0.439
3	0.927	0.437

4	0.927	0.434
5	0.927	0.430
6	0.927	0.427
7	0.927	0.427
8	0.927	0.426
9	0.927	0.420
10	0.935	0.419

Scope of the Work

In future, we would further like to add:

- Predictive Query Suggestions: Generate predictive models of user queries based on the content of the document and the history of the user for proactive support.
- Powered Up Document Parsing: Consider implementing advanced methods of parsing and extraction of information from complicated document formats, such as rich in multimedia material, to increase the compatibility of the software and its ease of use.
- Multilingual Capabilities: Make the system capable of working in several languages at once to enable users of different linguistic origins to interact with the documents in the preferred language.
- Semantic Understanding: Implement semantic understanding capabilities in the system to
- understand the meaning of the document better and generate more advanced queries about the information in it.
- Multi-User Interaction: Integrate the possibility of interaction with the chatbot by several users at the same time.

Conclusion

It is noteworthy that the AI chatbot FileFlow can be viewed as a big step forward in the field of documents management and data extraction. The current research proved the effectiveness of using the technologies of natural language processing, along with a convenient interface, in order to enable users to effectively work with documents in different formats such as DOC, PPT, CSV, PDF, and web-pages.

Analysis and Discussion

- Strengths: The system had high accuracy in the retrieval and interpretation of structured information, and it was able to respond to various queries. Quick generation of responses is critical in keeping users engaged.
- Weaknesses: Some problems were experienced in answering complex questions that required the comprehension of subtle meanings or implications. Additionally, there were issues in processing some documents due to their lack of structure in high-accuracy content retrieval.
- Areas of Improvement: Improvement will be made by enhancing NLP systems to answer complex questions and increasing the size of the training set to process different documents. Techniques such as deep learning and context-based modelling are some of the methods that have been considered.

FileFlow is not just an application; it is a groundbreaking invention, which helps to meet a pressing need in modern society where information plays a major role. With further improvement and development, FileFlow has all the chances to become a new standard in AI-based interaction with documents, thus improving the decision-making processes and efficiency of its users. The successful implementation of FileFlow proves the effectiveness and innovative nature of artificial intelligence.

References

1. Tarun Lalwani, Shashank Bhalotia, Ashish Pal, Shreya Bisen, Vasundhara Rathod, IEEE International Conference on Computational Systems and Information Technology for Sustainable Solutions 2017.
2. Emanuela Haller, Traian Rebedea, “Designing a Chat-bot that Simulates an Historical Figure”, IEEE Conference Publications, July 2013.

3. Amey Tiwari, Rahul Talekar, Prof. S. M. Patil, “College Information Chatbot System”, International Journal of Engineering Research and General Science, Volume 2, Issue 2, April 2017.
4. C. M. Bishop. Pattern recognition and machine learning. Springer New York, 2006
5. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. Journal of Machine Learning Research, 3:993-1022, 2003.
6. Kreimeyer K., Foster M., Pandey A., Arya N., Halford G., Jones S.F., Forshee R., Walderhaug M., Botsis T. Natural language processing systems for capturing and standardizing unstructured clinical information: A systematic review. J. Biomed. Inform. 2017;73:14–29. doi: 10.1016/j.jbi.2017.07.012. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
7. Chowdhury G.G. Natural language processing. Annu. Rev. Inf. Sci. Technol. 2003;37:51–89. doi: 10.1002/aris.1440370103. [CrossRef] [Google Scholar]
8. Choudhary N (2021) LDC-IL: the Indian repository of resources for language technology. Lang Resources & Evaluation 55:855–867. [google scholar]
9. Bangalore S, Rambow O, Whittaker S (2000) Evaluation metrics for generation. In proceedings of the first international conference on natural language generation-volume 14 (pp. 1-8). Assoc Comput Linguist
10. Baud RH, Rassinoux AM, Scherrer JR (1992) Natural language processing and semantical representation of medical texts. Methods Inf Med 31(2):117–125
11. Bengio Y, Ducharme R, Vincent P (2001) A neural probabilistic language model. Proceedings of NIPS.Xu, B., Bu, J. J., Chen, C. & Cai, D. An Exploration of Improving Collaborative Recommender Systems via User-item Subgroups. in Proceedings of the 21st International Conference on World Wide Web: WWW '12, Lyon, France. New York: ACM Press. (2012 April).
12. Ramya Ramanathan, Radha Kumaran, Ram Rohan R, Rajat Gupta, and Vishalakshi Prabhu, 2nd IEEE International Conference on Computational Systems and Information Technology for Sustainable Solutions 2017.
13. Analysis using Bezier Curve Fitting” 2013 16th International Conference on Network-Based Information Systems.
14. On Statistical Approach”, International Journal of Soft Computing and Engineering (IJSCE) Volume-2, May 2012.
15. https://www.sciencedirect.com/science/article/pii/S2666285X210007 28#bib0005
16. https://github.com/opencv/opencv/tree/master/data/haarcascades
17. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin Ma, Ghemawat S, Irving G, Isard M, et al. Tensorflow: a system for large-scale machine learning. In: 12th {USENIX} Symposium on operating systems design and implementation ({OSDI} 16), 2016; p. 265–283.

❑❑❑