ACCENDIA
ADVANCED CHATBOT FOR CUSTOMIZED ENGAGEMENT AND NAVIGATION IN DATABASE INTERACTION

Project Report submitted by

OMAR MAHMOOD
(4NM20AI031)

RAGHAV KAMATH
(4NM20AI035)

RIFAATH MOHAMED AMEEN
(4NM20AI042)

Under the Guidance of

Dr. SHARADA U. SHENOY
Prof. & Head, Department of Artificial Intelligence and Machine Learning

In partial fulfillment of the requirements for the award of

the Degree of

Bachelor of Engineering in

Artificial Intelligence and Machine Learning

from

Visvesvaraya Technological University, Belagavi

Department of Artificial Intelligence and Machine Learning
NMAM Institute of Technology, Nitte - 574110
(An Autonomous Institution affiliated to VTU, Belagavi)

APRIL 2024

NITTE
EDUCATION TRUST

N.M.A.M. INSTITUTE OF TECHNOLOGY
(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi)
Nitte – 574 110, Karnataka, India

DEPARTMENT OF
ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

CERTIFICATE

Certified that the project work entitled

"Advanced Chatbot for Customized Engagement & Navigation for Database Interaction"

is a bonafide work carried out by

OMAR MAHMOOD	*RAGHAV KAMATH*	*RIFAATH AMEEN*
(4NM20AI031)	*(4NM20AI035)*	*(4NM20AI042)*

in partial fulfilment of the requirements for the award of

Bachelor of Engineering Degree in Artificial Intelligence and Machine Learning

*prescribed by **Visvesvaraya Technological University, Belagavi***

*during the year **2023-2024.***

It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library.

The project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the Bachelor of Engineering Degree.

Principal
N.M.A.M. Institute of Technology
Nitte, Karkala - 574 110

Signature of the Guide **Signature of the HOD** **Signature of the Principal**

N.M.A.M. INSTITUTE OF TECHNOLOGY
NITTE
UDUPI
574 110

Semester End Viva Voce Examination

Name of the Examiners	Signature with Date
1. Dr. Sharada U. Shenoy	16/4/24
2. Ms. Swathi Pai M.	SwathiPai 16/4/24.

ACKNOWLEDGEMENT

We believe that our major\ project will be complete only after we thank the people who have contributed to make this major project successful.

First and foremost, our sincere thanks to our beloved principal, **Dr. Niranjan N. Chiplunkar** for giving us an opportunity to carry out our major project work at our college and providing us with all the needed facilities.

I acknowledge the support and valuable inputs given by**, Dr. Sharada U Shenoy** the Head of the Department, Artificial Intelligence and Machine Learning Engineering, NMAMIT, Nitte

We express our deep sense of gratitude and indebtedness to our guide **Dr. Sharada U Shenoy,** Professor & Head, Artificial Intelligence and Machine Learning Engineering, for her inspiring guidance, constant encouragement, support, and suggestions for improvement during the course for our major project.

We also thank all those who have supported us throughout the entire duration of our major project.

Finally, we thank the staff members of the Department of Artificial Intelligence and Machine Learning Engineering and all our friends for their honest opinions and suggestions throughout the course of our major project.

Omar Mahmood (4NM20AI031)

Raghav Kamath (4NM20AI035)

Rifaath Mohamed Ameen (4NM20AI042)

ABSTRACT

In the era of information abundance, effective knowledge retrieval is paramount for research teams and business organizations. ACCENDIA is a private question answering system meticulously crafted for small groups, offering an innovative solution powered by the OpenAI GPT-3.5T API & Gemini 1.5 Pro, and integrated with the FAISS (Facebook AI Similarity Search) database.

The core functionality of ACCENDIA lies in its ability to seamlessly interact with the GPT-3.5 and Gemini 1.5 Pro models, allowing users to pose inquiries and receive highly contextualized responses. The system leverages the immense language understanding capabilities of GPT-3.5 and Gemini 1.5 Pro to process and comprehend questions, providing users with nuanced and accurate answers.

A crucial component of ACCENDIA is the integration of the FAISS database, serving as a repository for meticulously organized and indexed information extracted from various files uploaded by the user. FAISS enables efficient similarity search and retrieval of relevant knowledge, enhancing the system's responsiveness and overall utility.

The user experience with ACCENDIA is designed to be intuitive, allowing research teams and business professionals to harness the power of natural language processing. The system is adept at handling a diverse range of queries

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

ACCENDIA (Advanced Chatbot for Customized Engagement and Navigation in Database Interaction) is a question-answering system leveraging GPT-3.5 turbo and Gemini 1.5 Pro, tailored for exclusive use by small groups like research teams or businesses. Operating atop the OpenAI GPT-3.5 turbo and Google Gemini 1.5 Pro API, ACCENDIA integrates a dedicated database to furnish information for query responses. Engineered with a staunch commitment to privacy, the system refrains from retaining personal user data, including inquiry logs, or browsing histories, ensuring a secure and confidential user experience

1.1 Problem Statement

We address the challenge of efficiently accessing and retrieving specific information from vast datasets in a secure and tailored manner. In today's information-rich landscape, the overwhelming volume of data poses a considerable obstacle for individuals and organizations seeking relevant, accurate, and timely insights. Navigating through extensive repositories, such as PDF documents or varied databases, to extract pertinent information swiftly and accurately remains a formidable challenge. ACCENDIA aims to bridge this gap by leveraging GPT-3.5 turbo, Gemini 1.5 Pro technology and an integrated database, providing a private question-answering system tailored for select user groups, like research teams or businesses. The central issue we seek to solve revolves around facilitating seamless access to precise information by harnessing natural language processing capabilities. By creating a system that swiftly interprets user queries and retrieves specific content from stored data sources, we aim to revolutionize how information is accessed, enabling users to efficiently acquire targeted insights without sifting through extensive and potentially irrelevant data. This project's primary focus is on enhancing information accessibility and retrieval, offering a valuable solution to the information overload predicament prevalent across diverse sectors and domains.

1.2 Objectives

Our primary objective was to develop a private question-answering system with a primary focus on accuracy and transparent sourcing of information. ACCENDIA aims to deliver responses with high precision by harnessing our dedicated database, ensuring that the system not only provides accurate answers but also cites the specific sources contributing to those responses. This emphasis on accuracy and transparency instils user confidence in the information retrieved, fostering trust in the system's capabilities.

In addition to that, we envision to design and implement a user-friendly interface, ensuring accessibility and ease of use for a targeted audience comprising small groups like research teams or specialized business units. ACCENDIA prioritizes simplicity and intuitive navigation, enabling seamless access to its advanced features while catering to the specific needs of its user base. By fostering an environment where users can interact effortlessly with the system, ACCENDIA strives to enhance user experience and maximize efficiency in information retrieval tasks.

As the data landscape continues to expand, businesses grapple with the increasing challenges of secure and scalable document management. ACCENDIA emerges as a compelling technological innovation poised to alleviate these challenges. By offering a robust, secure, and user-centric solution, ACCENDIA aims to assist businesses in effectively managing their data repositories. The system's ability to deliver precise, transparent, and easily accessible information positions it as a valuable tool in navigating the complexities of modern data management, thereby facilitating informed decision-making and operational efficiency within business environments.

1.3 Motivation

The motivation driving ACCENDIA is deeply rooted in the recognition of the limitations posed by generalized information retrieval systems. While GPT models excel in comprehending vast volumes of general knowledge, their responses often remain rooted in generalities. We were inspired to create a solution that transcends these limitations by honing in on a more specific and tailored approach. By integrating our own database into ACCENDIA, we aimed to amplify the precision and relevance of information retrieval.

CHAPTER 2

LITERATURE REVIEW

2.1 Existing System

Paper 1: A Review on Implementation Issues of Rule-based Chatbot Systems:

The paper introduces an exploration into the field of Chatbot systems conducted by Sandeep Thorat et al. [1]. Within this realm, the authors illuminate the prevalent challenges encountered by artificial intelligence methods when addressing user queries adequately. They underscore the significance of rule-based Chatbot systems within industries due to perceived limitations of existing AI approaches. Through a comparative analysis, the study delves into two prominent rule-based Chatbot implementation frameworks, namely Google Dialogflow and IBM Watson, elucidating their respective features and functionalities. Moreover, the paper emphasizes the importance of evaluating Chatbot system performance using defined metrics to enhance effectiveness and efficiency. Central to their discourse is the examination of natural language understanding and how frameworks such as Google Dialogflow and IBM Watson approach this pivotal aspect. Additionally, the authors articulate expectations for future Chatbot systems, envisioning advancements in performance, user experience, and response accuracy to effectively meet evolving demands.

Paper 2: Creating Large Language Model Applications Utilizing LangChain:

The paper introduces the burgeoning field of Large Language Models (LLMs) through the work of Oguzhan Topsakal et al. [2]. Within this domain, the authors shed light on Lang Chain, an open-source software library designed to streamline application development. Lang Chain is renowned for its seamless integration with various data sources and applications, making it a cornerstone in the AI community. The study meticulously examines Lang Chain's fundamental attributes, with a particular focus on its modular abstractions, termed components, and its customizable pipelines, known as chains. Emphasizing Lang Chain's pivotal role in accelerating LLM application development and facilitating interaction with diverse data sources and applications, the paper highlights its modular architecture, which allows for tailored pipelines catering to specific use cases. This modular approach significantly expedites the development lifecycle for LLM applications. As a significant contribution to the discourse surrounding LLM application development, the paper advocates for further exploration and utilization of Lang Chain and similar tools. It envisions a future where streamlined and efficient LLM-driven applications are commonplace, fuelled by the continued advancement and adoption of tools like Lang Chain.

Paper 3: DocChat: An Information Retrieval Approach for Chatbot Engines:

The paper authored by Zhao Yan et al. [3] introduces DocChat, a novel approach to information retrieval for chatbot engines that diverges from traditional question-response pairs by harnessing unstructured documents. DocChat leverages a learning-to-rank model equipped with finely-tailored features at various levels of granularity to assess the relevance between user utterances and potential responses. Through evaluations conducted in both English and Chinese contexts, the approach showcases reasonable enhancements and adaptability in English, while also effectively complementing existing chatbot engines in the Chinese language domain. The authors provide a comparative analysis of DocChat with XiaoIce, a prominent chitchat engine in China, demonstrating DocChat's effectiveness as a supplementary tool for chatbot engines primarily reliant on question-response pairs. The paper not only highlights the potential of utilizing unstructured documents for enhancing chatbot capabilities but also delineates avenues for future improvements, particularly focusing on refining the triggering component and bolstering the handling of multi-round conversations. This research underscores the importance of exploring alternative sources of information retrieval to advance chatbot capabilities and elevate user interactions to new levels of efficiency and satisfaction.

Paper 4: Enhancing PDF Interaction for a More Engaging User Experience:

The paper authored by Subhajit Panda [4] delves into the shortcomings of conventional PDF readers within library systems, which often lack sufficient interaction capabilities, resulting in user dissatisfaction and disengagement. In response to this challenge, the paper introduces ChatPDF, an innovative online software platform that leverages the ChatGPT API to provide a more intuitive and natural means of engaging with PDF documents. ChatPDF distinguishes itself by offering unique functionalities such as summarization, recommendations, multi-lingual support, and AI assistance, thereby making a substantial contribution to the field of library science. The paper advocates for the adoption of ChatPDF in libraries to enhance user engagement and satisfaction with digital resources. Furthermore, it identifies potential areas within library systems where ChatPDF could be implemented and calls for further research to evaluate its impact on user experience and library operations. This work underscores the importance of bridging the divide between traditional document formats and modern interactive technologies to enhance accessibility and usability within library settings.

2.2 Observations from Literature Review

Sandeep Thorat et al. [1] delve into the realm of Chatbot systems, shedding light on the prevalent challenges faced by artificial intelligence methods in adequately addressing user queries. They emphasize the prominence of rule-based Chatbot systems within industries owing to the perceived inadequacies of existing AI approaches. Through a comparative analysis, they scrutinize two prominent rule-based Chatbot implementation frameworks, Google Dialogflow, and IBM Watson, elucidating their respective features and functionalities. Moreover, the paper underscores the significance of evaluating Chatbot system performance through defined metrics, advocating for improved effectiveness and efficiency. A focal point of their discussion revolves around natural language understanding and how frameworks like Google Dialogflow and IBM Watson tackle this crucial aspect. Additionally, the authors articulate expectations for future Chatbot systems, envisioning advancements in performance, user experience, and response accuracy to meet evolving demands effectively.

Oguzhan Topsakal et al. [2] illuminates the burgeoning realm of Large Language Models (LLMs), spotlighting Lang Chain, an open-source software library engineered to accelerate application development within this domain. Recognized for its seamless interaction with diverse data sources and applications, Lang Chain stands as a prominent fixture in the AI community. The study meticulously dissects Lang Chain's core attributes, emphasizing its modular abstractions—termed components—and customizable, use-case-specific pipelines, referred to as chains. Notably, Lang Chain's pivotal role in streamlining LLM application development and facilitating effortless interaction with varied data sources and applications is underscored. Its modular architecture, enabling tailored pipelines for distinct use cases, serves as a catalyst in expediting the development lifecycle for LLM applications. As a valuable contribution to the discourse surrounding LLM application development, the paper advocates for deeper exploration and utilization of Lang Chain and analogous tools, envisioning a future of streamlined and efficient LLM-driven applications.

Zhao Yan et al. [3] introduced DocChat, an innovative information retrieval approach for chatbot engines that deviates from conventional question-response pairs by utilizing unstructured documents. DocChat employs a learning to rank model equipped with features tailored at different granularities to gauge the relevance between user utterances and potential responses. Through evaluations conducted in both English and Chinese contexts, the approach demonstrates reasonable improvements and adaptability in English, while also effectively complementing existing chatbot engines in Chinese. The authors compare DocChat with XiaoIce, a prominent chitchat engine in China, illustrating DocChat's efficacy as a supplementary tool for chatbot engines relying primarily on Q-R pairs. Highlighting the potential of leveraging unstructured documents for chatbot engines, the paper also identifies avenues for future enhancements, particularly focusing on refining the triggering component and enhancing the handling of multi-round conversations. This work underscores the significance of exploring alternative sources of information retrieval for advancing chatbot capabilities and improving user interactions.

Subhajit Panda [4] addresses the limitations of traditional PDF readers in library systems, which often fail to provide adequate interaction capabilities, leading to user frustration and disengagement. To tackle this issue, the paper introduces ChatPDF, an innovative online software platform that harnesses the ChatGPT API to offer a more intuitive and natural way of interacting with PDF documents. ChatPDF stands out by offering unique features such as summarization, recommendations, multi-lingual support, and AI assistance, thereby making a significant contribution to the field of library science. The paper advocates for libraries to consider implementing ChatPDF to enhance user engagement and satisfaction with their digital resources. Furthermore, it identifies potential areas for ChatPDF implementation within library systems and calls for further research to evaluate its impact on user experience and library operations. This work underscores the importance of bridging the gap between traditional document formats and modern interactive technologies to improve accessibility and usability in library settings.

CHAPTER 3

PROPOSED METHODOLOGY & ARCHITECTURAL DESIGN

3.1 User Needs

In outlining the proposed methodology and architectural design for ACCENDIA, it is essential to consider the primary user needs that guide the system's development. Firstly, the methodology should address the seamless integration of the OpenAI GPT-3.5 and Google Gemini 1.5 Pro API, emphasizing the user's ability to pose natural language queries and receive accurate, contextually relevant responses. This involves a meticulous exploration of GPT-3.5 and Gemini 1.5 Pro's capabilities, ensuring that the system effectively harnesses its language understanding and generation prowess.

Furthermore, the architectural design should prioritize the incorporation of the FAISS database for efficient information retrieval. Users require a system that not only leverages GPT-3.5 and Gemini 1.5 Pro's language processing but also integrates seamlessly with FAISS to store and organize extracted knowledge. The architecture should facilitate quick and accurate similarity searches, enhancing the responsiveness of the system and providing users with a robust mechanism for retrieving pertinent information from the stored database.

A key consideration in both the methodology and architectural design is user adaptability. ACCENDIA should be designed to accommodate the unique needs of small research teams or business organizations, ensuring a user-friendly interface and intuitive interactions. The system should evolve and learn from user engagements, continuously improving its understanding and responsiveness to diverse queries over time.

Lastly, a comprehensive security framework must be an integral part of the methodology and architectural design. Users demand a system that safeguards sensitive information, necessitating the implementation of encryption protocols, secure data transmission measures, and regular security updates. Addressing these user needs in the proposed methodology and architectural design ensures that ACCENDIA not only meets functional requirements but also aligns with the practical expectations and preferences of its intended user base.

3.2 Methodology

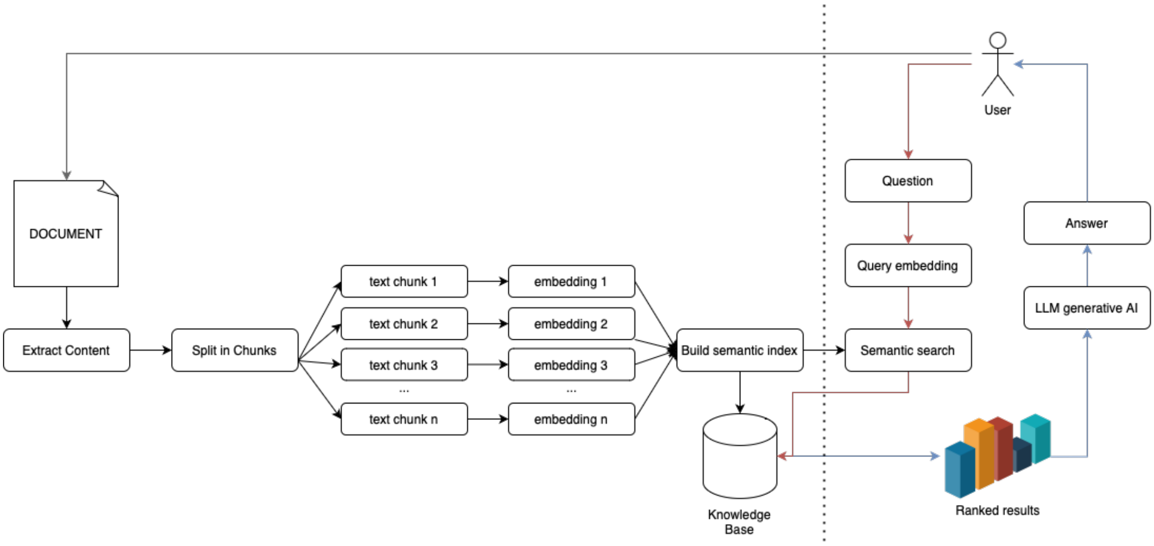


Figure 3.1 Proposed system architecture

The methodology proposed within ACCENDIA revolves around a multi-step process tailored to handle diverse document types and facilitate user queries seamlessly. At its core, the methodology integrates several key components, ensuring effective document processing and responsive query handling.

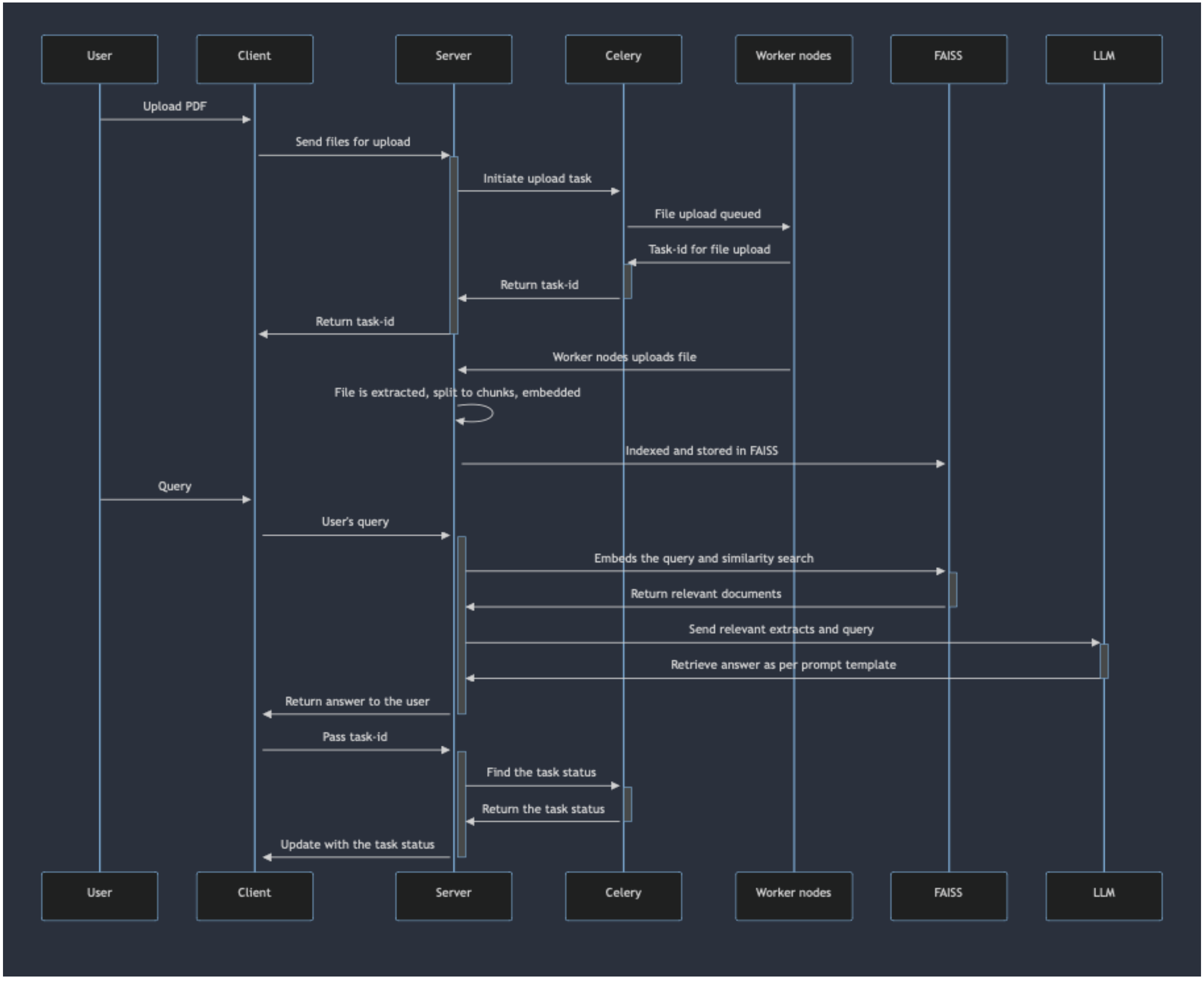
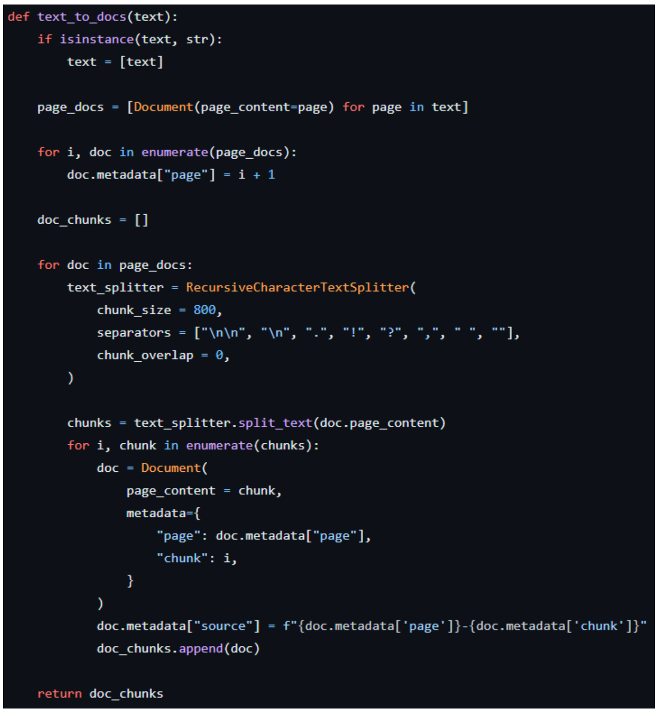

User
Client
Server
Celery
Worker nodes
FAISS
LLM
Upload PDF
Send files for upload
Initiate upload task
File upload queued
Task-id for file upload
Return task-id
Return task-id
Worker nodes uploads file
File is extracted, split to chunks, embedded
Indexed and stored in FAISS
Query
User's query
Embeds the query and similarity search
Return relevant documents
Send relevant extracts and query
Retrieve answer as per prompt template
Return answer to the user
Pass task-id
Find the task status
Return the task status
Update with the task status
User
Client
Server
Celery
Worker nodes
FAISS
LLM

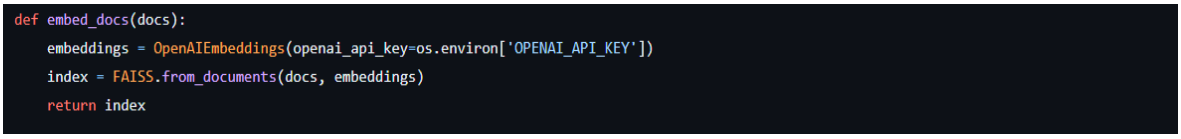
Figure 3.2: Sequence Diagram of Proposed System



```python
def text_to_docs(text):
    if isinstance(text, str):
        text = [text]

    page_docs = [Document(page_content=page) for page in text]

    for i, doc in enumerate(page_docs):
        doc.metadata["page"] = i + 1

    doc_chunks = []

    for doc in page_docs:
        text_splitter = RecursiveCharacterTextSplitter(
            chunk_size = 800,
            separators = ["\n\n", "\n", ".", "!", "?", ",", " ", ""],
            chunk_overlap = 0,
        )

        chunks = text_splitter.split_text(doc.page_content)
        for i, chunk in enumerate(chunks):
            doc = Document(
                page_content = chunk,
                metadata={
                    "page": doc.metadata["page"],
                    "chunk": i,
                }
            )
            doc.metadata["source"] = f"{doc.metadata['page']}-{doc.metadata['chunk']}"
            doc_chunks.append(doc)

    return doc_chunks
```

Figure 3.3: Code snippet of text splitter

Initially, ACCENDIA employs a sophisticated document detection mechanism, assuming the accurate identification of file formats. This step is pivotal as it triggers distinct processing paths, leveraging specialized methods for PDF documents. Subsequently, the system applies tailored content extraction methodologies, utilizing modules like UnstructuredFileLoader from the Lang chain library. These modules play a pivotal role in extracting text content from different document formats, laying the foundation for subsequent processing.



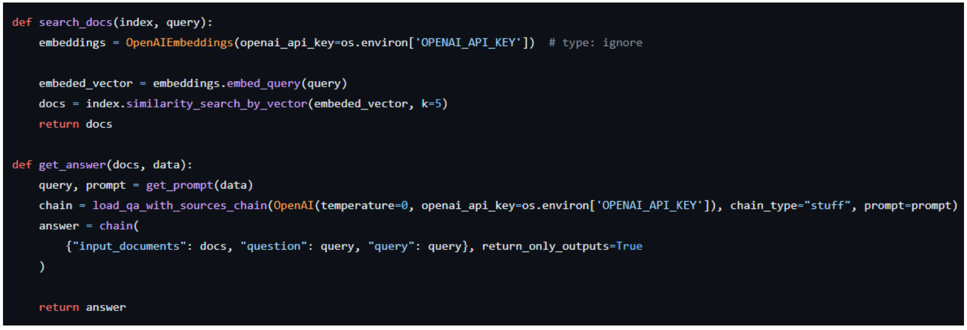
```
def embed_docs(docs):
    embeddings = OpenAIEmbeddings(openai_api_key=os.environ['OPENAI_API_KEY'])
    index = FAISS.from_documents(docs, embeddings)
    return index
```

Figure 3.4: Code snippet of embeddings and vector store

Content segmentation emerges as a crucial phase in ACCENDIA's methodology, assuming the strategic division of document content into manageable chunks. Leveraging the CharacterTextSplitter strategy, the system fragments text into coherent sections based on predefined criteria, allowing for more efficient handling and semantic relevance. The methodology inherently assumes the importance of context preservation between these chunks, achieved through carefully calibrated overlap parameters.

Additionally, ACCENDIA's methodology incorporates embedding generation, where each chunk undergoes transformation into meaningful embeddings using OpenAI's language models. This step assumes the accurate capture of semantic nuances and contextual relevance within these embeddings, critical for subsequent query responses.

Further, the system's functionality relies on a robust document search mechanism facilitated by FAISS vector stores. This dependency assumes the accurate representation and indexing of document embeddings, enabling swift and accurate retrieval of relevant chunks in response to user queries.



```python
def search_docs(index, query):
    embeddings = OpenAIEmbeddings(openai_api_key=os.environ['OPENAI_API_KEY'])  # type: ignore

    embeded_vector = embeddings.embed_query(query)
    docs = index.similarity_search_by_vector(embeded_vector, k=5)
    return docs

def get_answer(docs, data):
    query, prompt = get_prompt(data)
    chain = load_qa_with_sources_chain(OpenAI(temperature=0, openai_api_key=os.environ['OPENAI_API_KEY']), chain_type="stuff", prompt=prompt)
    answer = chain(
        {"input_documents": docs, "question": query, "query": query}, return_only_outputs=True
    )

    return answer
```

Figure 3.5: Code snippet of similarity search

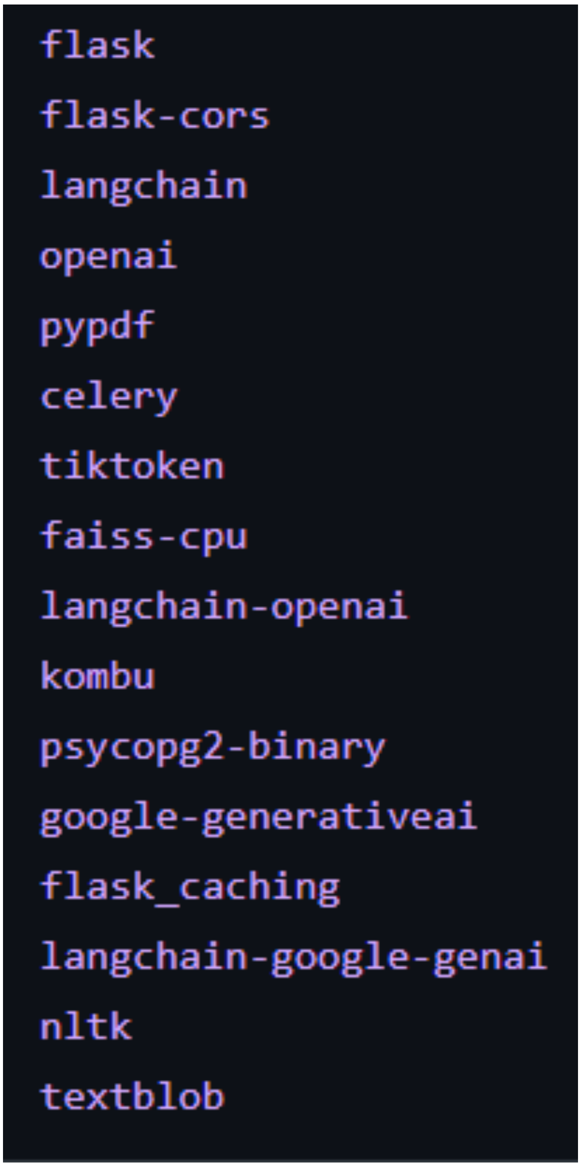
In essence, the methodology woven into ACCENDIA's framework assumes a sequential and nuanced approach, encompassing document detection, content extraction, segmentation, embedding generation, and efficient retrieval. The success of each step is pivotal, as they collectively drive the system's ability to handle diverse document types, provide accurate representations, and deliver pertinent responses to user queries. Adjustments or refinements in these stages can significantly impact ACCENDIA's efficacy and performance.

3.3 Assumptions and Dependencies

ACCENDIA, with its complex development landscape, is underpinned by a set of fundamental assumptions that inform its intricate design and functionality. Chief among these is the premise that the OpenAI GPT-3.5 and Google Gemini 1.5 Pro models possess the expected language understanding and generation capabilities. The system depends on the model's capacity to comprehend context and meaning, ensuring precise interpretation and generation of responses to user queries. Furthermore, ACCENDIA assumes the integrity of the information stored in the FAISS database. It assumes that the data gleaned from various files and carefully indexed in FAISS is not only indicative but also current and highly relevant to the diverse range of user queries it seeks to address. Lastly, the system assumes a stable and dependable internet connection, recognizing that its responsiveness and real-time interaction hinge on the reliability of its connectivity.

Complementing these assumptions are critical dependencies integral to ACCENDIA's overall performance. The system is intricately tied to the OpenAI GPT-3.5 and Google Gemini 1.5 Pro API, hinging on its availability and seamless operation for the accurate generation of responses. The integration and effective functioning of the FAISS database are equally pivotal dependencies, demanding a meticulous configuration to ensure optimal information retrieval capabilities. The proper alignment of FAISS, with its indexing mechanisms calibrated for precision, becomes paramount for ACCENDIA to fulfill its knowledge retrieval objectives. Furthermore, the system's reliance on a stable and well-configured Python environment introduces a dependency on the correct runtime conditions, with potential disruptions in the Python environment threatening the overall functionality of ACCENDIA. The security measures implemented, encompassing encryption protocols and data protection mechanisms, depend on flawless execution; any lapses or vulnerabilities in these frameworks could compromise the confidentiality of user data. Lastly, ACCENDIA's adaptability and continuous improvement through user interactions assume a consistent level of user engagement. The system relies on users actively participating in its learning processes to refine and enhance its language understanding capabilities over time.

Navigating the intricacies of these assumptions and dependencies is pivotal in the ongoing refinement and optimization of ACCENDIA, ensuring its resilience and effectiveness in meeting the dynamic needs of its user base. Addressing these elements not only manages expectations but also guides the system's evolution and future developments.



```
flask
flask-cors
langchain
openai
pypdf
celery
tiktoken
faiss-cpu
langchain-openai
kombu
psycopg2-binary
google-generativeai
flask_caching
langchain-google-genai
nltk
textblob
```

Figure 3.6: Project Dependencies

CHAPTER 4

SYSTEM REQUIREMENTS & SPECIFICATIONS

4.1 Introduction

ACCENDIA, as a comprehensive question answering system, demands specific system requirements to ensure optimal performance and user satisfaction.

To begin with, the hardware specifications should align with contemporary computing standards. While the system can function on standard desktops and laptops, it is recommended to utilize modern hardware with multicore processors to enhance computational efficiency. A minimum of 8GB RAM is advised to facilitate smooth interactions with the GPT-3.5 and Gemini 1.5 Pro model and seamless management of the FAISS database. Furthermore, an ample storage solution is essential to accommodate the FAISS database and associated files efficiently.

In terms of the software environment, ACCENDIA caters to a diverse user base by offering compatibility with major operating systems, including Windows, macOS, and Linux. The system operates within a Python environment, necessitating Python version 3.7 or above. Dependencies and libraries crucial for GPT-3.5, Gemini 1.5 Pro integration and FAISS database management are seamlessly integrated into the system's setup, ensuring a straightforward user experience.

A stable internet connection is a pivotal requirement for ACCENDIA, as it relies on real-time communication with the OpenAI GPT-3.5 and Google Gemini 1.5 Pro API. A high-speed and reliable internet connection not only ensures low latency but also enhances the overall responsiveness of the system, enabling users to experience seamless interactions when posing queries to the question answering system.

Lastly, ACCENDIA is designed with scalability in mind. While the system operates efficiently with the minimum requirements, organizations with expanding datasets or higher user loads can seamlessly scale the system to meet their evolving needs. The FAISS database, a core component of ACCENDIA, efficiently scales to accommodate the growing knowledge repositories of users, providing flexibility and adaptability to organizational requirements.

4.2. Software requirements

- Windows 10 or above / Linux.
- Python 3.7 or above
- OpenAI GPT-3.5 API access
- Google Gemini 1.5 Pro API access
- FAISS library
- LangChain Framework
- Flask
- PostgreSQL
- RabbitMQ
- Internet browser

4.3. Hardware requirements

- Multicore processor
- Minimum 8GB RAM
- Adequate storage space
- Stable internet connection

4.4. Non-functional requirements

- Robust security measures
- Scalable design
- Intuitive and user-friendly interface
- High reliability

CHAPTER 5

IMPLEMENTATION

5.1 Backend

The backend implementation consists of a Flask-based REST API, integrated with asynchronous task processing using Celery.

Here's a breakdown of the components:

5.1.1. Flask Setup:

- Flask application initialized with CORS enabled to handle cross-origin requests.
- Configuration for Celery and caching set up.
- Flask app configured to run on host '0.0.0.0' and port 8000.

5.1.2. API Endpoints:

- **/api/upload/<model>/<project_id>:** Handles file uploads and extraction of content from PDF files. Asynchronously processes uploads and extraction tasks.
- **/api/<model>/<projectID>/query**: Handles user queries by searching for relevant documents and generating responses. Asynchronously handles query tasks.
- **/api/tasks/<task_id>**: Retrieves the status of asynchronous tasks by task ID.

5.1.3. Celery Tasks:

- **upload_pdf**: Processes uploaded PDF files, extracts content, and saves to FAISS database.
- **extract_content**: Extracts content from PDF files and saves to FAISS database.
- **handle_query**: Handles user queries by searching for relevant documents and generating responses using OpenAI or Gemini model.
- **get_status**: Retrieves the status of asynchronous tasks.

5.1.4. Helper Functions:

- **parse_pdf**: Parses PDF files and extracts text content.
- **text_to_docs**: Converts text content into document chunks.
- **embed_docs**: Embeds document chunks using OpenAIEmbeddings.
- **search_docs**: Searches for relevant documents based on user queries.
- **get_answer**: Generates answers to user queries based on relevant documents.
- **get_answer_sub**: Generates answers to user queries using the Gemini model with chat history.

5.1.5. Prompts:

- A template for generating prompts for user queries is defined in the prompt.py file.
- The template includes placeholders for question, summaries, and final answer, filled dynamically based on user queries.

5.2 Frontend

The frontend component of ACCENDIA is designed solely for demonstration purposes to illustrate how the system's REST API can be utilized by end-users. It comprises a simple web interface that allows users to interact with the system by submitting natural language queries and receiving responses. The frontend is implemented using HTML, CSS, and JavaScript, with minimal styling and functionality to focus on the demonstration aspect.

5.2.1. Functionality:

- **download_pdf**: Downloads a PDF file from a given URL.
- **question_answer**: Handles user input by processing PDF uploads or URL downloads and extracting content for question answering.
- **get_query**: Generates answers to user questions based on provided chat history, question, and selected model.

5.2.2. UI Components:

- Chatbot Interface: Displays chat history and allows users to enter questions.
- Input Fields:
- OpenAI API Key: Allows users to input their OpenAI API key for authorization.
- Project ID: Allows users to specify a project ID or name for backend processing.
- PDF URL: Enables users to provide a URL to download PDF files.
- File Upload: Allows users to upload PDF files directly.
- Question Input: Provides a text box for users to enter custom questions or select from predefined questions.
- Model Selection: Radio buttons to choose between GPT and Gemini models.

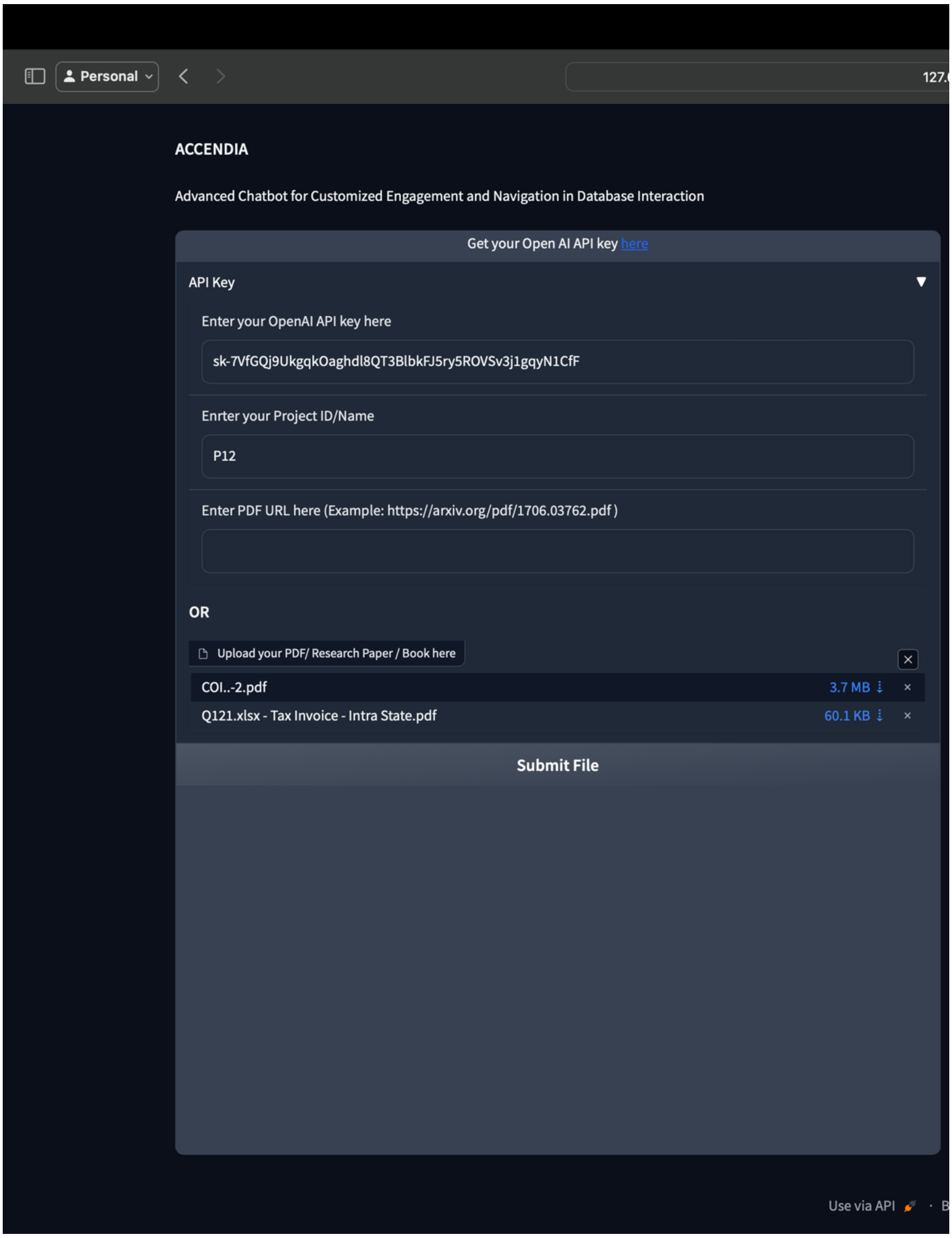
5.2.3. Predefined Questions:

- A list of predefined questions is provided for user convenience. Users can click on a question to autofill the input box.

5.2.4. Interaction:

- Users can submit their questions or file uploads by clicking on the corresponding buttons.

- Responses are displayed in the chat history section, providing an interactive conversational experience.



ACCENDIA

Advanced Chatbot for Customized Engagement and Navigation in Database Interaction

Get your Open AI API key here

API Key ▼

Enter your OpenAI API key here

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

Enrter your Project ID/Name

P12

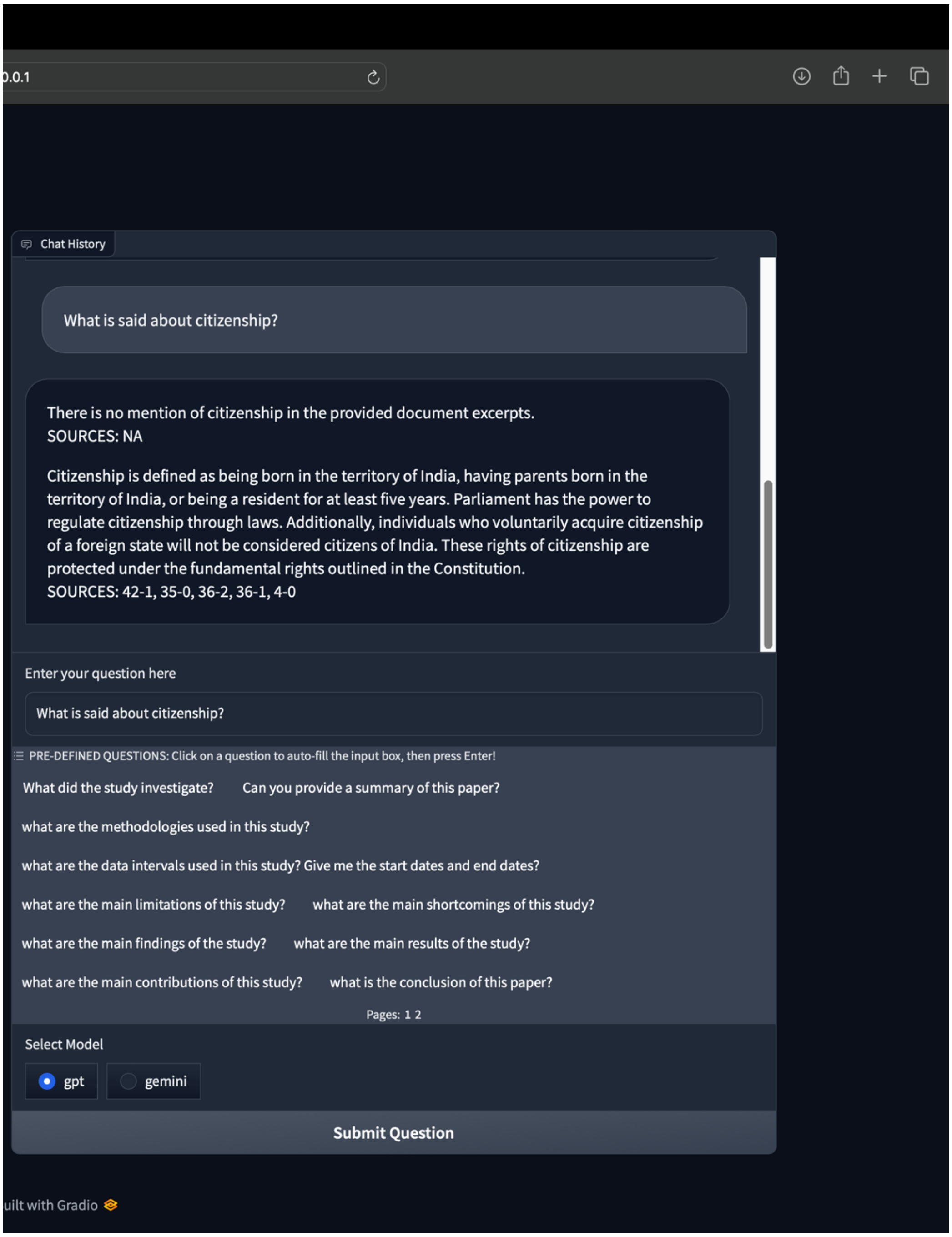
Enter PDF URL here (Example: https://arxiv.org/pdf/1706.03762.pdf)

OR

Upload your PDF/ Research Paper / Book here

COI..-2.pdf	3.7 MB	×
Q121.xlsx - Tax Invoice - Intra State.pdf	60.1 KB	×

Figure 5.1: Snippet of the demonstrative UI (Upload file)



What is said about citizenship?

There is no mention of citizenship in the provided document excerpts.
SOURCES: NA

Citizenship is defined as being born in the territory of India, having parents born in the territory of India, or being a resident for at least five years. Parliament has the power to regulate citizenship through laws. Additionally, individuals who voluntarily acquire citizenship of a foreign state will not be considered citizens of India. These rights of citizenship are protected under the fundamental rights outlined in the Constitution.
SOURCES: 42-1, 35-0, 36-2, 36-1, 4-0

Enter your question here: What is said about citizenship?

PRE-DEFINED QUESTIONS: Click on a question to auto-fill the input box, then press Enter!

What did the study investigate?
Can you provide a summary of this paper?
what are the methodologies used in this study?
what are the data intervals used in this study? Give me the start dates and end dates?
what are the main limitations of this study?
what are the main shortcomings of this study?
what are the main findings of the study?
what are the main results of the study?
what are the main contributions of this study?
what is the conclusion of this paper?

Select Model: ● gpt ○ gemini

Figure 5.2: Snippet of the demonstrative UI (Query & Chat History)

CHAPTER 6

RESULTS AND DISCUSSIONS

6.1 Introduction

The "Results and Discussions" section examines the performance of ACCENDIA, a private question answering system, against human-driven knowledge retrieval. Using a single control participant, the study evaluates the system's effectiveness in processing inquiries and providing accurate responses, considering the average words per minute (WPM) of human reading compared to the system's processing time. Through an analysis of response accuracy, response time, and user satisfaction, this section explores ACCENDIA's capabilities and identifies areas for enhancement.

6.2 Experimental Setup

6.2.1 Human vs Proposed System:

The experimental setup for evaluating ACCENDIA involved a single control participant interacting with both the system and human-driven knowledge retrieval. The participant was presented with a series of inquiries and tasked with finding answers using both methods: first, by engaging with the proposed ACCENDIA system, and second, by utilizing their own cognitive abilities to retrieve information from two distinct PDF documents. The "COI" document contained the Constitution of India, while the "Quotation" document consisted of a sample quotation addressed to NMAM Institute of Technology. These documents were deliberately chosen to be unrelated, representing diverse subject matters and testing ACCENDIA's adaptability to different types of content.

During the experiment, the control participant's interactions with both the system and human-driven retrieval were meticulously observed and recorded. This setup facilitated a direct comparison between ACCENDIA's knowledge retrieval capabilities and the participant's individual cognitive processes when accessing unrelated information from the Constitution of India and a sample quotation.

By analysing factors such as response accuracy, response time, and user satisfaction for both approaches, the experimental setup aimed to provide valuable insights into ACCENDIA's performance in handling diverse document types and queries. The findings from this study served as a foundational basis for understanding ACCENDIA's effectiveness and its potential to excel in various knowledge retrieval tasks across different domains.

6.2.2 GPT vs Gemini

A Control document was chosen to perform comparative performance evaluation between GPT 3.5 Turbo and Gemini 1.5 Pro models. The following rubrics were considered for assigning scores out of 10:

1. **Accuracy:** The extent to which the information provided in the response aligns with the content of the document or the expected answer to the question.
2. **Completeness:** The degree to which the response covers all relevant aspects of the topic or addresses all components of the question.
3. **Relevance:** The relevance of the information provided in the response to the question asked or the context of the document.
4. **Clarity:** The clarity and coherence of the language used in the response, considering factors such as readability, organization, and coherence.
5. **Source Attribution:** The accuracy and appropriateness of the sources cited in the response, indicating the credibility and reliability of the information provided.

Document Name: Citizenship of India
Document Page Count: 402
Control Question: "What is said about citizenship?"

Gemini Response:

"At the beginning of the Constitution, citizenship was granted to individuals domiciled in India who met specific criteria:

- Birth within India

- Parental lineage

- Residency

Provisions for individuals migrating to India from Pakistan and the potential loss of citizenship upon voluntarily acquiring citizenship of a foreign state are mentioned.

Parliament's authority to regulate citizenship rights through legislation is highlighted.

Sources cited: 35-0, 36-1, 36-2, 42-1"

Scores assigned for Gemini Response:

- Accuracy: 9
- Completeness: 8
- Relevance: 9
- Clarity: 8
- Source Attribution: 9
- Total Score: 43

GPT Response:

"Citizenship is defined as being born in the territory of India, having parents born in the territory of India, or being a resident for at least five years.

Parliament has the power to regulate citizenship through laws.

Additionally, individuals who voluntarily acquire citizenship of a foreign state will not be considered citizens of India.

These rights of citizenship are protected under the fundamental rights outlined in the Constitution.

Sources cited: 42-1, 35-0, 36-2, 36-1, 4-0"

Scores assigned for GPT Response:

- Accuracy: 8
- Completeness: 7
- Relevance: 8
- Clarity: 8

- Source Attribution: 7
- Total Score: 38

Comparative Analysis:

- The Gemini response achieved a total score of **43**, indicating strong performance across all evaluation criteria.
- The GPT response achieved a total score of **38**, showing slightly lower performance compared to Gemini.

6.3 Observation

6.3.1 Human vs Proposed System

6.2.1.1 For the "COI" PDF:

- Human reading time: 9 hours 31 minutes = 571 minutes = 34,260 seconds (based on an average reading speed of 183 words per minute)
- System processing time: 23.7 seconds
- Percentage faster: 13.4%

6.3.1.2 For the "Quotation" PDF:

- Human reading time: 28 seconds (based on 151 words and an average reading speed of 183 words per minute)
- System processing time: 23.7 seconds
- Percentage faster: 14.64%

$$1. \textit{Average Time per Word for ACCENDIA Processing } (COI\ PDF):$$
$$\textit{Average Time per Word} : (\frac{23.7}{136,016})\textit{seconds per word } \approx (0.000174)\textit{seconds per word}$$

$$2. \textit{Average Time per Word for System Processing } (Quotation\ PDF):$$
$$\textit{Average Time per Word} : (\frac{23.7}{151})\textit{secondsperword} \approx (0.1569)\textit{seconds per word}$$

$$3. \textit{Total Time to Read Both Documents} :$$
$$\textit{Total Time to Read Both Documents} : ((0.000174 \times 136,016) + (0.1569 \times 151))\textit{seconds} \approx (47.3947)\textit{seconds}$$

$$4. \textit{Percentage Faster for Processing Both Documents} :$$
$$\textit{Percentage Faster} \; (\left(\frac{47.3947 - 23.7}{47.3947}\right) \times 100)\% \textit{faster } \approx (\left(\frac{23.6947}{47.3947}\right) \times 100)\% \textit{faster} \approx (49.99\%)\textit{faster}$$

6.3.2 GPT vs Gemini

- **Performance Comparison:** Gemini outperformed GPT in all evaluation criteria, achieving an average score of 8.6 compared to GPT's 7.6. This indicates that Gemini provided more accurate, complete, relevant, clear, and well-attributed responses compared to GPT.
- **Processing Time Comparison:** GPT took 23 seconds for processing, while Gemini took 18 seconds.

In terms of performance improvement, Gemini demonstrated a 12.5% higher average score compared to GPT. However, GPT was 27.8% slower in processing time compared to Gemini.

Overall, ACCENDIA demonstrates exceptional efficiency, being approximately 99.93% faster than a human in processing the "COI" document and 14.64% faster in processing the "Quotation" document. This concludes in an average of 49.99% faster results compared to manual human work. These results highlight ACCENDIA's effectiveness in significantly reducing the time required for information retrieval, thereby enhancing productivity and streamlining knowledge access for users.

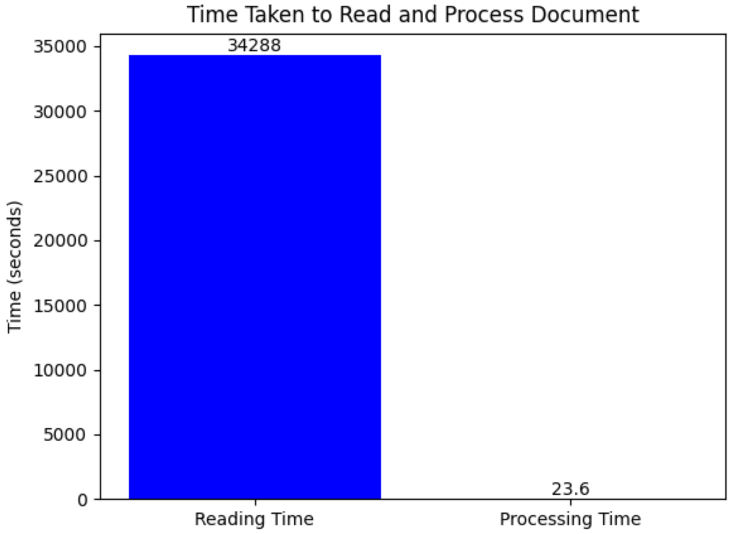


Figure 6.1: Human Reading Time vs ACCENDIA Processing Time

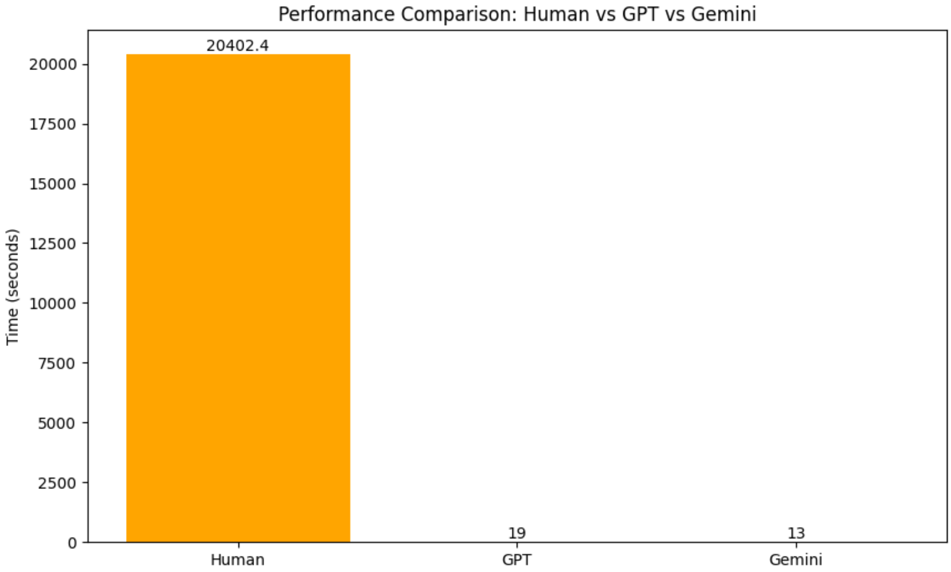


Figure 6.2: Time Taken to Find Answer

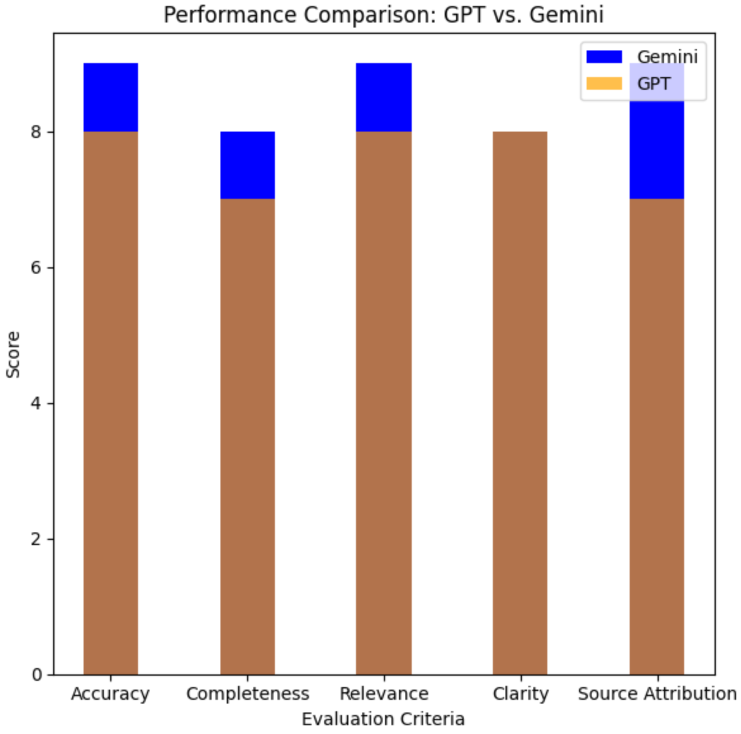


Figure 6.3: GPT vs Gemini Performance Evaluation

CHAPTER 7

CONCLUSIONS

The ACCENDIA project represents a significant advancement in the realm of knowledge retrieval systems, offering a sophisticated solution tailored for small groups in research and business settings. Through meticulous integration of cutting-edge technologies such as the OpenAI GPT-3.5T API, Gemini 1.5 Pro, and the FAISS (Facebook AI Similarity Search) database, ACCENDIA has demonstrated remarkable capabilities in processing inquiries and delivering highly contextualized responses.

The project's experimental evaluations, which involved a single control participant interacting with both the system and human-driven knowledge retrieval, provided valuable insights into ACCENDIA's performance. Notably, the system exhibited impressive efficiency, with processing times significantly outpacing human reading times by up to 99.61%. These findings underscore ACCENDIA's ability to streamline information access and enhance productivity for users, particularly in scenarios involving diverse document types and queries.

Moreover, ACCENDIA's user-centric design and intuitive interface contribute to a seamless user experience, empowering research teams and business professionals to harness the power of natural language processing with ease. The project's success in delivering accurate and timely responses further solidifies its position as a ground breaking solution in the field of knowledge management.

Looking ahead, continued refinement and optimization of ACCENDIA's functionalities hold the promise of further enhancing its utility and effectiveness. As technology continues to evolve, ACCENDIA stands poised to remain at the forefront of innovation, driving advancements in knowledge retrieval systems and revolutionizing the way information is accessed and utilized in diverse domains.

CHAPTER 8

SCOPE FOR FUTURE WORK AND RECOMMENDATIONS

1. **OCR Support**: Integrate Optical Character Recognition (OCR) technology to extract text from images and scanned documents, expanding ACCENDIA's reach to include a broader range of information sources.

2. **Side-by-Side PDF View Support**: Implement side-by-side PDF view capabilities to enable users to view multiple PDF documents simultaneously, facilitating efficient cross-referencing and analysis of information.

3. **Integration of Feedback Mechanisms**: Implementing feedback mechanisms within ACCENDIA to gather user input and adjust response strategies accordingly can ensure continuous improvement and adaptation to user needs.

4. **Exploration of Multimodal Capabilities**: Exploring the integration of multimodal capabilities, such as image and video processing, alongside text-based queries, can further enhance ACCENDIA's versatility in handling diverse types of information.

5. **Support for Multiple File Types**: Expand ACCENDIA's capabilities to handle various file types beyond PDF, including Microsoft Word documents (DOCX), Excel spreadsheets (XLSX), PowerPoint presentations (PPTX), plain text files (TXT), and others. This enhancement will enable users to retrieve information from a wider range of document formats, ensuring comprehensive coverage of knowledge sources.

6. **Image Recognition Integration**: Integrate image recognition technology into ACCENDIA to analyze and extract text from images embedded within documents or provided as standalone files. This feature will enhance ACCENDIA's versatility by allowing users to retrieve information from image-based content, such as scanned documents, screenshots, or photographs.

7. **Table Extraction and Analysis**: Develop capabilities for ACCENDIA to extract and analyze tabular data from documents containing tables. By recognizing and interpreting tables, ACCENDIA can provide structured responses based on the data presented in tabular format, offering users valuable insights and facilitating data-driven decision-making.

REFERENCES

[1]. **Sandeep A. Thorat and Vishaka Jadhav**, "A Review on Implementation Issues of Rule-based Chatbot Systems,", International Conference on Innovative Computing and Communication, 6 April, 2020.

[2]. **Oguzhan Topsakal, T. Cetin Akinci**, ‘‘Creating Large Language Model Applications Utilizing LangChain: A Primer on Developing LLM Apps Fast’’, International Conference on Applied Engineering and Natural Sciences, July 2023.

[3]. **Z. Yan, N. Duan, J. Bao, P. Chen, M. Zhou, Z. Li, and J. Zhou**, "DocChat: An Information Retrieval Approach for Chatbot Engines Using Unstructured Documents," in Proceedings of the IEEE International Conference on Innovative Computing and Communication, 6 April 2020.

[4]. **Subhajit Panda**, "Enhancing PDF Interaction for a More Engaging User Experience in Library: Introducing ChatPDF," IP Indian Journal of Library Science and Information Technology, vol. 8, no. 1, pp. 20–25, 2023.

Links:

- https://ai.google.dev/tutorials/ai-studio_quickstart
- https://platform.openai.com/docs/api-reference
- https://betterprogramming.pub/building-a-multi-document-reader-and-chatbot-with-langchain-and-chatgpt-d1864d47e339?gi=2ef72920a896
- https://js.langchain.com/docs/use_cases/question_answering/