DOCUMENTATION

Name: Ackyshma Fleur Fernandes

Department: Electronics and Telecommunication Engineering

College Name: Don Bosco College of Engineering

Table of Contents

Problem Statement

Building a Simple Q&A Chatbot with RAG and LangChain.

Technical Stack

- **Python 3.12.8**

  Core programming language used to build and execute the application.

- **OpenAI**

  Provides powerful language models for generating answers to user queries and creating embeddings for vector representation of text.

- **LangChain**

  Framework to seamlessly integrate LLMs with tools like vector databases and text processing for efficient question-answering workflows.

- **Qdrant**

  Vector database used to store and retrieve vectorized embeddings of text for similarity-based search.

- **Streamlit**

  Creates an interactive web-based user interface for the chatbot, allowing users to ask questions and view answers.

- **Pdfplumber**

  Extracts text from PDF documents to make the content accessible for embedding and retrieval.

Files and Documents

1. app.py

Contains the main code for:

- Processing PDF documents using pdfplumber.
- Generating vector embeddings for extracted text.
- Storing and retrieving data from Qdrant.
- Integrating LangChain for question-answering.
- Connecting the OpenAI API to generate answers.
- Building a user-friendly interface using Streamlit.

2. requirements.txt

Includes all the dependencies required to develop and run the chatbot.

- OpenAI
- LangChain
- Qdrant
- Streamlit
- Pdfplumber
- Python dotenv

3. .env file

Used to store sensitive credentials securely. Contains:

- OpenAI API key
- Qdrant Host URL
- Qdrant API key

4. PDF Documents

- Data Communication.pdf
- Digital Signal Processing.pdf

Workflow

1. PDF Processing:

- Extract text from uploaded PDF files using pdfplumber.
- Parse content into manageable chunks for efficient processing.

2. Embedding Generation:

- Converts text data into vector embeddings using the OpenAI model.
- Store embeddings in Qdrant for fast retrieval.

3. Query Processing:

- Accept user queries through the Streamlit Chat interface.
- Retrieve relevant document chunks from Qdrant using similarity search.

4. Answer Generation:

- Use LangChain to integrate retrieved documents and query the OpenAI API.
- Generates a concise, contextually accurate response.

5. Web Interface:

- Provide an interactive chat interface using Streamlit.
- Allow users to upload PDFs, type questions, and receive answers seamlessly.

Setup Instructions

Step 1: Install Dependencies

```
pip install -r requirements.txt
```

Step 2: Run the Application

```
streamlit run app.py
```

Design and Implementation

Chat Interface Description

The Chat Interface is designed to facilitate user interaction with the Q&A chatbot and provides a seamless experience for querying information from uploaded PDF documents.

Main components and functionalities are as follows:

1. Prompt Bar

The interface includes a text input field called the Prompt Bar, where users can type their queries. The prompt must be a question related to the content of the uploaded PDF documents, which in this case are:

- Data Communication.pdf
- Digital Signal Processing.pdf

2. Submit Button

- Once the user has entered a question in the Prompt Bar, they can click the Submit Button to submit their query.
- The chatbot processes the submitted question, retrieves relevant information from the PDF documents, and generates an accurate answer.
- The system uses advanced retrieval and language processing techniques to ensure that the answer is contextually relevant and based on the content of the uploaded PDFs.

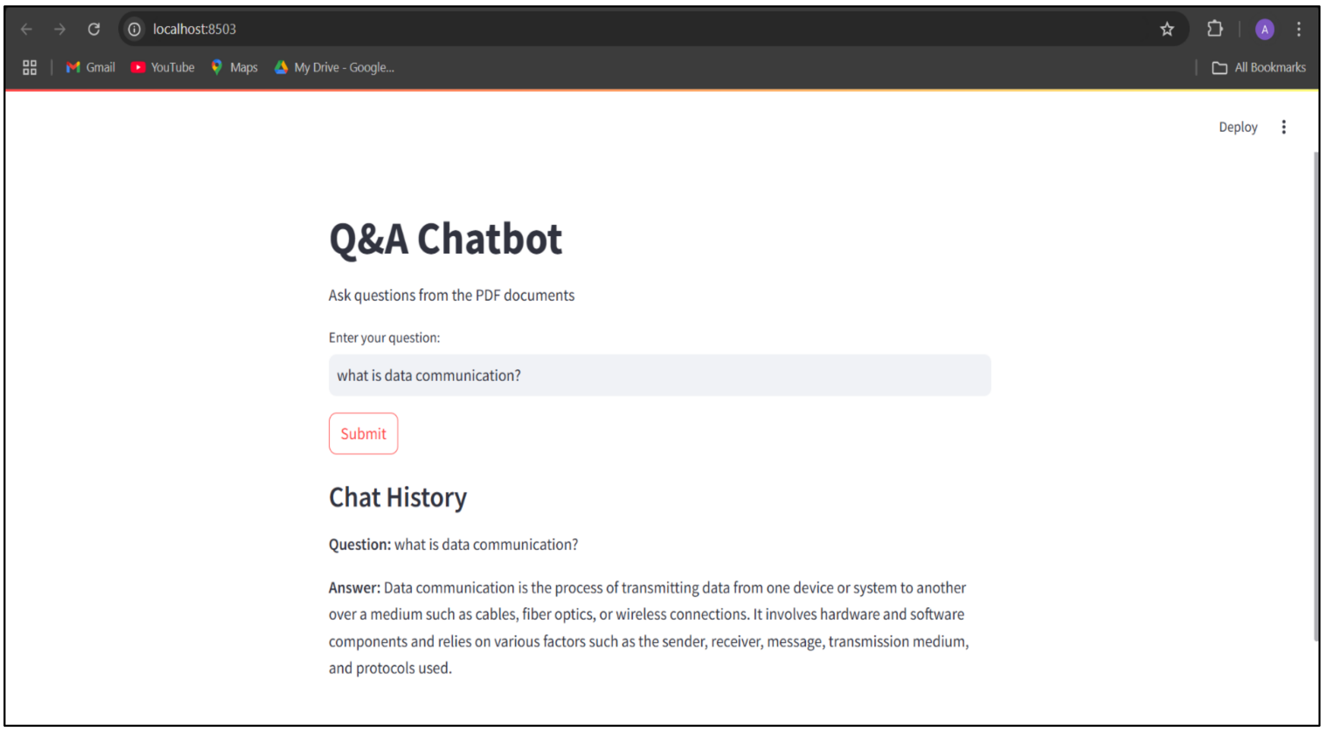
3. Answer Retrieval

- Upon submission, the system identifies the most relevant sections of the PDF documents using similarity search in the vector database (Qdrant).
- These sections are provided as context to the language model (OpenAI), which then formulates a coherent answer based on the retrieved information.

4. Chat History:

- The interface maintains a Chat History that displays all previous interactions during the session.
- Each entry in the Chat History includes:
  - The Question submitted by the user.
  - The corresponding Answer generated by the Chatbot.
- This feature allows users to review past queries and answers.

1. First Prompt



Q&A Chatbot

Ask questions from the PDF documents

Enter your question:

what is data communication?

Submit

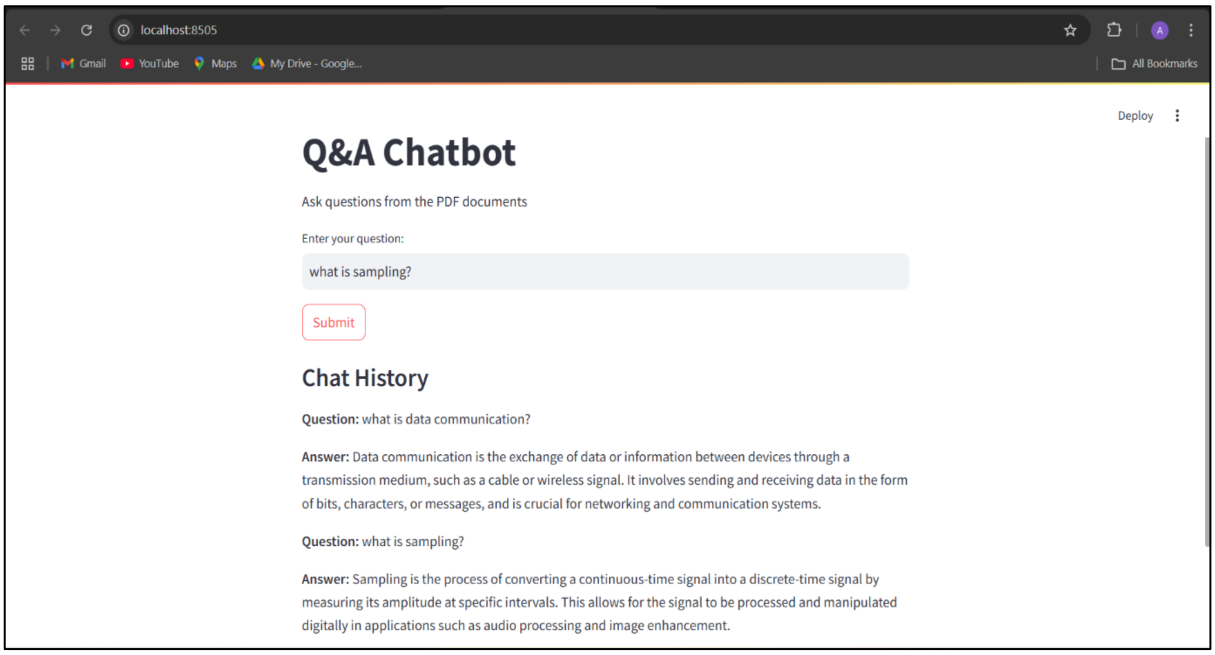
Chat History

Question: what is data communication?

Answer: Data communication is the process of transmitting data from one device or system to another over a medium such as cables, fiber optics, or wireless connections. It involves hardware and software components and relies on various factors such as the sender, receiver, message, transmission medium, and protocols used.

Fig. 1

2. Second Prompt



Q&A Chatbot

Ask questions from the PDF documents

Enter your question:

what is sampling?

Submit

Chat History

Question: what is data communication?

Answer: Data communication is the exchange of data or information between devices through a transmission medium, such as a cable or wireless signal. It involves sending and receiving data in the form of bits, characters, or messages, and is crucial for networking and communication systems.

Question: what is sampling?

Answer: Sampling is the process of converting a continuous-time signal into a discrete-time signal by measuring its amplitude at specific intervals. This allows for the signal to be processed and manipulated digitally in applications such as audio processing and image enhancement.

Fig. 2