■ PDF Chatbot with Ollama

Project Documentation

Prepared by: Polipalli Rajesh

1. Abstract / Overview

The PDF Chatbot with Ollama is an interactive system designed to allow users to upload PDF documents and interact with them using natural language queries. The chatbot leverages Ollama as a local Large Language Model (LLM) server to generate context-aware responses. By integrating PDF text extraction and conversational AI, this project aims to simplify document understanding and improve accessibility of information.

2. Objectives & Features

- Provide a user-friendly interface using Streamlit. - Enable PDF text extraction with multiple libraries (PyPDF2, pdfplumber). - Allow users to query documents in natural language. - Integrate with Ollama for local LLM inference. - Support adjustable model parameters (temperature, top_p). - Export chat history and extracted text in JSON/TXT format. - Handle connection errors and timeouts gracefully.

3. System Requirements

• Operating System: Windows 10/11 or Linux with Python 3.9+ • Dependencies: Streamlit, Requests, PyPDF2, pdfplumber, ReportLab • Ollama installed and running (`ollama serve`) • At least 8 GB RAM (more recommended for large models) • Web browser (Chrome/Edge/Firefox) for UI access

4. Architecture & Workflow

The chatbot consists of the following main components: 1. **User Interface**: Built with Streamlit, providing PDF upload, chat, and export options. 2. **PDF Text Extraction**: Uses PyPDF2 and pdfplumber to process uploaded PDFs into structured text. 3. **Chat Engine**: Sends user queries and PDF context to Ollama via REST API. 4. **Ollama Server**: Runs locally to process LLM requests and return context-aware responses. 5. **Output Management**: Displays responses, saves chat history, and allows export in JSON/TXT formats. The workflow follows these steps: • User uploads a PDF • Extract text page-by-page • User asks a question • Context and question sent to Ollama • Ollama streams back the response • Chat history is updated and can be exported

5. Step-by-Step Usage Guide

1. Start Ollama server: `ollama serve` 2. Launch the app: `streamlit run app.py` 3. Upload a PDF file through the interface. 4. Use the chat box to ask questions about the document. 5. Download extracted text or chat history if needed. 6. Adjust model settings (temperature, top_p) in the sidebar. 7. Use reset/clear chat buttons for session management.

6. Future Enhancements

- Multi-document upload and querying. - Improved GPU acceleration for faster inference. - Adding support for summarization and keyword extraction. - Integration with cloud-hosted LLMs as fallback. - Enhanced PDF parsing for scanned/image-based documents using OCR.

7. Conclusion

The PDF Chatbot with Ollama demonstrates the potential of combining local LLM inference with document processing techniques to create a powerful knowledge assistant. This system not only improves accessibility to information stored in PDFs but also provides a foundation for building advanced AI-driven tools in research, education, and industry. Future improvements will further extend its capabilities and performance.