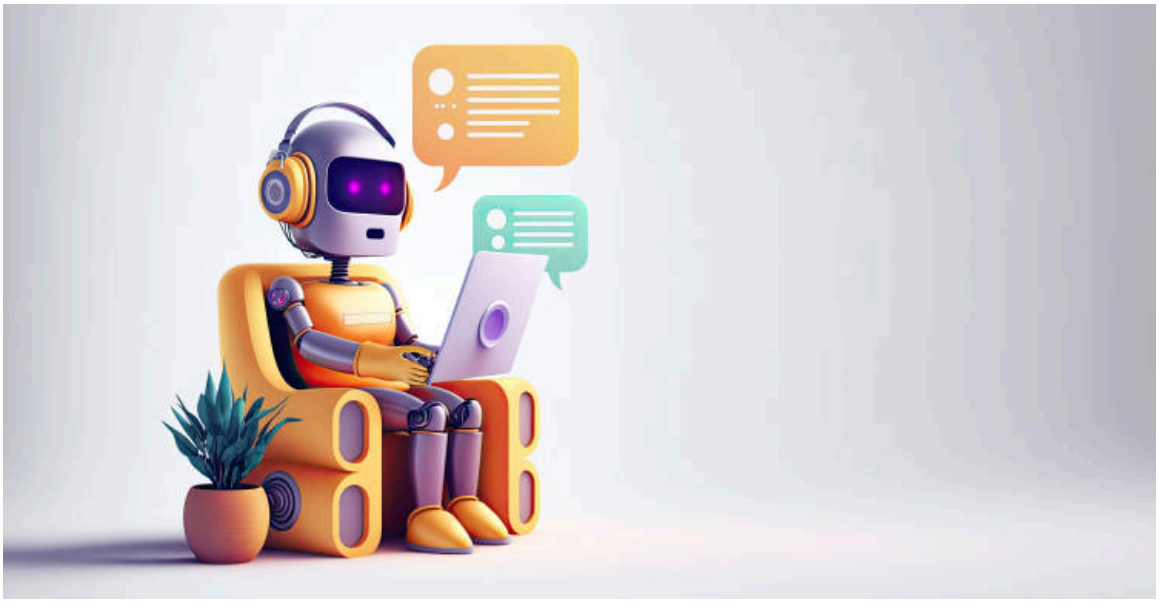


Documentation of G_chatbot

BY TEAM 14

Name | Course Title | Date

Documentation of the Godrej_chatbot-case study

Document Processor

This Flask application processes various document types (PDF, PPT, Word) to extract and summarize their content. It also provides a chat interface for users to ask questions based on the extracted summaries.

Features

- Upload PDF, PPT, and Word documents.

- Extract text and images from documents.

- Generate summaries of the extracted content.

- Allow users to ask questions based on the summary.

Dependencies

To install all required dependencies, use the following `pip` command:

```bash

**pip install Flask Flask-SQLAlchemy pyngrok transformers pillow fitz pdfplumber python-pptx torch torchvision pytesseract groq-client python-docx**

```

Hardware requirements:

the minimum requirement is cpu which makes it work on any computer

Code Overview

Steps:

Step 1: Import Required Libraries

- Import necessary libraries for Flask, file handling, text and image processing, database management, and AI models.
- import os
- import io

- from flask import Flask, render_template_string, request, session
- from flask_sqlalchemy import SQLAlchemy
- from pyngrok import ngrok
- from transformers import pipeline
- from PIL import Image
- import fitz # PyMuPDF
- import pdfplumber
- from pptx import Presentation
- from pptx.enum.shapes import MSO_SHAPE_TYPE
- import torch
- import torchvision.transforms as transforms
- from torchvision import models
- import pytesseract
- import groq
- from docx import Document
- from datetime import datetime
- import re

Step 2: Define the Flask App

- Initialize Flask app and configure SQLAlchemy for database operations.
- Define the `Summary` model for storing document summaries.
- Define the `QuestionAnswer` model for storing questions and answers.

Step 3: Utility Functions

- `preprocess_text(text)`: Clean and preprocess text.
- `preprocess_image(image_path)`: Preprocess images for model input.
- `extract_images_from_ppt(ppt_path)`: Extract images from PPT files.
- `extract_text_from_image(image_path)`: Extract text from images using OCR.
- `extract_text_from_pdf(pdf_path)`: Extract text from PDF files.
- `extract_text_from_ppt(ppt_path)`: Extract text from PPT files.
- `read_word_file(file_path)`: Read and extract text from Word documents.
- `summarize_text(text)`: Summarize large chunks of text.
- `extract_limited_images_from_pdf(pdf_path, image_dir, limit=4)`: Extract a limited number of images from PDF files.
- `merge_captions_and_text(text, image_captions)`: Merge text and image captions.
- `chunk_text(text, chunk_size=1000)`: Split text into smaller chunks.
- `summ(text, chunk_size=1000)`: Summarize text using Groq API.
- `answer_question(question, context)`: Generate answers to questions based on provided context using Groq API.

Step 4: Define Routes and Templates

- /: Handle file uploads, text extraction, summarization, and Q&A through forms and render results using HTML templates.

Instructions:

Installation and Setup Instructions

1. Install Dependencies:

Use the provided pip command to install all necessary libraries:

pip install Flask Flask-SQLAlchemy pyngrok transformers pillow fitz pdfplumber python-pptx torch torchvision pytesseract groq-client python-docx

2. Set Up the Flask App:

Save the provided code into a Python file, e.g., `app.py`.

3. Run the Flask App:

Execute the Flask application using the following command:

```

python app.py

```

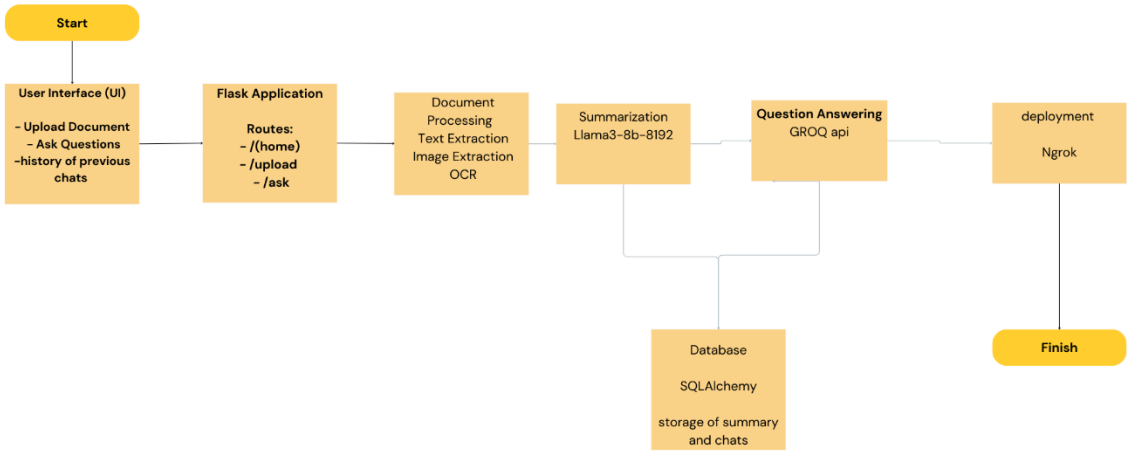
4. Access the App:

Use the URL provided by ngrok (displayed in the console) to access the application in your browser.

5. Upload Documents:

Upload PDF, PPT, or Word documents to extract and summarize their content. Ask questions based on the summaries.

Architecture:


Start
User Interface (UI)
- Upload Document
- Ask Questions
-history of previous chats
Flask Application
Routes:
- /(home)
- /upload
- /ask
Document Processing
Text Extraction
Image Extraction
OCR
Summarization
Llama3-8b-8192
Question Answering
GROQ api
deployment
Ngrok
Database
SQLAlchemy
storage of summary and chats
Finish

S