RAG

1. Imports:

- **Langchain Components**:
  - `RunnablePassthrough`: Passes input (question) directly through the processing chain.
  - `HuggingFaceInferenceAPIEmbeddings`: Generates embeddings from the document content using Hugging Face API.
  - `StrOutputParser`: Parses the output of the model into a string.
  - `ChatPromptTemplate`: Formats the prompt for the language model.
  - `RecursiveCharacterTextSplitter`: Splits large texts into smaller, manageable chunks for better processing.
  - `Chroma`: Stores the document embeddings for vector-based search.
  - `PyPDFLoader`: Loads the content of PDF files.
  - **Streamlit**: Used to create a user-friendly web app interface.
  - **dotenv**: Loads API keys from a `.env` file.
  - **ChatGroq**: A large language model (LLM) used to generate responses based on the retrieved document context.

2. Environment Setup:

- **`load_dotenv()`**: Loads environment variables (`HF_TOKEN` and `GROQ_API_KEY`) from a `.env` file for API authentication.

3. Function: `pdf_file(file, question):`

- **Input**: A PDF file and a question from the user.
- **PDF Handling**:
  - The uploaded PDF file is saved temporarily, and its content is loaded using `PyPDFLoader`.
- **Text Splitting**:
  - The PDF content is split into chunks of 1000 characters, with a 200-character overlap.
- **Embeddings**:
  - Embeddings for the document chunks are generated using Hugging Face's model (`BAAI/bge-base-en-v1.5`) and stored in a **Chroma vector store**.
- **Retriever**:
  - A retriever is created to retrieve relevant document chunks based on vector similarity. It searches for the top 3 matches.
- **Prompt Template**:
  - A template is designed to ask the question based on the retrieved document context.
- **Language Model**:
  - The **ChatGroq** model (`llama-3.1-8b-instant`) is invoked to generate an answer based on the provided context and question.

- **Processing Chain**:
    - The document retriever, prompt template, and model are combined into a chain to process the user's input and produce an answer.

4. Streamlit Application (`main()` function):

- **UI Setup**:
    - A simple interface is created using **Streamlit**, where users can:
        - Upload a PDF file.
        - Ask a question via text input.
- **Answering Process**:
    - Upon clicking the "Get Answer" button, the uploaded PDF and user question are passed to the `pdf_file()` function.
    - The chatbot processes the input and displays the answer.
- **Conversation History**:
    - The conversation history, including user questions and bot answers, is displayed below the input section.

OUTPUT

QUESTION:
1. which dataset is used to train the florence 2 model

ANSWER BY BOT:

The dataset used to train the Florence-2 model is FLD-5B, which contains an extensive collection of 126M images paired with 5B comprehensive annotations.

2. what is comprehensive multi tasking?

Answer by bot:

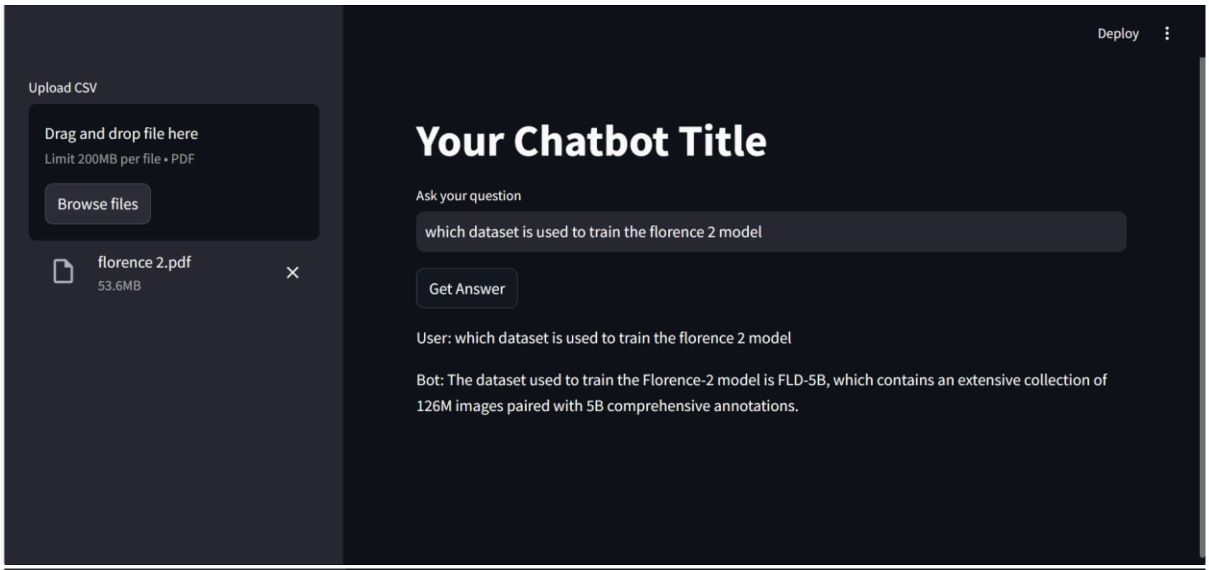
Comprehensive multitask learning is a learning approach that involves formulating a range of multitask learning objectives, each tailored to address specific aspects of visual comprehension. This approach incorporates multiple learning objectives, each addressing a different level of granularity and semantic understanding.

In the context of the provided document, comprehensive multitask learning is used to develop a versatile vision foundation model. The approach involves three distinct learning objectives:

1. Image-level understanding tasks that capture high-level semantics and foster a comprehensive understanding of images through linguistic descriptions.
2. Other objectives that align with the predefined criteria of spatial hierarchy and semantic granularity, inspired by recent research on multitask learning.

The goal of comprehensive multitask learning is to effectively capture visual understanding nuances and bridge the gap between vision and language understanding.

Screenshot:



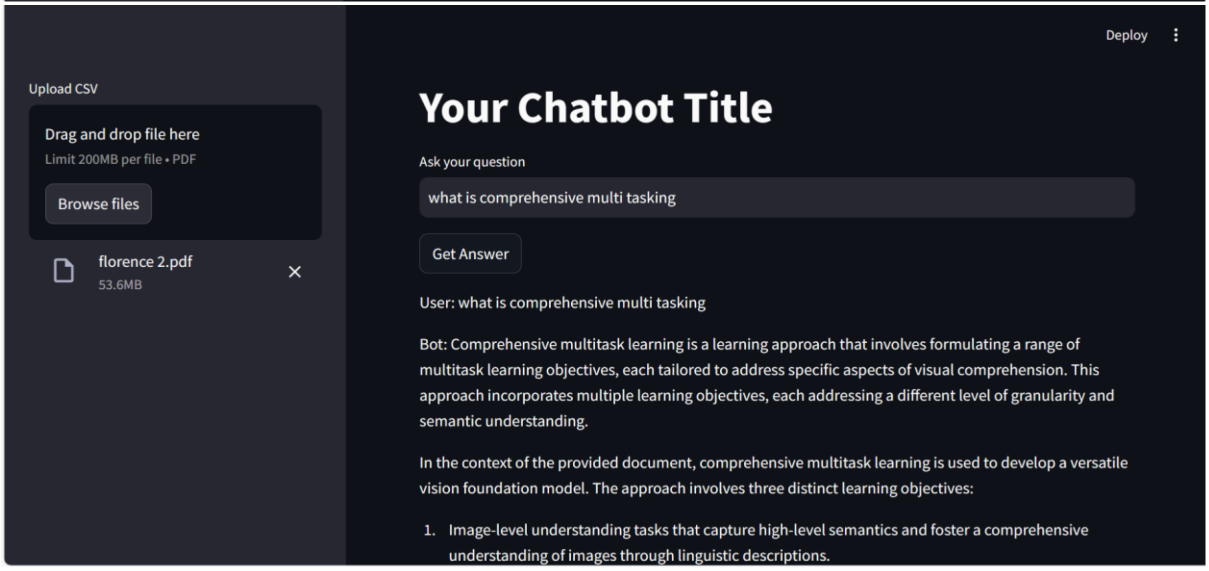
Your Chatbot Title

Ask your question

which dataset is used to train the florence 2 model

User: which dataset is used to train the florence 2 model

Bot: The dataset used to train the Florence-2 model is FLD-5B, which contains an extensive collection of 126M images paired with 5B comprehensive annotations.



Your Chatbot Title

Ask your question

what is comprehensive multi tasking

User: what is comprehensive multi tasking

Bot: Comprehensive multitask learning is a learning approach that involves formulating a range of multitask learning objectives, each tailored to address specific aspects of visual comprehension. This approach incorporates multiple learning objectives, each addressing a different level of granularity and semantic understanding.

In the context of the provided document, comprehensive multitask learning is used to develop a versatile vision foundation model. The approach involves three distinct learning objectives:

1. Image-level understanding tasks that capture high-level semantics and foster a comprehensive understanding of images through linguistic descriptions.