As suggested in the assignment the dataset should be of a health care with patient query and professional response, but which is not available in the link <https://data.gov.in/sector/Finance>.

so i took a data set available in the above link Capitalaccountreceipts\_0

so as an initial step you need to download the phi3 model from hugging face and give the model path in code.

**model running**:

To run the model just type a command in command promt : ‘streamlit run file\_name.py’ so your server starts running and you can see a web page in browser.

**Model Working:**

I have made a sample RAG architecture to make a Chatbot which provides answers to questions using excel or csv as a database.

**1- preprocessing data:**

This process includes removing the noise in the data using panda’s library which includes replacing nan values, removing unnecessary rows and columns.

**2 - Loading CSV:**

To load the preprocessing csv file, I used a langchain framework which has csv loader which loads the file to the llm

**3 - Converting embeddings:**

The loaded csv file is converted into embeddings using an open-source embedding model so that the organisation data will be safe within local,

we need to do embeddings because the model can understand only the embeddings and it not able to understand the text.

**4 - Loading embeddings into vector DB**

Here I have used a FIASS vectorDB an opensource vectorDB which is used to store the embeddings which is created previously.

**5 - LLM and Retrieval chain:**

LLM: I have used an opensource LLM called PHI3 because the model will be in our local server and security to the organisation data

so next making a chain using langchain so chaining all the retriever , embeddings and query into the llm to answer the query

**6 - Streamlit interface:**

I have used streamlit as an interface to interact with the model which is very simple to create for time being but for production purpose use Django or flask to create APIS

**question 1 : Why used LLM why not LSTM or GRU or DL models**

To Data to train A chatbot for DL models like GRU or LSTM the data should be in a conversation type which is not available in

the link provided as its a excel file which making a RAG architecture with Open source LLM which provides a good results.

so choosing the model dataset plays a primary role! There are more number of hyperparameters in RAG and LLMS to control the context and desired output

**question 2: How to evaluate a rag model?**

A rag model can be evaluated by using RAGAS framework in which we have a metrics like Faithfulness, answer relevancy,

context precision, context recall which is used to evaluate a rag architecture which is working with highest amount of accuracy or not.

**question 3: Potential issues in model?**

As the model I have prepared a basic rag structure and there is other structure called advance rag structure which makes up a huge time, as the time provided is low so I made a demo model.

**question 4: How to solve the potential issues:**

1 - Using the advance Rag structure which includes rerankers, advance retrievers and similarity search

2 - Making code as production grade using flask /Django to create APIs to access the model

3 - preprocessing the data set before sending to model

4 - Fine tuning the OpenAI model so that the model gets new weights according to the input data to be trained

5 - By trial and error methods which include changing of hyperparameters like no of tokens, chunk size, chunk overlap, temperature etc....

6 - using various prompting technics and improving the prompt according to the output generated.

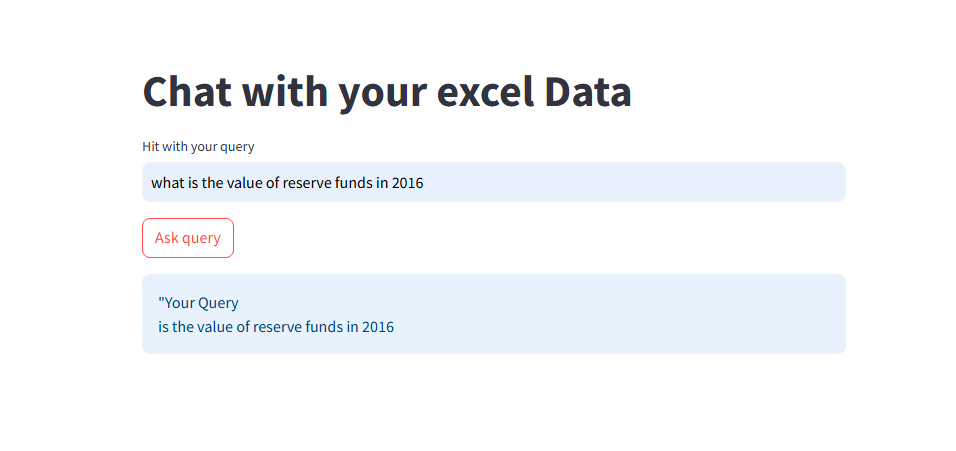
7 - By writing code in for of try and exception method

**question - 5: insights from the model?**

The model answers all the questions that provided in the data set if its outside the data set it throws you a response.

This is a basic model so it cannot work on complex excel data which i observed by testing it with different questions and

changing simple to complex excel file.



The above snap is the interface which is created by the streamlit library and when you ask the question and hit ask query it will connect to LLM and generate answer and question provided in the snap is from the excel file in the repository

The below snap provides the answer generated by the LLM

