DSCI-560 Assignment No. 6 - Part 2
Custom Q&A Chatbot

This is the second part of a 2-part assignment where you will gain experience building a Question & Answer (Q&A) chatbot using Large Language Models (LLMs) and embedding models. In this part, you will work with your team to replicate the solution of part 1 using open-source models, then create a GUI for your chatbot. You will use a GUI/Website to analyze uploaded PDF documents on the fly, then through another GUI/Web interface, ask questions and have the system generate the answer based on the set of documents uploaded by the user.

1. **Initial Setup**

   Resources

   As before, you may use the source codes in the following drive.
   (***https://drive.google.com/drive/folders/1hdUoDvtQoFkIbJyUr8kwDQghLNTptoCQ***)

   Tools / Libraries

   For the assignment, you will need to use Python scripts and other useful tools in the Linux environment. ***(Make sure to document any setup steps / requirements for running your scripts in the document you submit)***

   The **Drive folder** has a **Readme.md** file with all the setup instructions and a list of libraries and their versions and installation steps required for Parts 1 and 2 of Lab 6. Use the ***App.py and html.py*** as reference for this lab. ***You may use any tools to create a GUI for the chatbot, including any combination of PHP, Python, Javascript, HTML, StreamLit, etc.***

2. **Domain-Specific Chatbot with Open-Source Resources**

   In Part 1, you may have noticed that OpenAI charges fees for using their tools and models. This development path may not be the best for developing and deploying domain-specific chatbots, especially because functionally equivalent solutions with acceptable performances can be built with open-source tools and models. Moreover, developers may not want ChatGPT to analyze their original proprietary data that contains domain-specific answers. In some cases, the solution may not have access to the Internet.

   Therefore, your task in Part 2 of Lab 6 is to replicate the solution for Part 1 using only the open-source tools and models. You may use any open-source local embedding models and LLMs if it can produce similar output as OpenAI embeddings. Many open-source text embedding models are available at https://huggingface.co/models

   Additional guidance and information may be found at https://sbert.net/, and the following video gives a good summary of open-source text embedding models and how to use them https://www.youtube.com/watch?v=QdDoFfkVkcw

3. **Web Interface Design**

For this task, you will focus on designing a user-friendly chatbot interface using HTML and CSS. Implement JavaScript to handle user interactions and communicate with the chatbot.

a. **PDF Upload**
Provide a sidebar option where the user can upload single or multiple PDF documents for analysis.

b. **Analyzing PDFs**
Store the PDFs input by the user and execute your Python script to analyze the input PDFs, extract text, create chunks, generate word embeddings, and store it in the vector database.

c. **Input Field and Chat Window**
Create a chat window with an input field for the user to ask questions based on the pdf and display all the messages as conversations. Store the questions and messages in chat history for later use.
Refer to the ***html.py*** for some starter code on displaying user and bot messages.

4. **Team Discussions**
Your team is expected to meet in person / virtually each day of the week and discuss the assignment progress & next steps. Document and compile minutes of all meetings in a separate file called ***'meeting_notes_A6_P2_<team_name>.pdf'***

Submission

Make one submission per team. Each team must submit all the code files for the working solution, a readme document containing information for running the code in PDF format, and a document that outlines the minutes of all team meetings in PDF format.

Provide a video per team that demonstrates the entire working solution and explains how the data tables were loaded, and data was visualized. Please include the team name and the names of all three team members in the video.

There will be a 50% penalty for all late submissions.