AI Patient Summaries Demo Usage

Marlon Lugo, Gordon Wong, John Bowen

This is a document detailing the usage and future concerns regarding the demo of this AI Patient Summary Google Collab File:
https://colab.research.google.com/drive/1dROVrPf1YRQM_mjDzF9ysOtJ0Y8Oi3LJ?usp=sharing
You should be able to make a copy of this file where you can make edits as needed, though the only action needed to run the demo without code modification is dropping the file to be analyzed in the collab environment, more detailed instructions are below.

This file also requires a Replicate API token, which can be generated at https://replicate.com/ from your account under "API Tokens".

Usage:

1) Make a copy of the demo file my going to File → Save a copy in drive.
2) Go to table of contents and click the file icon on the left side of the environment.
3) When in the files section of table of contents, click the "upload to session storage" button, the icon farthest to the left.
    a) Select the file that you want to analyze, Whisper is capable of reading most common video/audio formats. (Specifically, mp3, mp4, mpeg, mpga, m4a, wav, and webm)
4) Once you have the files(s) you want to analyze uploaded, run each code block either by clicking the play button that appears when you hover over a code segment, or by clicking in a code segment and pressing Cmd Enter, or Shift Enter. The first block will install the needed packages for Whisper and Replicate.
5) The next code block will prompt you when ran to input the name of the file you want to analyze, here is where you type the name of the file you uploaded, with the format of the file included. (Ex: recording.mp4) Then hit enter once you have the correct file name. If you have multiple files, you would return to this step when you want to move onto a different file, as they can only be analyzed one by one.
6) The next code block will also prompt you when ran, this time for your Replicate API token. This can be found in your Replicate account under "API Tokens". Copy paste this token (It should look like a random sequence of letters and numbers) and hit enter.
7) Run the next two code segments, one defines prompt_yn() which submits a single prompt and the other defines three_prompt_yn() which submits three prompts at once. The details of these are not necessary to know, you just have to

run the blocks so we can run them later. There should be no output after running these segments.

8) In the next code segment, you should see the list of prompts used to evaluate the transcript. Run this segment, and you should see an output of groups of three prompts with their response from the model, along with a few single prompts at the end.
9) Run the next segment to see the evaluation in percent format
10) Run the next segment to define the function prompt_summary(), again you should see no output
11) Run the next segment to get the summary of key points discussed in the transcript

Future Direction:

Implementation for processing batches of files would be pretty easy to implement, as well as some sort of table to store the evaluation value and summary for each file that is processed. Creating functions that would expand upon any one rubric criteria would also be easy to implement based on the other prompting functions.

One large area that could be another area of research would be to use AI in order to read and process the rubric into a series of criteria that you can use for prompting evaluation. If this is not realistic or the scope is too large for one semester, another option would be having some sort of implementation to make setting the rubric prompts for evaluation easier for future pharmacy school staff. Though a UI was deemed not necessary for our project, some sort of simple UI may help in this case if usability becomes more of a priority.

Looking into a version that runs Llama2 locally could also be an interesting direction to research, as features like specific prescription name recognition could be trained assuming you have access to labeled data including these prescription names. Further fine tuning of the LLM including expanded summary details are also another route that the project could take, though the end product would likely be less accessible to general staff.