**Summer Training Report  
Week 3**

**Data Vectorization and LLM Querying**

**Submitted by**

Name: Mahmoud Mustafa Oushi

University ID: 4312239

University: University of Prince Muqrin

Department: College of Computer Science and Cyber Security

Advisor: Dr.Ahmed Elhayek

**Training Organization**

Hulool Al Sahabah for IT & Communications Co

AI Department

**Report Topics – Week 3**

* RAG Systems
* Vectorizing JSON data
* Using FLAN-T5, GPT2, and Groq
* Linking an LLM to FastAPI

**1. RAG Systems**

Retrieval augmented generation or (RAG) for short is what can be described as a middle agent that is able to understand the human natural language request and then retrieve the required most suitable answer from a vectorized database at bay.

**1.1. RAG Components**

* **Embedding Model:** A machine-learning model that receives chunks of text as inputs and produces a vector usually between 256 and 1024 dimensions. This embedding represents the meaning of the chunk of text in an abstract space. The similarity/proximity of the embedding vectors is interpreted as semantic similarity
* **Vector Database**: A database built for handling storage and retrieval of vectors. These databases typically have highly efficient ways to compare vectors according to predetermined similarity measures.
* **Large Language Model (LLM):** A machine-learning model that takes in a textual prompt and outputs an answer. In a RAG system, this prompt is usually a combination of retrieved contextual information, instructions to the model, and the user’s query.

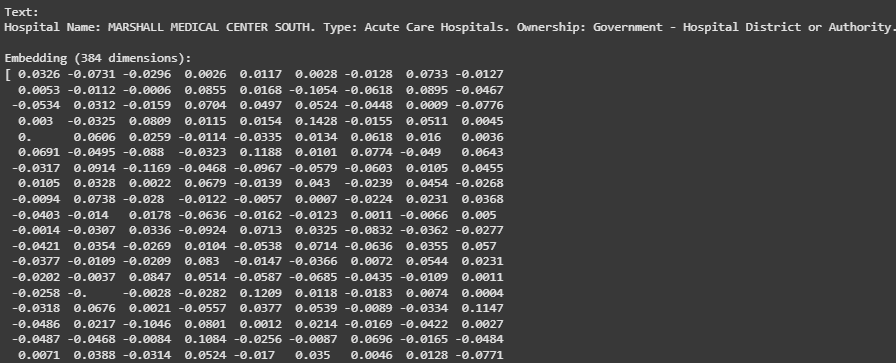
**2.Vectorizing JSON data**

One of the first tasks this week was to learn how to convert the data I have into a vector format readable by the model responsible for retrieving information in a dynamic way which is essential in RAG systems which is part of the tasks of this week.

To go into more details about why and how, I managed to find so examples in stack overflow as well as a GitHub repository in which a model called "all-MiniLM-L6-v2" was used to embed the data into vector format.

The process went as follows, I uploaded my JSON file which is the same one I have been using for all the previous tasks, afterwards I converted it to a data frame for ease of use. Then while going online I read that when trying to do such task for data that has many fields or entries, it is always better to wrap them in text serialization. And so that is what I did for each line in the json. By line I mean that each feature. After wards Installed “langchain sentence-transformers” and imported the model mentioned previously and just fed it the wrapped data.

Example:



**3. Using GPT2, FLAN-T5, and Groq**

After successfully…(I hope) vectorizing the data to the suitable format, I began to see what kind of LLM models I can use. I had already played and tested with FLANT-TF before and I would say those experiments were a success. So I decided to include it to a list of stuff I wanted to test. Starting with Llama which is not included in the list of models because it simply **ALMOST CRASHED MY POOR DEVICE**. I don’t know why that is, it was funny to have that happen. Thankfully nothing serious came out of it other than colab giving me the side eye. So after that crash I searched for the most light and CPU friendly model I could use. And I stumbled upon our first model GPT2.

**3.1. GPT2**

The Model that outputted the most text yet gave the least benefit out of them all. To be fair the model is light and user friendly…..to install and run. And I am sure if faced with small amount of data and simple (very simple) queries that it will perform well. The bellow is the prompt and the answers given by the model. (the yellow text is the answer in case the image is too blurry to read)

A screenshot of a computer

Description automatically generated

Answer:

Use the following pieces of context to answer the question at the end. If you don't know the answer, just say that you don't know, don't try to make up an answer.

Hospital Name: GREATER REGIONAL MEDICAL CENTER. Type: Critical Access Hospitals. Ownership: Government - Local. Overall Rating: 4. Mortality: Not Available. Safety: Not Available. Readmission: Same as the national average. Patient Experience: Above the national average. Effectiveness: Same as the national average. Timeliness: Above the national average. Efficiency of Imaging: Same as the national average.

Hospital Name: EAST LOS ANGELES DOCTORS HOSPITAL. Type: Acute Care Hospitals. Ownership: Proprietary. Overall Rating: 2. Mortality: Same as the national average. Safety: Same as the national average. Readmission: Same as the national average. Patient Experience: Below the national average. Effectiveness: Same as the national average. Timeliness: Below the national average. Efficiency of Imaging: Not Available.

Hospital Name: THEDACARE MEDICAL CENTER NEW LONDON. Type: Critical Access Hospitals. Ownership: Voluntary non-profit - Private. Overall Rating: 3. Mortality: Not Available. Safety: Not Available. Readmission: Same as the national average. Patient Experience: Same as the national average. Effectiveness: Same as the national average. Timeliness: Not Available. Efficiency of Imaging: Not Available.

Question: Which hospital has the best patient experience?

Helpful Answer: The National Health Service says:

"High-quality care is one of the most important factors for improving patient health. The high-quality care that is provided in hospitals is a necessary part of a successful hospital's overall health outcome. All hospitals should offer the highest quality of patient care and patient satisfaction. Hospital employees and health care professionals should be trained to evaluate, compare and improve patient experience, and report their findings to the National Health Service. Hospitals must follow the National Health Service's recommendations to improve patient experience and improve patient safety."

To see a list of hospitals that offer patient satisfaction, click here.

**3.2. FLAN-T5**

FLAN-T5. The powerhouse of the notebook cells. At least the base version of the model. I did try to look at some other larger FLAN models, but they did require a lot of installation and GBs and since I am using my own mobile data to work on all of this. Sparing almost 12 GBs just to download the immediate next larger model to the current one is a bit….wasteful. I am sure the performance would have been better. But as the time of writing this. It has been just the base model and playing with the user query and model prompt.

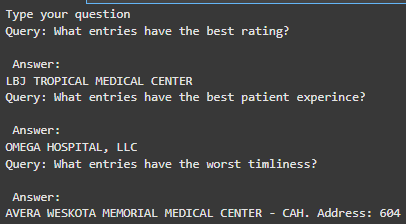
The first pic and yellow text are from the first variant I used which was loading the file via the path and manually wrapping the features based on it

A screenshot of a computer program

Description automatically generated

{'query': 'Which hospital has the best timeliness and patient experience?', 'result': 'GREATER REGIONAL MEDICAL CENTER. Type: Critical Access Hospitals. Ownership: Government - Local. Overall Rating: 4. Mortality: Same as the national average. Safety: Not Available. Readmission: Same as the national average. Patient Experience: Above the national average. Effectiveness: Same as the national average. Timeliness: Same as the national average. Efficiency of Imaging: Same as the national average.'}

The bellow one allows you to upload a file and type in your query. More flexible for usage but less range in data retrieving. Auto text wrapping as well. AUTOMATION!!!!!



**3.4. Groq**

I came across this API when I somehow used over 1mil Gemini tokens and had to look for an alternative. openAI are the Mr. Krabs of API, as soon as I ran an experiment the quota ended, and they required a payment for continuation. So, I looked into other APIs and that’s when I came across this one.

By far the best one just for the advantage of being able to provide the logic behind why it chose what it chose.



Groq Answer:

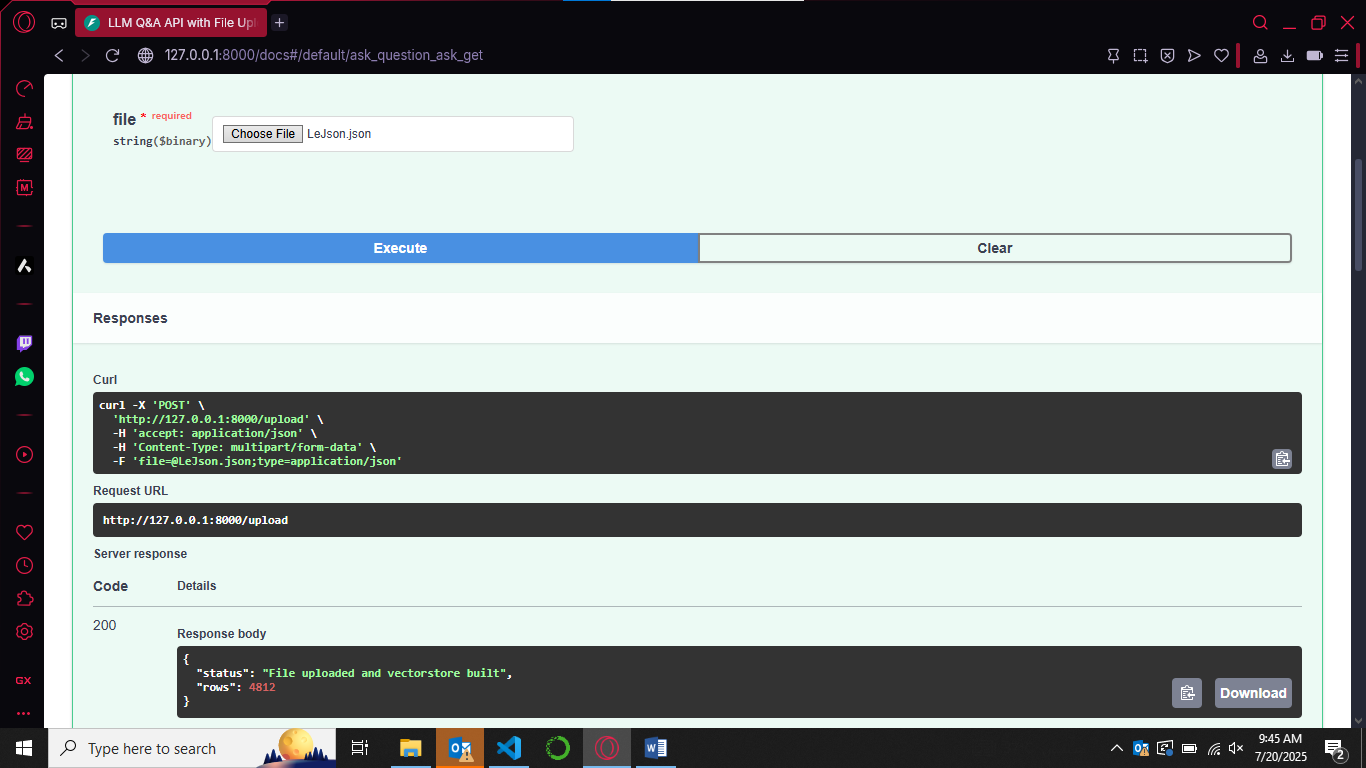
According to the provided data, the Greater Regional Medical Center and the Regional Medical Center both have "Above the national average" Patient Experience, while the Weeks Medical Center does not have available data for Patient Experience.

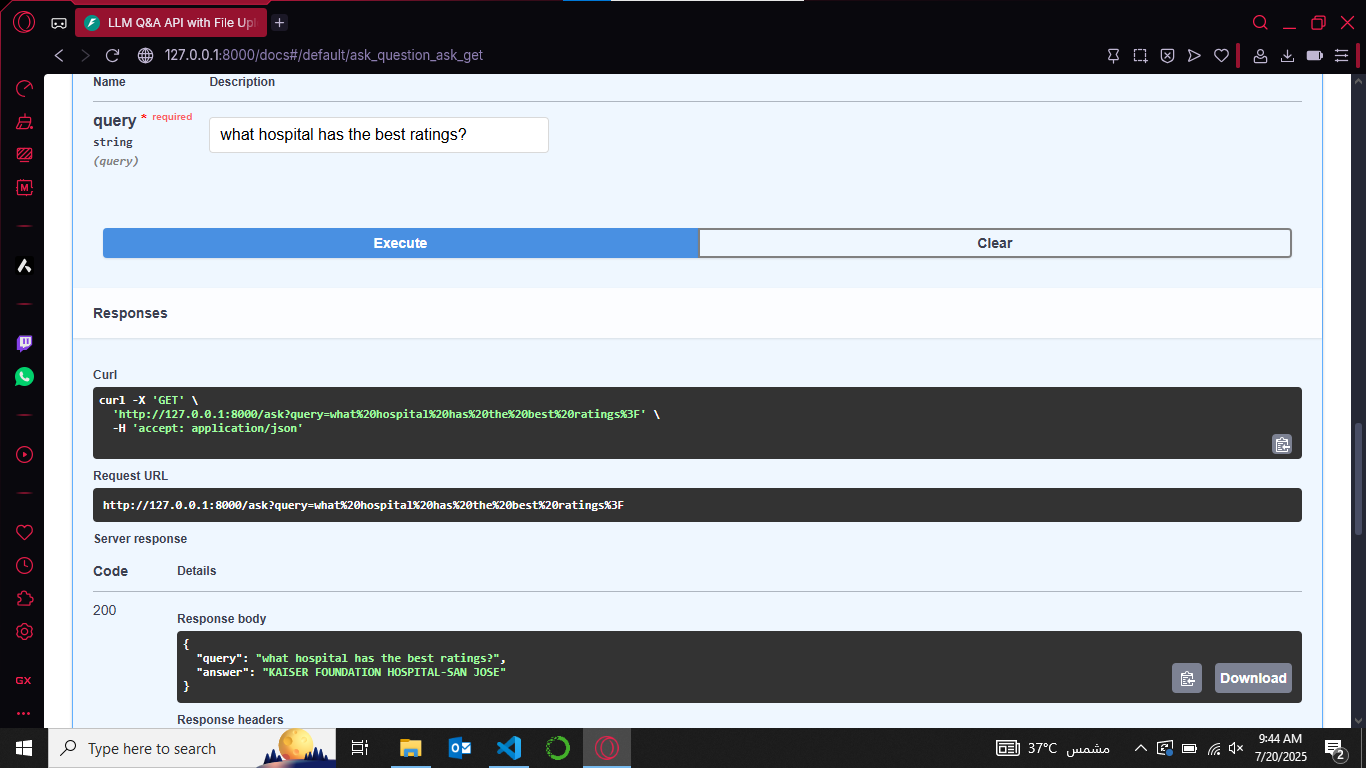
For Timeliness, the Greater Regional Medical Center has "Above the national average", while the Regional Medical Center and the Weeks Medical Center both have "Same as the national average".

Based on this data, the Greater Regional Medical Center has the best Timeliness and Patient Experience.

**4. Linking an LLM to FastAPI**

After all of that I used the Flan model and linked it to FastAPI. In the swagger ui page the users are able to upload their files. Will get a message indicating the number of entries received and vectorized. Then another field to input their queries. Below are some examples.





**References**

<https://stackoverflow.com/questions/78347994/what-is-the-easiest-way-of-converting-json-to-vector-database>

<https://medium.com/@ritik.kumar8146160/integrating-google-gemini-api-with-postgresql-for-intelligent-data-interaction-4cc696b4f8d8>

<https://console.groq.com/home>