Graph Rag LLM Research Consultant Project

James Hopham and Raja Solanki

July 12, 2024

The objective of this project was to implement emerging techniques for improving the performance of generative AI large language models (LLMs) under specific contexts. In particular, the application of graph database construction in natural language processing (NLP) to provide better Retrieval-Augmented Generation (RAG) performance allows for an improved likelihood of both avoiding model hallucinations and generating more accurate responses compared to basic LLM querying alone. In this project, I created a program that considers a database I compiled containing 58,067 biomedical articles/research papers to act as a consultant for professionals specializing in cancer treatment development within the biomedical industry who need to stay up-to-date on the most relevant and important research being conducted in the field. The program not only generates a response based on the user prompt, but additionally provides visual context that demonstrates the connections made by the model between different relationships within the database that was used to generate the answer provided to the user.

Introduction

Large language models (LLMs) developed through transformers allowed for a surge in development of tools such as Chat-GPT and Llama to allow for users to quickly query responses for various inputs including general questions, tasks, and text/image generation. This is made possible through advancements in data preprocessing techniques and notably through the introduction of the notable neural attention mechanism introduced by Vaswani et al. [2017] that distinguished transformer models from previously considered recurrent neural networks (RNNs) or convolutional neural networks (CNNs).

When querying LLMs, transformer-based models are used to encompass sequence-to-sequence learning tasks such as chatbots, question answering, and text summarization. Employing transformer encoders as opposed to RNNs provides superior performance in these tasks due to the capability of the former capturing richer relationships and patterns within the data to generate more complex answers when utilizing the database provided to the model under the mechanism of neural attention. When processing an input sequence using in natural language processing (NLP), feature engineering is carried out to generate standardized tokens which allows us to process the original text into units that can be encoded into a unique integer that is indexed as a database containing the mappings of words to their vector representations.

Embedding spaces using transformer-based models allows us to create richer context-aware representations compared to embedding models used in RNNs due to the self-attention feature. As opposed to the orthogonal representations that plague one-hot encoding or the fixed relationships within traditional embedding models, transformer encoders input context-aware representations into a sequence of embedding vectors. These vectorized representations can then be used to query the LLM for the respective data needed to produce a response to our input sequence.

Certain key issues have plagued LLMs including hallucinations, being trained on old data, and a lack of clarity as to how an LLM arrived at a given output sequence. To allow for more up-to-date information, retrieval-augmented generation (RAG) is used to compel the LLM to reference databases external to the model itself without retraining a new model. Another advantage is that the model will openly state if it is unable to identify relevant data to the user's input sequence rather than hallucinate an answer. Recent literature and advancements have increasingly focused on using an additional mechanism on top of RAG to better avoid hallucinations in the data, obtain insight on the representations identified by the LLM, and provide more accurate information. This new mechanism has been referred to as knowledge/property graphs or Graph RAG.

This mechanism relies on the construction of abstract graph representations of the data to prompt attention to key entities represented as nodes and relationships represented as edges within the external RAG database. Extractors can be used to construct one or more property graphs that can then be stored as a knowledge graph in which graph retrievers can be called to reference given regions respective to the input sequence of the user's query. This process allows us to not only output superior responses to basic RAG and LLM answers, but also provides us visual representations of the inner workings of the model's identification process to observe points of interest such as entities from particular documents that may warrant further analysis.

Methods

Professionals within various industries are required to maintain awareness of updates to advancements in areas such as research, methodologies, trending, and high-impact areas within their respective fields. The goal of this project was to implement and demonstrate the use of graph representations in RAG to serve the needs of a specialist within the biomedical industry who may need to consult for updated information on emerging research that they should be aware of.

I generated an external database to integrate into the model by compiling a database of 58,067 biomedical articles and research papers from 2000 to 2020 for my program to reference when processes input query sequences. Due to resource limitations, I took a random sample of the documents to decrease the computational load. This dataset would allow a working

professional to have accurate and updated information up to the point of the compiled dataset, rather than solely relying on the time horizon in which the LLM was trained.

To generate our knowledge graphs, I used the LLamaIndex data framework in conjunction with the Mistral AI LLM. The use of external tools within this methodology inherently leads to a preference of LLMs that provide support for native function calling such as Google's Gemini or OpenAI's GPT4-o, leading to our decision to use Mistral AI's LLM rather than one such as LLama 3. I also call the Mistral AI 1,024 dimension embedding model to be able to have a pre-trained model for providing context-aware vectorized representations of text data available when querying our input sequences.

Using LLamaIndex, we call an extractor to generate our property graphs. As opposed to SimpleLLMPathExtractor, SchemaLLMPathExtractor uses prompt engineering to allow us to define how the output should look and provide a parsing function that extracts the output from an LLM to create a knowledge graph by defining which node labels and relationships we want to extract. When indexing the constructed property graphs into a knowledge graph, we can additionally identify if multiple nodes in the knowledge graph reference the same entity and merge them into a single node to have better structural integrity such as through a combination of text embeddings and word distance heuristics to find potential candidates.

Based on the user input, property graph retrievers can be called to retrieve respective information from the knowledge graph, then pass the relevant data to an LLM where a final answer can be obtained. Various retrievers can be used such as LLMSynonymRetriever for customized exact keyword match node searches, or Text2Cypher for constructing cypher statements to ask questions such as how many nodes are in the graph. For this project, I provide a demonstration of VectorContextRetriever which is more robust and less reliant on exact keyword match than LLMSynonymRetriver.

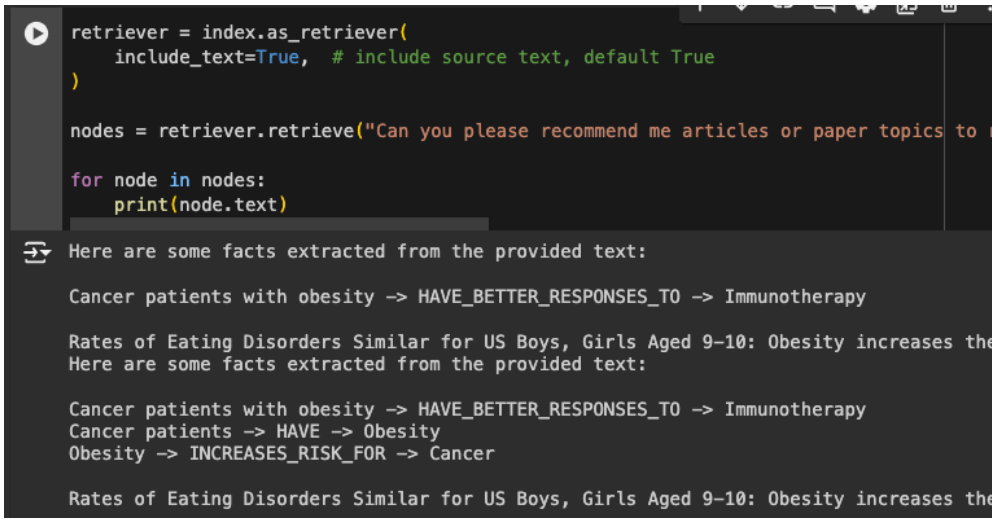
VectorContextRetriever takes the whole string, embeds it, and then finds the relevant nodes. It always returns at least a few results from the database by allowing you to choose to take the top n in hopes of identifying relevant nodes. We then return the direct neighborhood of relevant nodes found using vector search. The downside of this retriever is that if multiple entities are mentioned in the text, it might embed both into a single embedding.

The final query engine I created now implements the entire process and provides an output sequence containing the visual representation of the nodes and relationships identified by the model within the external database reference through RAG to understand the model's observations. I additionally provide an example of the outputted text response as well as an example of the output in a scenario where the question being asked in the user input sequence

does not contain any corresponding data within the RAG external database to identify relevant answers to show how the model avoids hallucinations.

Results

Figure 1. Node Generation

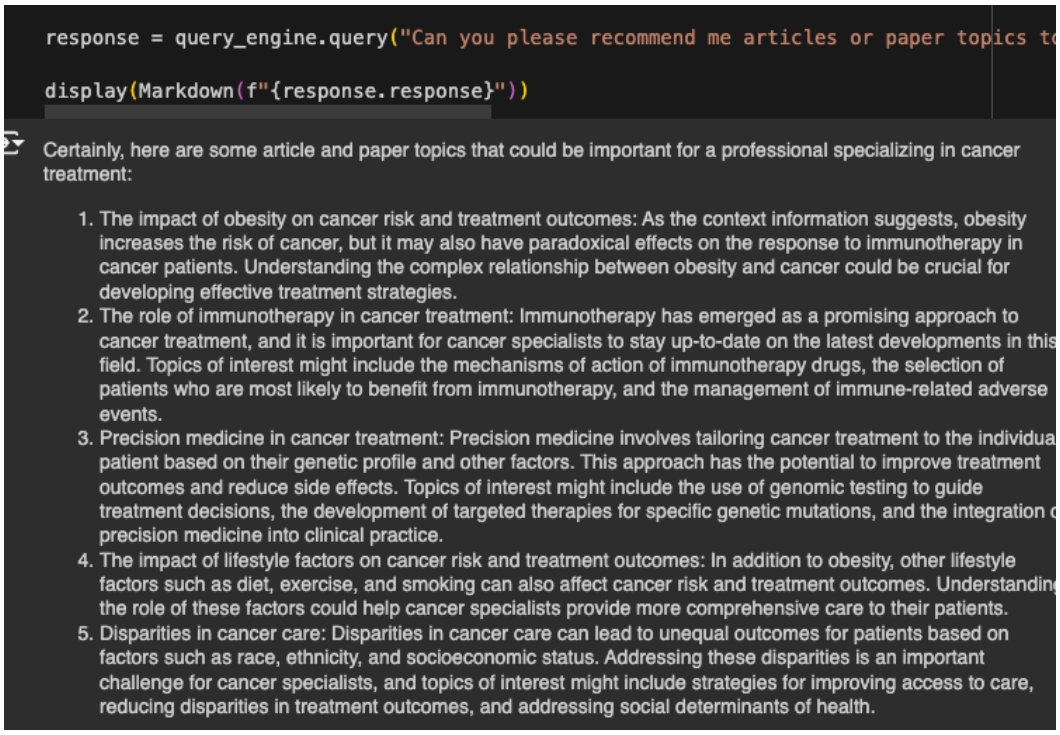


```
retriever = index.as_retriever(
    include_text=True,  # include source text, default True
)

nodes = retriever.retrieve("Can you please recommend me articles or paper topics to

for node in nodes:
    print(node.text)
```

```
Here are some facts extracted from the provided text:

Cancer patients with obesity -> HAVE_BETTER_RESPONSES_TO -> Immunotherapy

Rates of Eating Disorders Similar for US Boys, Girls Aged 9-10: Obesity increases the
Here are some facts extracted from the provided text:

Cancer patients with obesity -> HAVE_BETTER_RESPONSES_TO -> Immunotherapy
Cancer patients -> HAVE -> Obesity
Obesity -> INCREASES_RISK_FOR -> Cancer

Rates of Eating Disorders Similar for US Boys, Girls Aged 9-10: Obesity increases the
```

Query: *"Can you please recommend to me articles or paper topics to read that would be important for a professional specializing in cancer treatment to be aware of?"*

Figure 2. Outputted Response

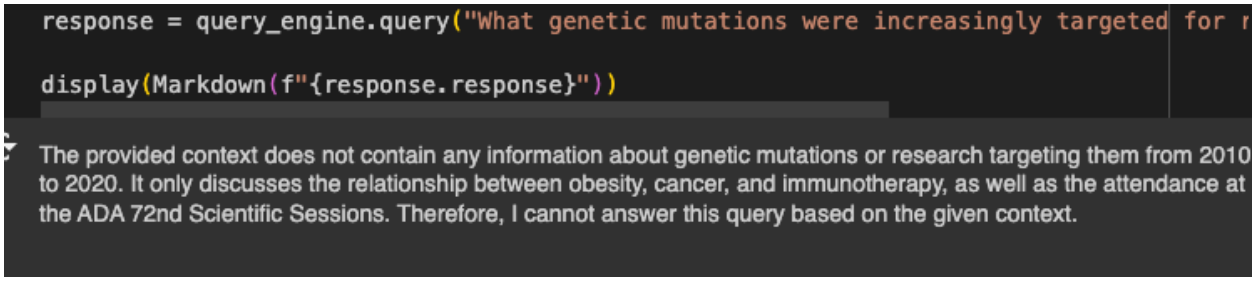


```
response = query_engine.query("Can you please recommend me articles or paper topics t

display(Markdown(f"{response.response}"))
```

Certainly, here are some article and paper topics that could be important for a professional specializing in cancer treatment:

1. The impact of obesity on cancer risk and treatment outcomes: As the context information suggests, obesity increases the risk of cancer, but it may also have paradoxical effects on the response to immunotherapy in cancer patients. Understanding the complex relationship between obesity and cancer could be crucial for developing effective treatment strategies.
2. The role of immunotherapy in cancer treatment: Immunotherapy has emerged as a promising approach to cancer treatment, and it is important for cancer specialists to stay up-to-date on the latest developments in this field. Topics of interest might include the mechanisms of action of immunotherapy drugs, the selection of patients who are most likely to benefit from immunotherapy, and the management of immune-related adverse events.
3. Precision medicine in cancer treatment: Precision medicine involves tailoring cancer treatment to the individua patient based on their genetic profile and other factors. This approach has the potential to improve treatment outcomes and reduce side effects. Topics of interest might include the use of genomic testing to guide treatment decisions, the development of targeted therapies for specific genetic mutations, and the integration precision medicine into clinical practice.
4. The impact of lifestyle factors on cancer risk and treatment outcomes: In addition to obesity, other lifestyle factors such as diet, exercise, and smoking can also affect cancer risk and treatment outcomes. Understandin the role of these factors could help cancer specialists provide more comprehensive care to their patients.
5. Disparities in cancer care: Disparities in cancer care can lead to unequal outcomes for patients based on factors such as race, ethnicity, and socioeconomic status. Addressing these disparities is an important challenge for cancer specialists, and topics of interest might include strategies for improving access to care, reducing disparities in treatment outcomes, and addressing social determinants of health.

Figure 3. No Relevant Context Indication



Query: *"What genetic mutations were increasingly targeted for research from 2010 to 2020?"*

Conclusion

The program appears to generate accurate results and successfully identifies relevant contextual relationships within the external RAG database for the presented consultation within the biomedical industry space.