Retrieval Augmented Generation

by jai-llm
14-Sep-2023

1.0 TL;DR

Retrieval **A**ugmented **G**eneration (RAG) refers to an approach to provide contextual information to Large Language Models (LLMs). The purpose of this contextual information is to elicit better responses from the LLM and reduce hallucinations. One side benefit of RAG is that it can increase trust by providing the source of the LLM responses. In this project we answer ML questions using LLM and RAG to show the benefits of RAG. Key Takeaways are:

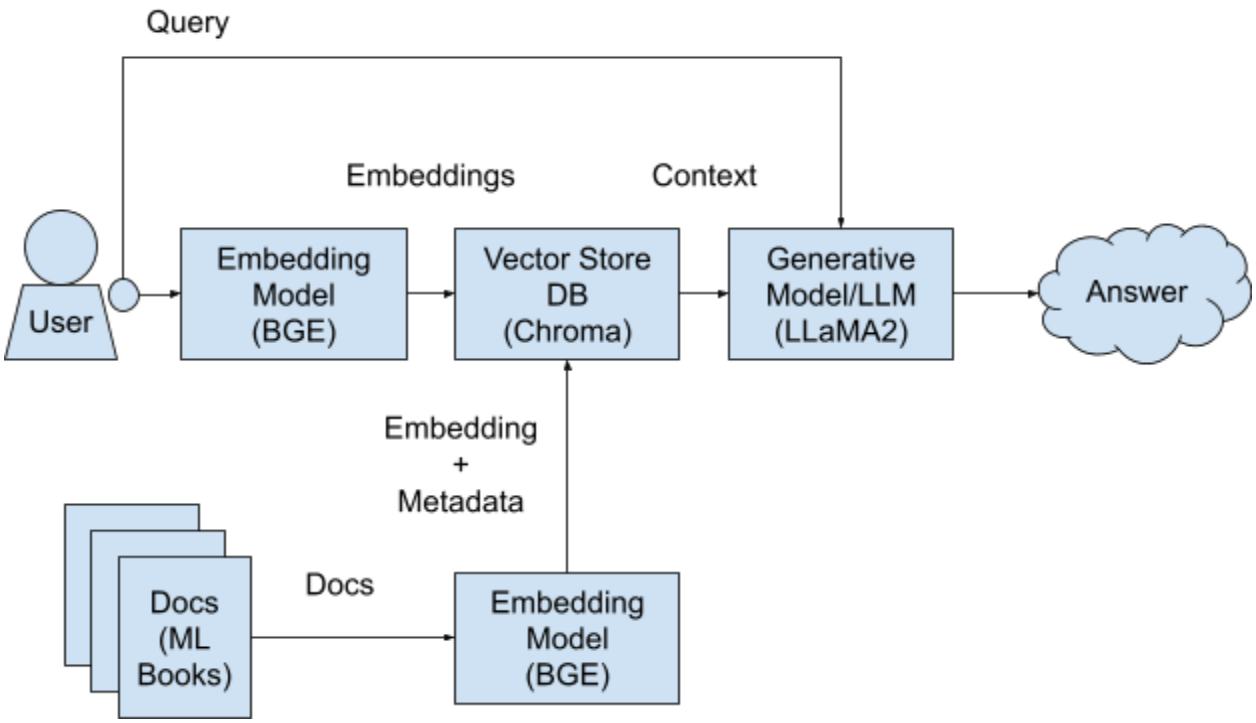
1. Even small quantized Open Source LLMs such as the LLaMA-7B-GPTQ model are really good at Question Answering tasks.
2. Inference with our model is fast even on a T4 GPU.
3. Benefits of RAG include:
    - Reducing "hallucinations" by providing better context.
    - Increasing trust by providing the response sources (White-boxing LLMs).
4. RAG + Prompting gives the best responses to our ML questions.

2.0 RAG Overview

LLMs have revolutionized the NLP/NLU space over the last year. However, LLMs have several drawbacks:

1. **Static** - LLMs know what is in their training data so events that occured post-training are not known to the LLM.
2. **Lack Domain Knowledge** - LLMs are trained to perform well on multiple tasks but do not know the specific problem context.
3. **Black Boxes** - It is not easy to know the sources the LLM considered when it arrived at its response. This makes it hard to trust LLMs.
4. **Training is Costly** - Building LLM and fine-tuning them requires specialized skills and is costly in terms of compute time.

RAG addresses the above issues by providing contextual information that helps bring the LLM up to date by providing context, white-boxes LLM by providing the source of their information, and makes it easy with scripts for anyone to tune LLM responses to their custom tasks. The end result is a reduction in “hallucinations” due to better context and an increase in trust and transparency with LLM models, provides the source of information used to arrive at the LLM response.



RAG: Conceptual Overview

Additional Comments:

- **VectorDB** - Is used to do semantic search (based on user query intent rather than text matching) to return the relevant chunks of text. In our example on Linear Regression the ISLP text contains a chapter on linear regression so the Top-2 results or text chunks are from this chapter of the book.
- **LLM** - Can answer questions related to ML but sometimes it "hallucinates". For example when asked about trees it responds about trees in general rather than decision trees which is what we intended.
- **RAG** - Retrieves Top-5 chunks from Vector store (context) and provides it to the LLM in addition to the query. This improves the LLM results for the ML questions by providing additional context and also reduces LLM "hallucinations". For example, asking RAG about trees results in LLM not giving a response since it does not know if we meant a "decision" tree or a "boosted" tree.
- **RAG + Prompting** - We not only use a vector store but also prompt the LLM to provide better response. This approach seems to give the best results. For example, when asked about trees the LLM gives a succinct response about “decision” trees instead of simply refusing to answer the question.

3.0 Conclusion

RAG can help improve LLM responses by reducing “hallucinations” and white-boxing LLM models. Using RAG in conjunction with better prompting gives the best results in this project.