RAG System for HDFC Insurance Policy Documents

In this case study, we developed a RAG framework to retrieve information from seven types of HDFC insurance policy documents using LlamaIndex.

The RAG system involves two main components: an information retriever and an LLM generator. Once a vector database is constructed using the chunked documents, the most semantically similar documents to the query are retrieved. These documents along with the query are fed to the LLM generator to produce an appropriate response. The system architecture is discussed below in detail.

System Architecture

- We employed LlamaIndex as it allows us to read files of various formats with ease to create a vector database.
- SimpleDirectoryReader and PDFReader were utilized to read the insurance PDF documents.
- The documents were parsed using SimpleNodeParser and nodes were created out of them.
- An index was created using the VectorStoreIndex of LlamaIndex.
- Finally, a query engine was constructed using the constructed index which could respond well to the user queries.
- These components were then connected by defining a query_response and initialize_conv function which allows the user to ask the model questions in chat-type setting about the insurance policies. The user is also asked to rate each response of the model.
- A pandas dataframe was constructed which included the following columns: 'Question', 'Response', 'Page', and 'Review'.

Improvements

After constructing the basic pipeline, we tried out modifications to improve the query responses:

- We tried a custom prompt setup OpenAI's gpt-3.5 turbo model and Microsoft's phi-3 mini 4k LLM model from HuggingFace to check how the responses are enhanced.
- We further utilized "all-mpnet-base-v2" and "bge-small-en-v1.5" embedding models.
- Finally, we also employed a sub-question query structure for a faster and smoother querying process.

Check this document along with the "HelpMateAI_HDFC.ipynb" notebook for easy implementation.