HELP MATE AI FOR INSURANCE POLICY DOCUMENT GENERATIVE SEARCH

Objective

Building a Help Mate RAG system.

Problem Statement - The goal of the project will be to build a robust generative search system capable of effectively and accurately answering questions from various insurance policy documents. The project will build a RAG system using LlamaIndex.

Why LlamaIndex and not LangChain?

LlamaIndex is preferred, because:

- LlamaIndex's is good in data ingestion and retrieval. It is apt in document heavy use-cases.
- LlamaIndex's core strength is on use-cases such as getting answers from documents.
- LlamaIndex simplifies the process of Q&A
- LangChain is more suitable in cases of LLM Orchestration

Design

The below diagrams outlines the overall system design and the various layers of the project.

RAG High Level Architecture

Retrieval Augmented Generation (RAG)

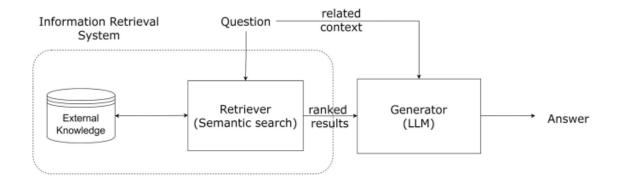


Figure 1: RAG High Level Architecture

Basic RAG pipeline in LlamaIndex

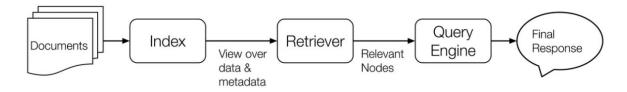
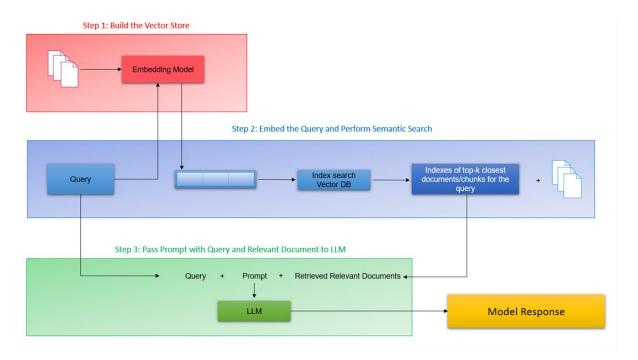


Figure 2: Basic RAG pipeline in LlamaIndex

RAG: Layers and Steps



RAG: Layers and Steps

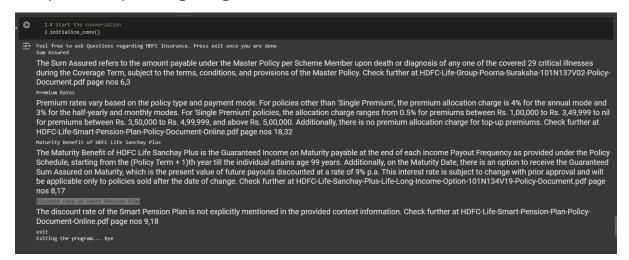
Implementation

The implementation details is covered in the jupyter notebook attached with this file. To run it,

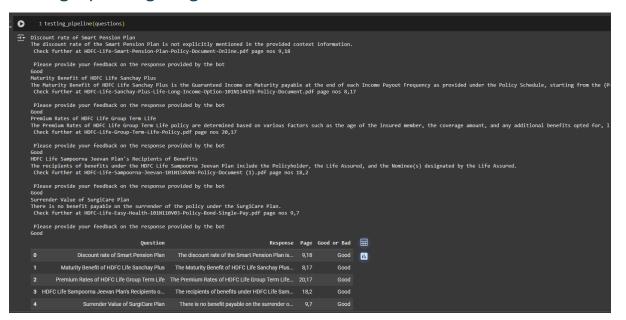
- Please use Google Colab.
- Please access the insurance documents mentioned in the jupyter notebook and upload them to your google drive,
- Ensure the chatgpt keys are also made available

Results

Response Pipeline gave good results:



Testing Pipeline gave good results:



LLM/Chatgpt gave good results:

Output1:

Output2 (Negative Test gave right results):

Challenges and/or Lessons Learnt

- 1) SimpleDirectoryReader is sufficient if the number of PDF documents to be read is less
- 2) Do negative test to check the effectiveness of your query. E.g. Ask a question whose response is not available in the document.
- 3) Conduct tests to check the effectiveness or getting the data from the Vector DB i.e. basic retriever is working, & retriever streaming is working
- 4) While testing the query engine try out various options to build confidence in your own code.
- 5) Ensure the Testing Pipeline output is scored.
- 6) Even when using prompts, use negative testing.