Leveraging Retrieval-Augmented Generation for Financial Advisory

KWONG Cheuk Lam
Advised by Prof. WU Daoyuan

Overview

The emergence of GPT-based Large Language Models has revolutionized the technology and finance sectors, enabling tasks such as summarization, chat services, and data retrieval to be performed with high precision and low latency. However, a notable limitation of LLMs is their inability to access information beyond their training data. To address this, Retrieval-Augmented Generation (RAG) can be employed, which involves intercepting prompts and incorporating real-time information as context. Magnum Research, a leading financial SaaS provider in Hong Kong, has extensively developed RAG with LLMs as a novel format for data provision and service delivery. In this project, various RAG tools have been implemented and integrated into Magnum's AI service.

YOU: What is the current AUM of JPMorgan Global Bond Fund?

AI: I'm sorry, but I don't have real-time data access to provide the current Assets Under Management (AUM) for the JPMorgan Global Bond Fund. You can check the latest AUM on financial news websites or the official JPMorgan Asset Management website.

⇩ RAG Integration

What is the current AUM of JPMorgan Global Bond Fund?

✓ Thinking process: Completed

The current Assets Under Management (AUM) for the JPMorgan Global Bond Fund is approximately USD 2,337,461,787.71.

Objectives

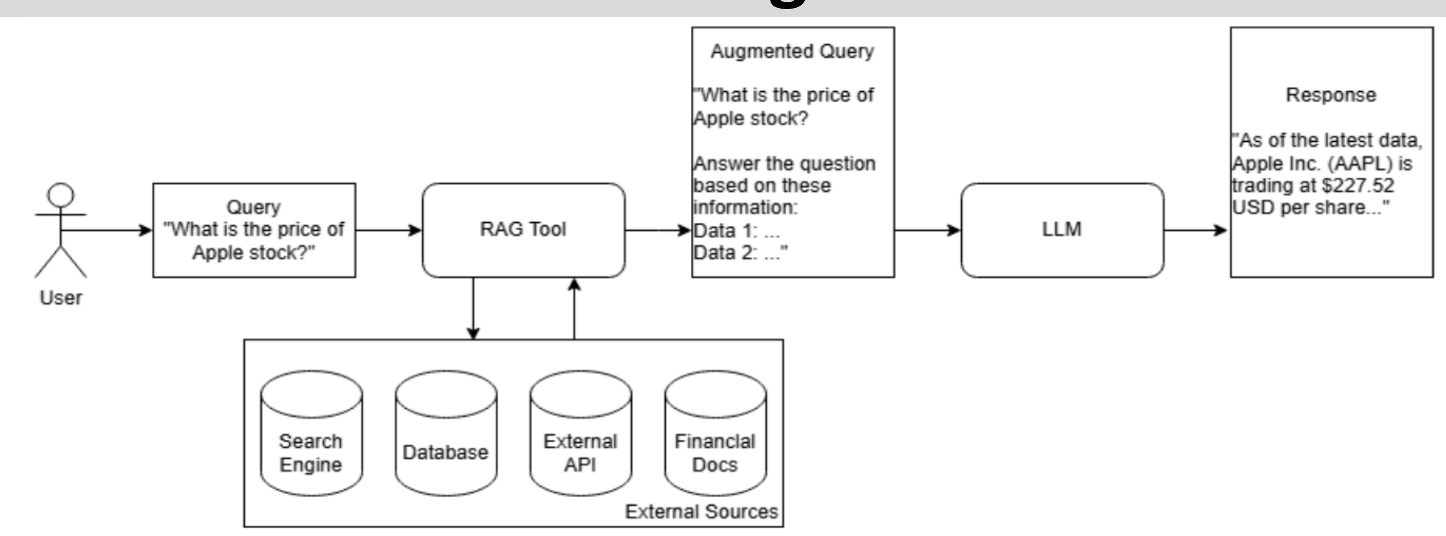
The primary objective of the project is to implement and integrate the effective RAG tools into the company's AI applications. To achieve this, three goals have been identified:

✓ Design, implement and benchmark RAG tools that align with the company's use cases, leveraging external financial documents, real-time data, news and internal data sources.

✓ Fine-tune and optimize the model and prompts to enhance the accuracy and relevance of data retrieval and generation of responses by the LLM.

✓ Facilitate users' investment decisions with user-friendly representation of the retrieved data and LLM responses.

Design

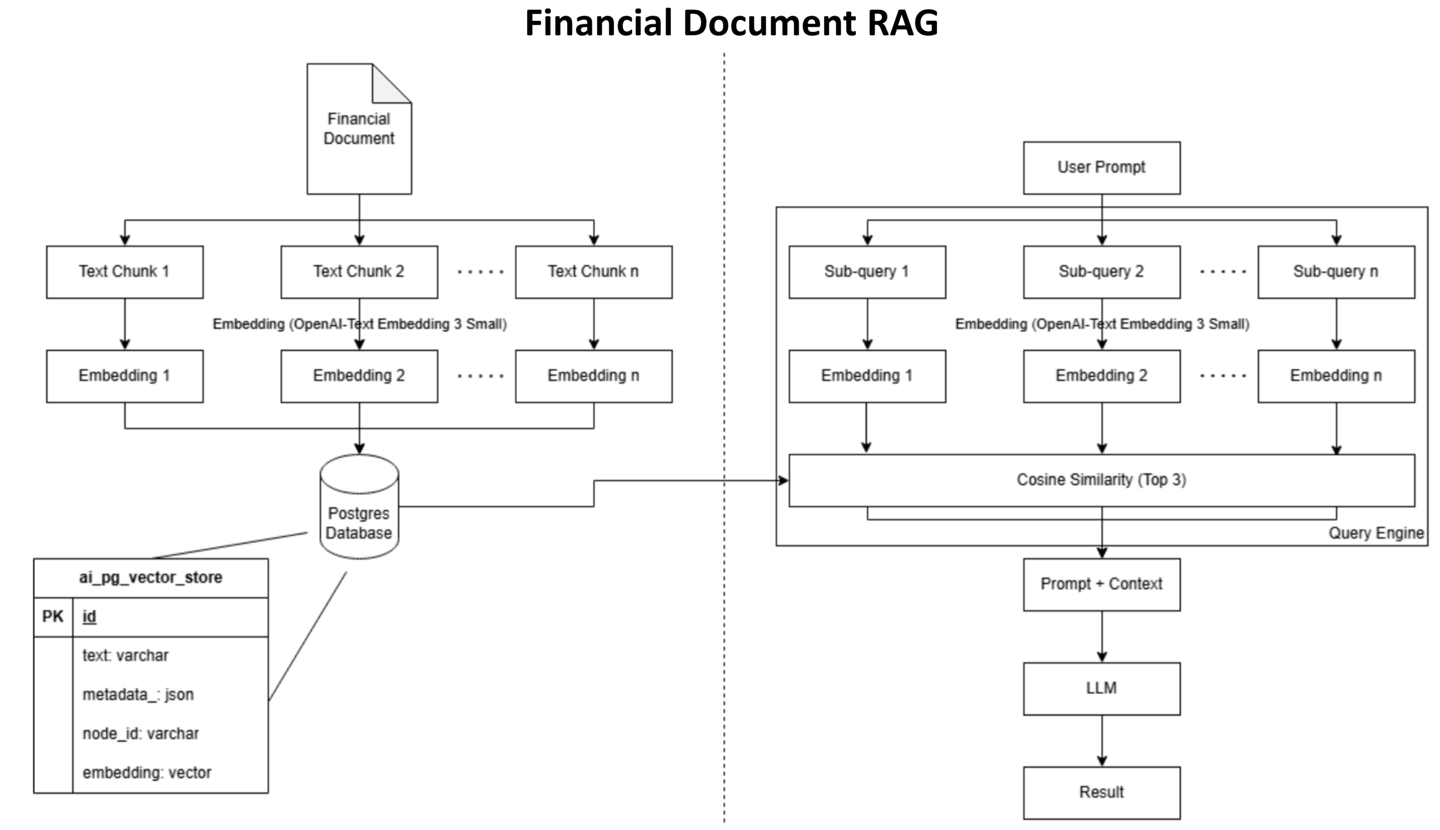


Different RAG tools employ various methods to retrieve relevant information and append to the user query, with similarity search being a common approach.

Implementation

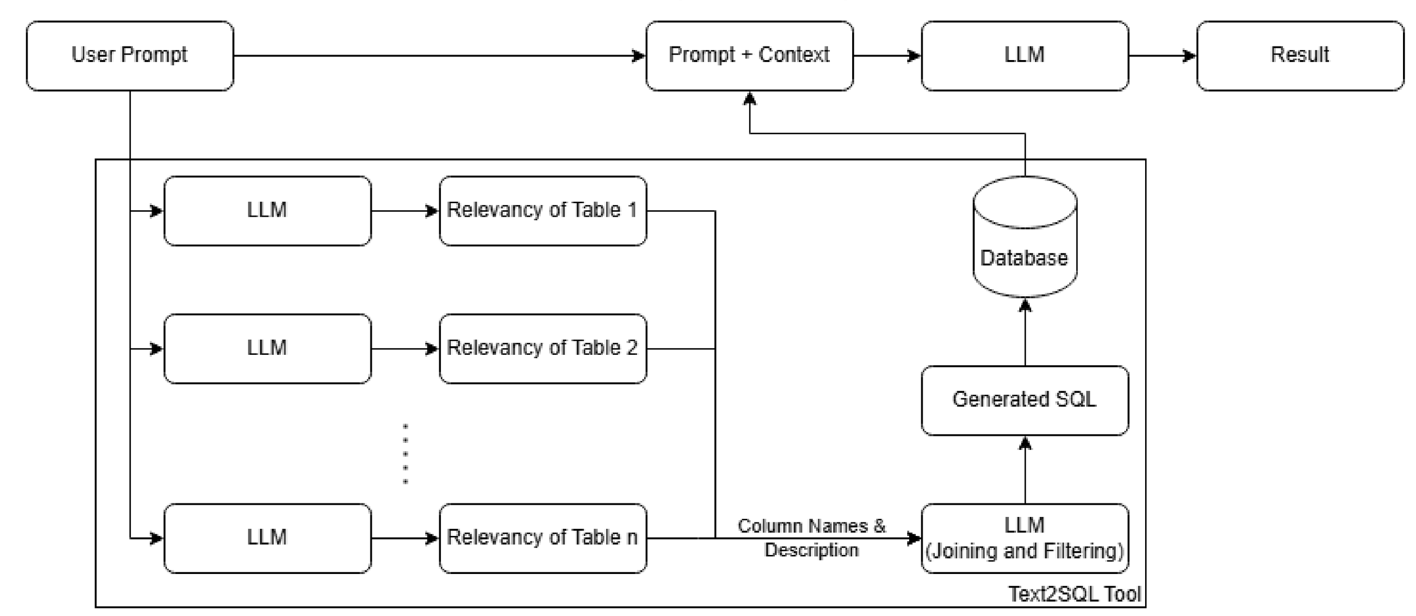
LlamaIndex is employed as the basic framework to integrate RAG tools with external data sources. Two examples of the RAG tools implemented are given.

Financial Document RAG



The process begins with indexing, vectorizing document text via **OpenAI's text-embedding-ada-002** model, which generates contextualized embeddings and stored in a **PostgreSQL** database. Upon receiving a user query, the RAG system vectorizes the query using the same model and **cosine similarity** identifies the top three similar text chunks, which are synthesized with the query and analyzed by the LLM.

Text2SQL (Database) RAG



The tool was developed to **convert text queries into SQL queries**, enabling data access as context for the LLM. Initially, the LLM assesses table relevance in the database based on user queries and schema. Relevant tables and columns are joined and filtered by a subsequent LLM stage. Finally, the LLM constructs and executes the SQL query against the database to retrieve the necessary information. The parallel and sequential LLM processes are implemented using **DiFy workflows**.

Evaluation

To systematically evaluate the performance of RAG tools, a custom evaluator and the RAGAS package are employed. These tools utilize LLM to assess the accuracy of generated responses against predefined standard answers. Two statistics are shown: the first assesses financial document RAG using the RAGAS package, while the second evaluates Text2SQL RAG with a custom evaluator.

	Mean	Median
Precision	52.48%	50%
Faithfulness	82.83%	100%
Relevancy	81.72%	98.24%
Recall	78.19%	100%
Manual Rating	88.54%	100%

	Mean	Median
Accuracy	92.96%	100%
Relevancy	71.81%	100%
Recall	92.66%	100%
Manual Rating	88.96%	100%

The tests show an adequate accuracy of 85-90% on fund-related questions, with over 87.3% of queries fully answered. Despite this performance, instances of incorrect or incomplete answers persist, indicating room for optimization and fine-tuning of the tools to enhance accuracy and reliability.