Architecture of Multi-hops Retrieval Augmented System

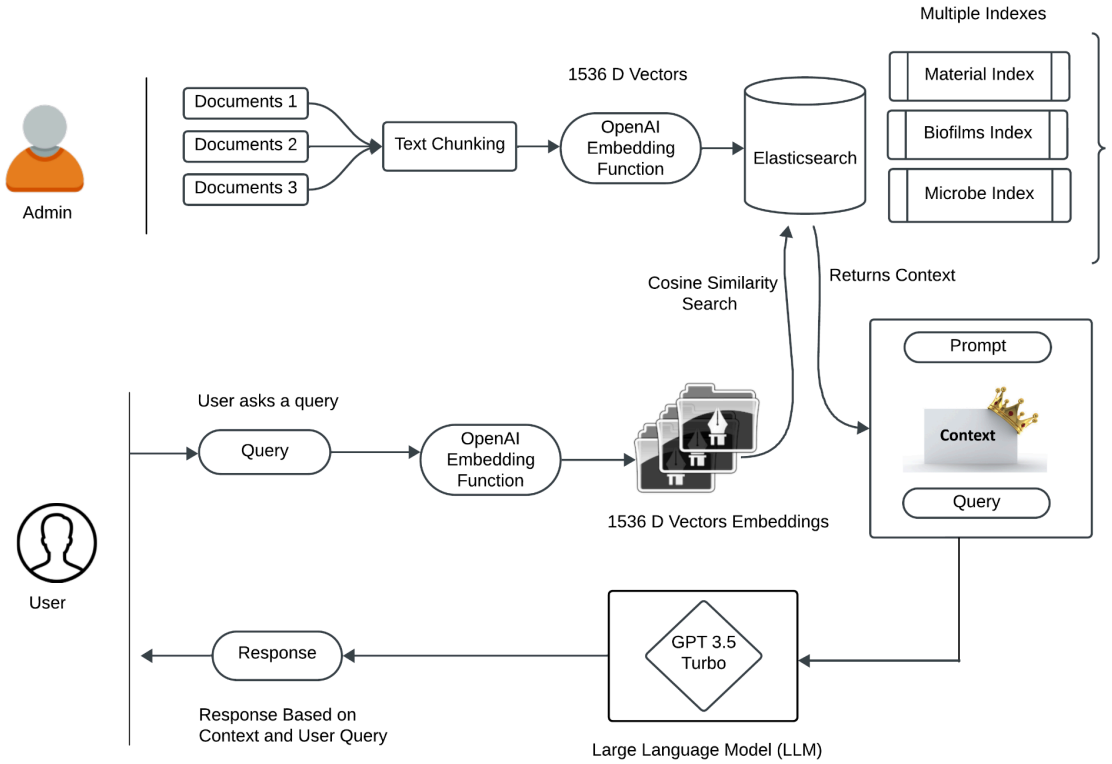

Figure 1.0 : Architecture of RAG with Multi-Hops Query

In the multi-hop RAG system, we have various components, including a vector database, a Large Language Model (LLM), and Embedding Generators, also known as sentence transformers. Additionally, there are two user roles: system admin and a regular user.

Examining the architecture, the system admin performs indexing on multiple documents related to domains such as materials, biofilms, and microbes. These documents are then segmented into chunks of text, typically of size 1024 bytes. These chunks undergo further processing through the embedding function. In this case, the OpenAI Embedding function is employed, generating vectors of size 1536 dimensions. These vectors are stored in Elasticsearch based on their respective indices. For instance, if a document is related to the material domain, it is stored in the material index of Elasticsearch along with the relevant document metadata.

On the user side, a query is sent to the Multi-hop RAG system, and the query is processed using the same OpenAI embedding function, resulting in a 1536-dimensional vector. These vectors are then used in a cosine similarity search, filtering for contexts with a score of >= 0.5 in the Elasticsearch indices. The similar documents obtained are combined to create a context

for the GPT 3.5 Turbo LLM. This context, along with the user's original query, is then fused to form a comprehensive prompt. This prompt is fed into the GPT 3.5 Turbo, which processes it and provides relevant information to the user. This approach shows promising potential in mitigating LLM hallucinations and enhancing response quality.