Query Data

> This guide is part of a learning series about Retrieval-Augmented Generation (RAG). For context, see the Introduction to RAG.

You have a meeting with a well-known expert in ten minutes. You need to ask them an important question, and you don't want to waste their time or appear foolish. Before the meeting, you do some quick research on the question's subject matter to bring yourself up to speed, and to also provide some important context to the expert, like:

How has this question been answered before?

What related information might impact the answer?

Do you want them to provide a concise answer with some supporting docs, or a highly detailed, all-day speech?

Your quick pre-question research is similar to the RAG query step. Like time with an expert, LLM calls can be expensive, so providing accurate context with an effective prompt will lower costs and improve performance.

Querying is the process of searching through the indexed, embedded data for relevant information before submitting your prompt to the LLM. During querying, a query (often in the form of a question or a statement needing more information) is issued against the indexed database. The system then uses the query to retrieve the most relevant documents or information from the index. The querying process often involves comparing the vector representation of the query to the vectors in the index to find the best matches.

The output is a set of documents or snippets from the indexed database that are most relevant to the query. These results can then be used to augment a language model's responses, providing additional context or information that the model might not have had access to otherwise.

Improve querying performance

- **Metadata Filtering**

Metadata filtering gives you a way to narrow down the pieces of content by first filtering the documents and then applying the nearest neighbor algorithm. In cases where you're dealing with a large number of possible matches, this initial pre-filtering can help you narrow the possible options before retrieving the nearest neighbors.

- **Prompt Refinement**

Craft explicit, specific prompts aligned with the intended output. Key considerations include:

- Defining the LLM's role: Direct the LLM to act in a specific way, such as a lawyer or a customer service agent, to tailor its responses.
- Using the provided context: Instruct the LLM to incorporate the given context into its responses.
- Incorporating examples: Provide examples relevant to the prompt, like contract scenarios, to guide the LLM's understanding and response.
- Specifying output format: Dictate the desired format of the LLM's responses, particularly for specific information requirements.
- Applying chain of thought: For reasoning-heavy tasks, guide the LLM through a logical sequence of steps to reach a conclusion. These techniques improve the accuracy and

relevance of the LLM's responses, although sometimes multiple interactions with the LLM might be necessary to obtain optimal results.

What's next?

Now that you understand the querying process, it's time to measure and evaluate your overall RAG process. Continue on to evaluating.