# Retrieval Augmented Generation

can code communities

# Review Take Home

Use one (or more) of the simple to use tools that LangChain provides to create an agent.

- AlphaVantage
- Brave Search
- Dall-E Image Generator
- OpenWeatherMap
- SerpAPI
- WolframAlpha

# RAG

Retrieval Augmented Generation

- Another method of connecting a large language model with an external datasource.
- Usually focused on private or unorganized data.
  - Over 95% of the world's data is private (personal, company, etc.)
- RAG offers a way to supply the model with the helpful information that pertains to the prompt from the datasource within the context.

# 3 Steps of RAG

1. Indexing - formatting the database so specific data can be retrieved based on a natural language prompt.
2. Retrieval - finding the relevant data in the database using a function so that it can be passed to the model through the context.
3. Generation - taking the retrieved data and the initial user prompt and prompting the model to get a factual response.

# Indexing

Indexing is the act of taking the external documents and loading them into a database for the retrieval function to be applied to. We index a document by converting it into a numerical representation.

# Embeddings

An embedding is data indexed to a vector of floating point numbers.

- Embeddings also have context limits.
  - To get embeddings that properly reflect the data it represents, longer documents will be split into chunks.
- The user prompt also gets converted into an embedding for the retrieval function to find the prompt's closest comparison within the dataset.

# Vector Database

- Embeddings are stored in a vector database.
  - Allows for fast and accurate similarity search and retrieval of data.
  - Instead of querying databases based on exact matches or predefined criteria, you can find the most similar or relevant data based on their semantic or contextual meaning.
- Chromadb is an open source vector database that we can use locally.
- Pinecone and MongoDB that use cloud storage to better manage and scale the databases being used.

# Retrieval

- Cosine similarity is the distance between two vectors in a multidimensional space.
  - This represents their relatedness.
- K-nearest neighbor search (kNN search) is one of the most common retrieval functions.
  - It finds the lowest cosine similarity values between the prompt embeddings and the dataset embeddings.
  - K represents the number of nearest neighbors that will be retrieved.
    - This increases the probability that the information needed to properly answer the question in the prompt is passed into the context.

# Generation

Once we have retrieved the data that is most likely to help answer the user prompt, we prompt the model with a prompt template that combines the data and user prompt.

Answer the following question based on this context: {context}
Question: {question}

# Advanced RAG Pipeline

- RAG pipeline (indexing, retrieval, and generation),
- There are many methods that can be implemented in each of these core steps to improve the system.
- These advancements are more helpful for certain use cases.
    - When building more complex RAG pipelines, we should always consider the data, the database system, and the desired output format.

# Query Translation

- The first stage of advancing a RAG pipeline.
- The process of converting the user prompt into another prompt that is designed to improve the accuracy of retrieval.
- User prompts can range in quality heavily, so query translation creates consistent prompts to be used by the retrieval function.

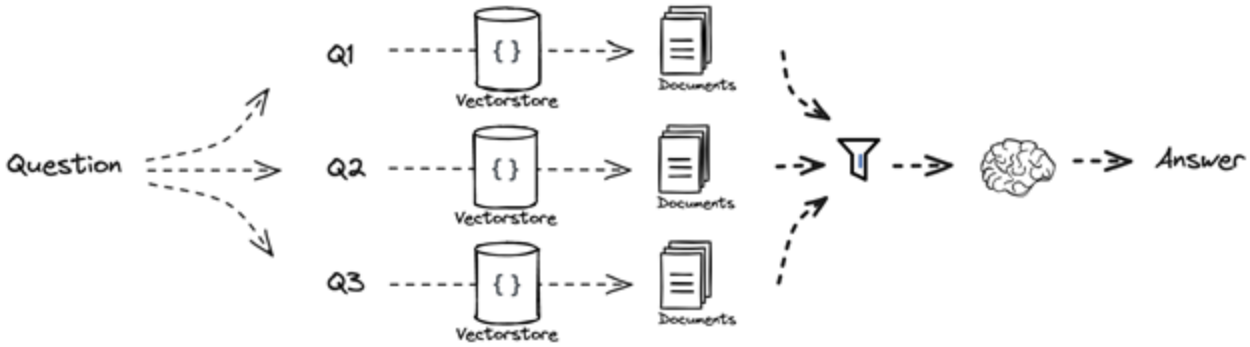
# Multi Query

- Creates multiple prompts from different perspectives to increase the likelihood of reliable retrieval.
- These prompts can retrieve their own set of documents and then be combined in the context window for generation.

You are an AI language model assistant. Your task is to generate five different versions of the given user question to retrieve relevant documents from a vector database. By generating multiple perspectives on the user question, your goal is to help the user overcome some of the limitations of the distance-based similarity search. Provide these alternative questions separated by newlines. Original question: {question}

# RAG Fusion

- Multi Query increases the number of documents that are passed into the generation context, which in some cases, can overload the context.
- To ensure the context limit isn't being reached, RAG fusion can be used
  - Rank the list of retrieved documents by relevance to the prompt.
  - Then we only pass a certain number of the top ranked documents to the context instead of the whole list.



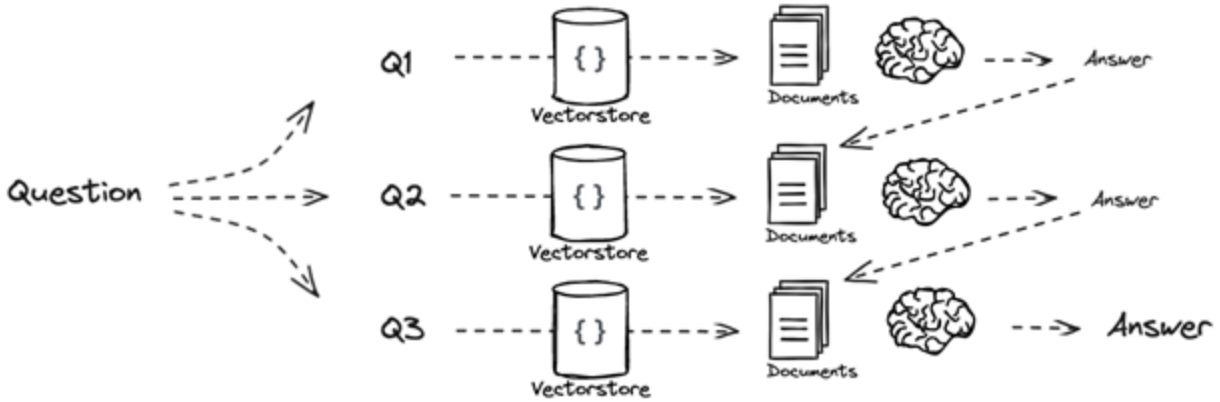
# Decomposition

- Breaks down the user prompt into a list of sub questions to then be used for retrieval.
- Similar to chain of thought prompting, but different because we mainly use decomposition for retrieval and not generation.

You are a helpful assistant that generates multiple sub-questions related to an input question. \n The goal is to break down the input into a set of subproblems / sub-questions that can be answered in isolation. \n Generate multiple search queries related to: {question} \n Output (3 queries)

# IR-CoT

IR-CoT (Interleave Retrieval with Chain of Thought prompting) is a method where we generate an answer for each sub question with the documents it retrieved and the generation response from the earlier sub questions.



# Step-back Prompting

- Essentially the opposite of decomposition.
- Creates a higher abstraction of the prompt to make sure retrieved documents are focused on the overall topic instead of the finer details.
- The step-back prompt and the initial user prompt are then both used for retrieval which pass both sets of documents into the context for generation.

You are an expert at world knowledge. Your task is to step back and paraphrase a question to a more generic step-back question, which is easier to answer. Here are a few examples:

```
{
    "input": "Could the members of The Police perform lawful arrests?",
    "output": "what can the members of The Police do?",
},
{
    "input": "Jan Sindel's was born in what country?",
    "output": "what is Jan Sindel's personal history?",
}
```

# HyDE

- Hypothetical Document Embedding (HyDE)
- Addresses the difference in embedding sizes between the prompt and the documents within the database.
- A hypothetical document is generated based on the prompt to then be used in retrieval.

Please write a scientific paper passage to answer the question. Question: {question} Passage:

# Take Home Exercise

Create a RAG chain using LangChain and connect at least three documents within the same topic. We are using beautifulsoup to get the document from the URL provided. There are some sites that block this, and there are a couple ways of using beautifulsoup to load documents (check if the documents are loaded properly first). Test the chain with some prompts asking for information you know can be retrieved in the documents to test.