RAG in Depth

Phases of a RAG App

Intro: RAG with LangChain

- What is RAG?
- LangChain and RAG.
- Our focus now: RAG for unstructured data.
- RAG Architecture.
- Steps of the Indexing Phase.
- Steps of the Retrieval and Generation Phase.
- Resources available from LangChain.

RAG Phases: LangChain Quickstart

- Build a basic RAG app with LangChain and trace it with LangSmith.
- Virtualenv.
- Dependencies.
- Necessary modules.
- .env file
- LangSmith activation.
- LangSmith project name: RAGquickstart
- Goal of the RAG app: QA over a blog post.
- Required modules.

Phase 1: Indexing

- Step 1: Loading data.
  - WebBaseLoader.
  - BS4.
  - Additional info on Loading data.

- Step 2: Split data into small chunks.
  - RecursiveCharacterTextSplitter.
  - Additional info on Splitting chunks of text.

- Step 3: Transform chunks into embeddings and store them in the vector DB.
  - OpenAI Embeddings.
  - Chroma DB.
  - Additional info on Embeddings and Vector DBs.

Phase 2: Retrieval

- Set the retriever.
- Additional info on Retrievers.

Phase 3: Generation

- Determine Foundation LLM Model.
- Set the Prompt.
  - Pre-defined prompt from LangSmith Hub vs. custom prompt.

- Pre-processing function to better format the document.
- Define the RAG chain.
  - Runnables, RunnableParallel, RunnablePassThrough.

- Start asking questions to the RAG app.
  - Streaming.
  - Alternative approach with customized prompt and no streaming.

- Additional info on Chat Models and LLM Models.