Information Retrieval and Synthesis Workflow with Gen AI

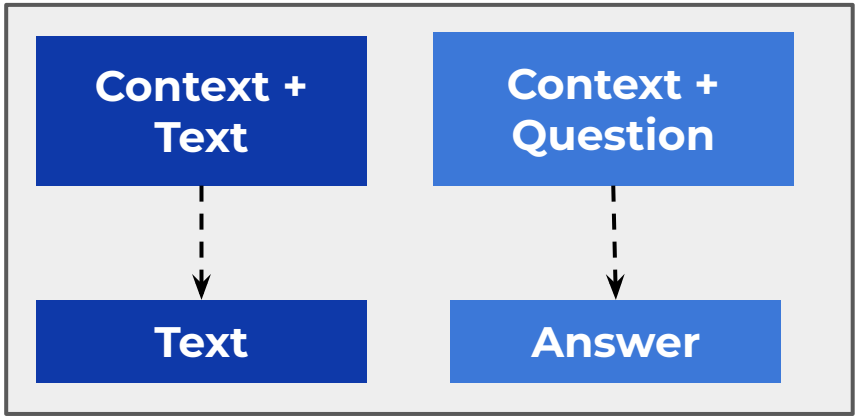
Agenda

In this session, we will discuss :

- Overview of Retrieval Augmented Generation (RAG) and its Working
- Building Blocks of RAG
- Data Preparation Process with respect to RAG
- Devising and Evaluating Prompts with respect to RAG

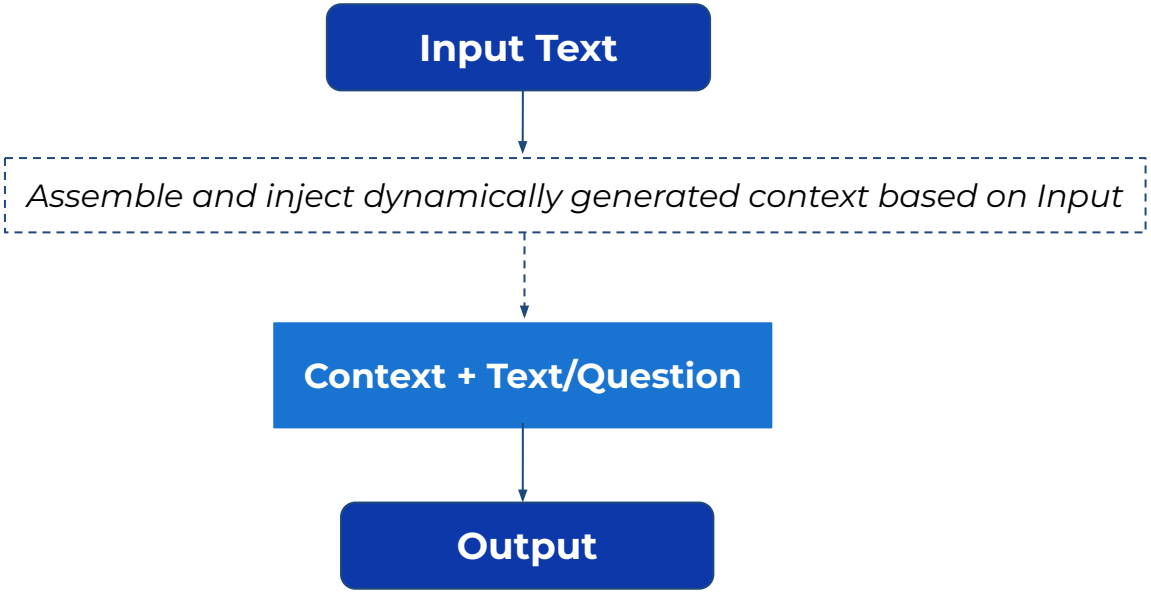
Retrieval Augmented Generation (RAG)

Contextualized

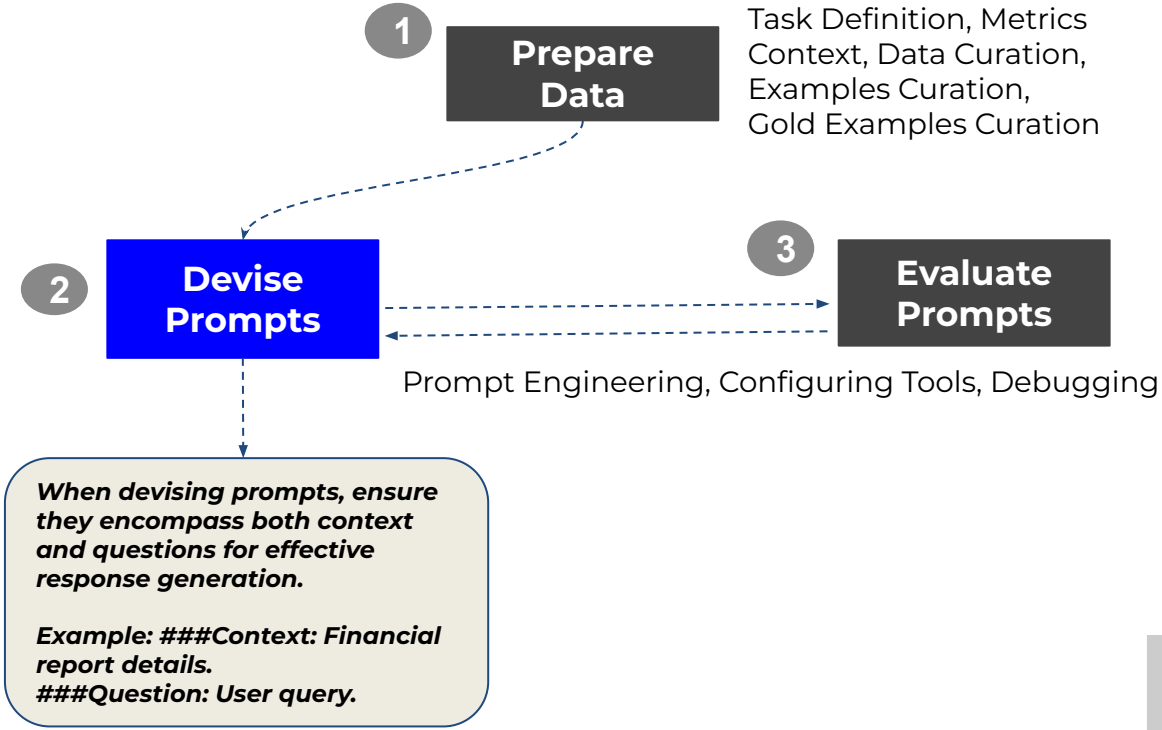


RAG

Working of RAG

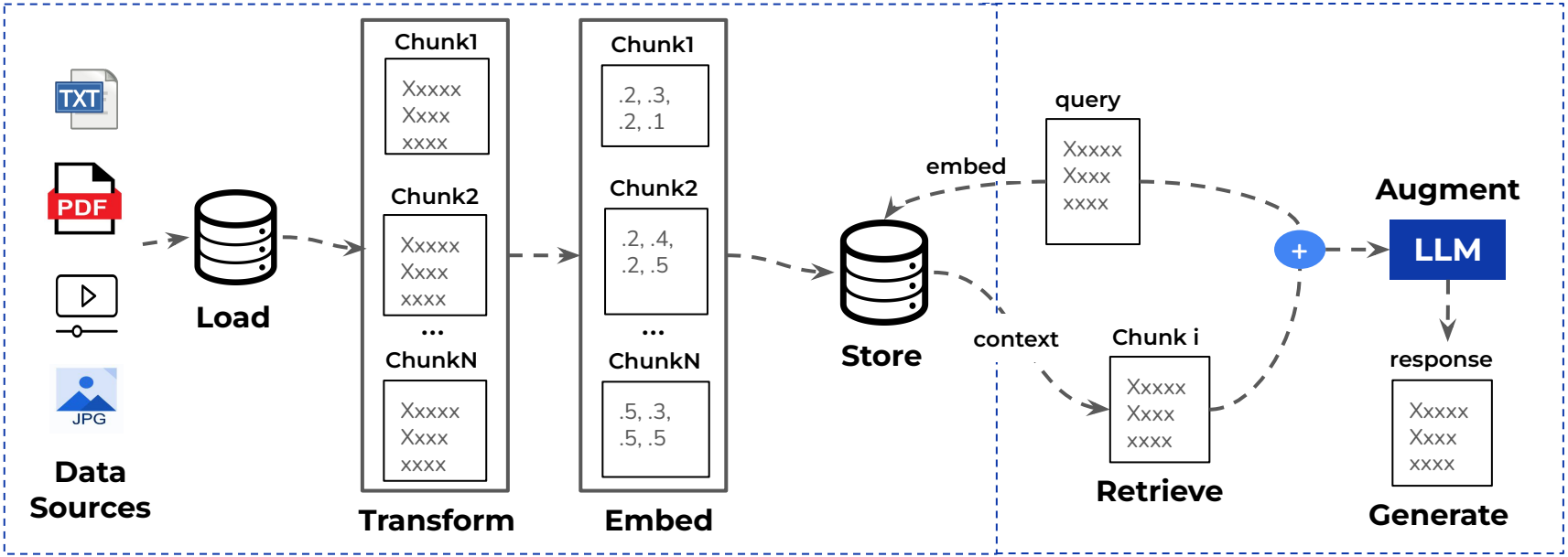


Structure of RAG



Building Blocks of RAG

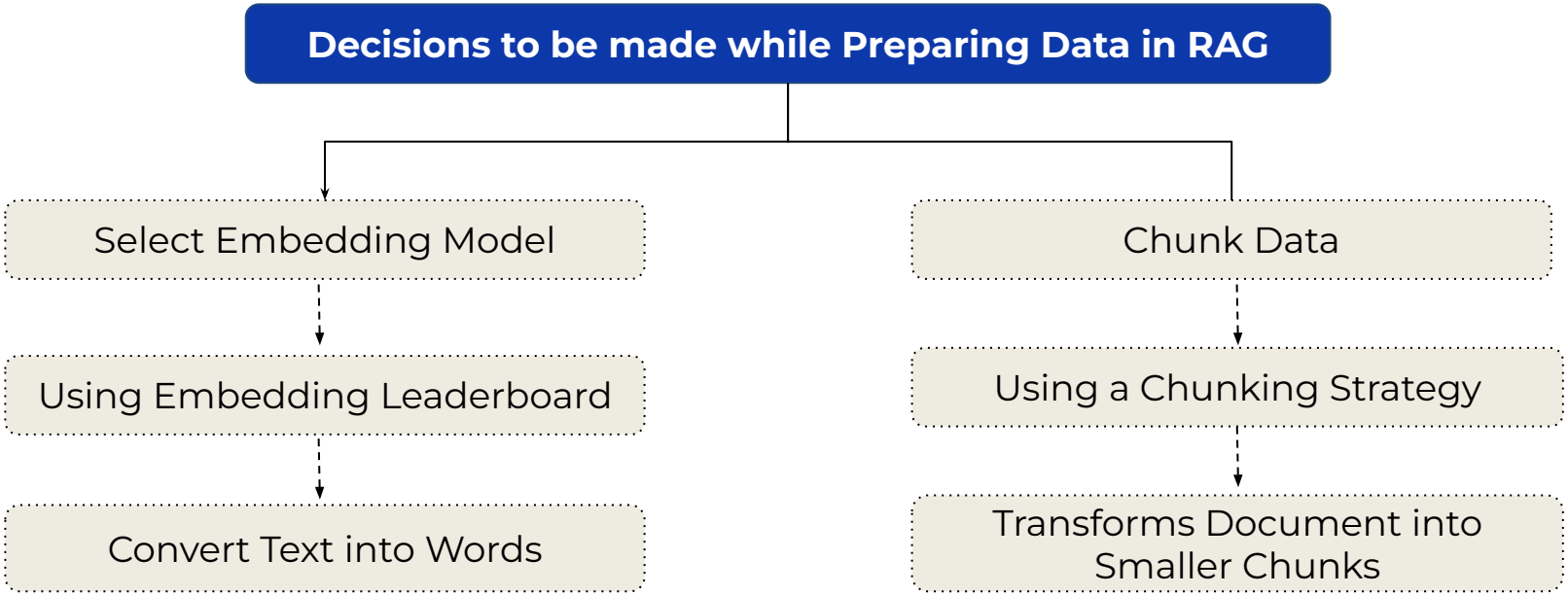
Building Blocks of RAG



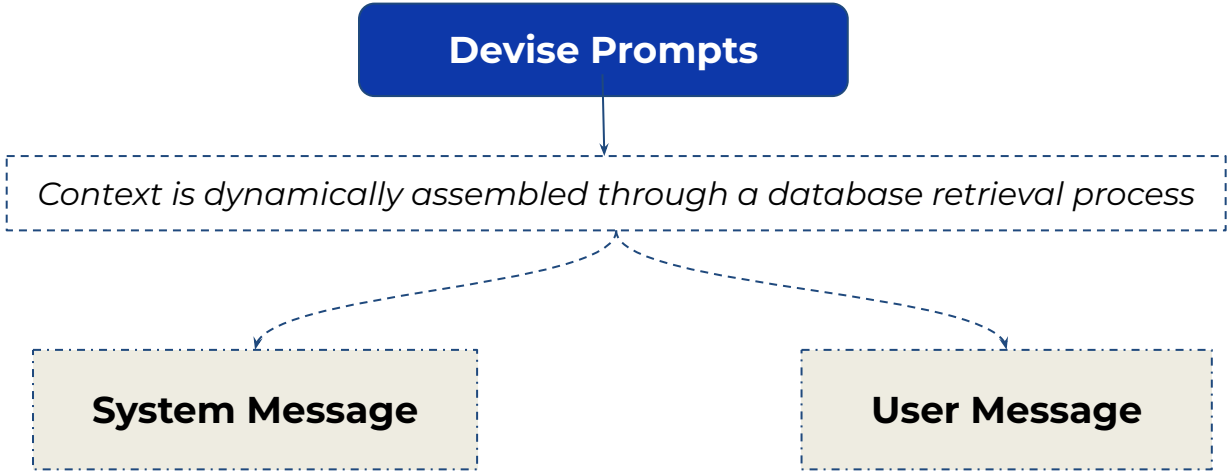
Step 1: Prepare Data

Step 2: Devise & Evaluate Prompts

Prepare Data in RAG



Devise Prompts in RAG



Evaluate Prompts in RAG

Evaluate Prompts

Accuracy

Assess the effectiveness of prompts used in RAG tasks.

Factors:

Clarity: *How clear is the prompt in conveying the task?*

Relevance: *Is the response relevant to the query posed by the user?*

Faithfulness to the context: *Is the context used correctly to create the response?*

Ensure prompts facilitate accurate and meaningful model predictions.

Data Preparation Process

Select Embedding Model

Embedding Model

Encodes text into vector representations that act as good features for LLM retrieval tasks

Selecting an open source model from Embedding Leaderboard

(To make this choice, look at the task to solve and then choose the embedding model close to Open AI `text-embedding-ada-002` on the leaderboard)

create a vectorized representation of the user_ input by using the `embed_query` method

Chunk Data

Chunk Data

Need for a method to parse individual portions of the document aligned with the embedding model

transform the pdf file into chunks of text that are no more than 512 chunks long

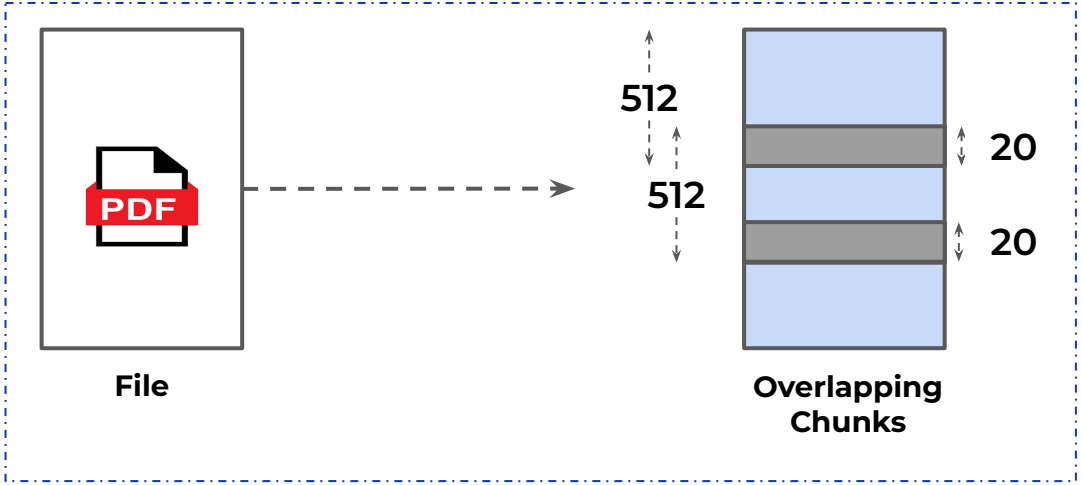
split text using the tokenizer of Hugging Face embedding model

(Use a chunking strategy)

Chunking Strategy: Example

chunk_overlap = 20

ensures that the chunks are related to each other (i.e., there is some continuity between the chunks)



Create Vector Database

Create vector database

Generate a vector for each chunk and save this chunk along with the vector representation

To add embeddings data to the database, create an index and push the embeddings by chunk to the index

Important components of the index to be specified during creation

Dimension of the embedding generated by the embedding model

Metric used to define the similarity between a pair of documents

(E.g., - Cosine similarity for indexing text)

Devising and Evaluating Prompts

Devising Prompts

Prompt Design

Context is dynamically assembled through a database retrieval process

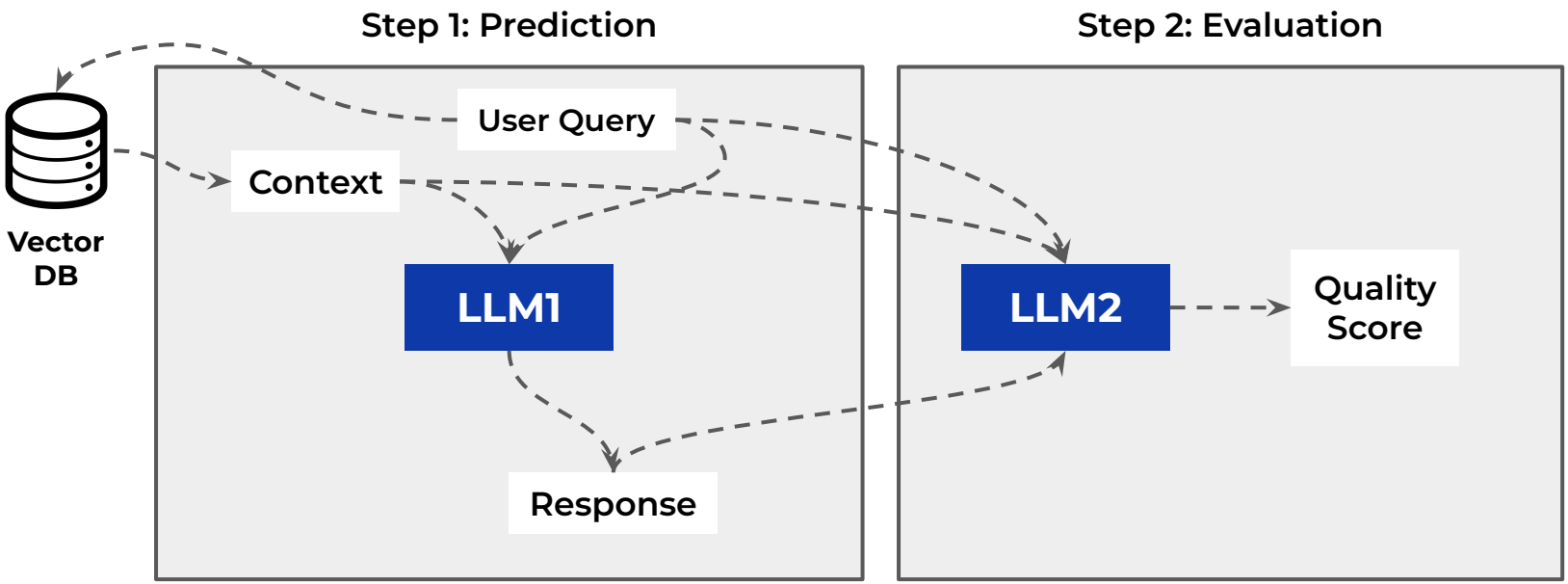
System Message

Here we provide a distinct set of instructions regarding the task

User Message

Here we clearly define the sections where the context will be inserted and where the user input will be injected

Evaluation Process in RAG



Summary