

## **LangChain and its RAG Components (Retrieval-Augmented-Generation)**

### **1. Data Loader**

- Bring your documents (PDF, DOCX, ...)
- PDFLoader, UnstructuredLoader, WebBaseLoader

### **2. Text Splitter**

- Break long texts into small meaningful chunks
- RecursiveCharacterTextSplitter

### **3. Metadata Attachment**

- Attach filename, page number, heading with every chunk for tracing
- Metadata field in document

### **4. Embeddings**

- Convert text into numeric vectors for better searching using similarity
- OpenAIEmbeddings, HuggingFaceEmbeddings, GoogleEmbeddings

### **5. Vector Store Index**

- It stores the embeddings and also searches them quickly
- Chroma, Pinecone, FAISS, Milvus

### **6. Retriever**

- Takes user query and retrieves relevant chunks
- VectorDB.as\_retriever

### **7. Query Transformation**

- Improve user's query like rephrase, multi query to make retrieval better
- MultiQueryRetriever, ContextCompressionRetriever

### **8. Prompt Template**

- Make good format for retrieved content to give it to LLM
- PromptTemplate, ChatPromptTemplate

### **9. LLM Reasoning and Answer Generation**

- Gives response by reading context
- ChatOpenAI, ChatAnthropic, ChatGoogle, local models

### **10. Output Parser**

- Makes the LLM response into clean format or structured JSON
- StrOutputParser, JsonOutputParser