Master LangChain Retrievers Deep Dive

Complete reference covering retriever theory, implementations, RAG/Agentic mappings, decision frameworks, and hands-on mini labs.

1. Retriever Fundamentals

A retriever is responsible only for fetching relevant documents given a query. It does not generate answers. LangChain retrievers provide a unified interface over many retrieval backends such as vector databases, keyword search engines, and APIs.

2. Retriever Types Overview

- VectorStoreRetriever – semantic similarity search using embeddings
- BM25Retriever – sparse keyword-based retrieval
- WikipediaRetriever – public knowledge via Wikipedia API
- SelfQueryRetriever – LLM-generated metadata filters
- MultiQueryRetriever – query expansion via LLM
- ParentDocumentRetriever – hierarchical retrieval
- EnsembleRetriever – combines multiple retrievers
- TimeWeightedRetriever – recency-aware retrieval

3. Code Examples Per Retriever

VectorStoreRetriever Example

```
from langchain.vectorstores import FAISS
from langchain.embeddings import OpenAIEmbeddings

db = FAISS.from_texts(texts, OpenAIEmbeddings())
retriever = db.as_retriever(search_kwargs={"k": 5})
```

WikipediaRetriever Example

```
from langchain.retrievers import WikipediaRetriever

retriever = WikipediaRetriever(top_k_results=3)
docs = retriever.get_relevant_documents("Tesla Motors history")
```

BM25Retriever Example

```
from langchain.retrievers import BM25Retriever

retriever = BM25Retriever.from_texts(texts)
```

4. Retriever Mapping to RAG & Agentic RAG

- Simple RAG → VectorStoreRetriever
- Hybrid RAG → VectorStoreRetriever + BM25Retriever
- Agentic RAG → Multiple retrievers as tools
- Adaptive RAG → Router selects retriever dynamically
- Hybrid Agentic RAG → Agent controls structured + semantic retrievers

5. Retriever Decision Flowchart

Is data structured?

- ■■ Yes → SQL / Feature Store Retriever
- ■■ No
  - ■■ Exact terms important? → BM25Retriever
  - ■■ Semantic meaning important? → VectorStoreRetriever
  - ■■ Public factual data? → WikipediaRetriever
  - ■■ Mixed → Hybrid / EnsembleRetriever

6. Hands-on Mini Labs

- Lab 1: Build FAISS-based semantic retriever over PDFs
- Lab 2: Add BM25 retriever and compare results
- Lab 3: Combine retrievers using EnsembleRetriever
- Lab 4: Add WikipediaRetriever as fallback
- Lab 5: Convert retrievers into Agentic RAG tools

Final Takeaway: Retrievers are interchangeable strategies. Mastering RAG means mastering retrieval selection and composition.