Langchain

Converse with your data

Adnan Ribic

Overview

- Open-source developer framework for building LLM applications
- Python and TypeScript
- Focused on composition and modularity

Components

- Prompts
- Models (LLM’s: 20+ integrations)
- Indexes (Document loaders, text splicers, vector stores, retrievers)
- Chains (20+ different types)
- Agents(Agent types: 5+ types, Agent Toolkits: 10+ implementations)

Retrival Augmented Generation

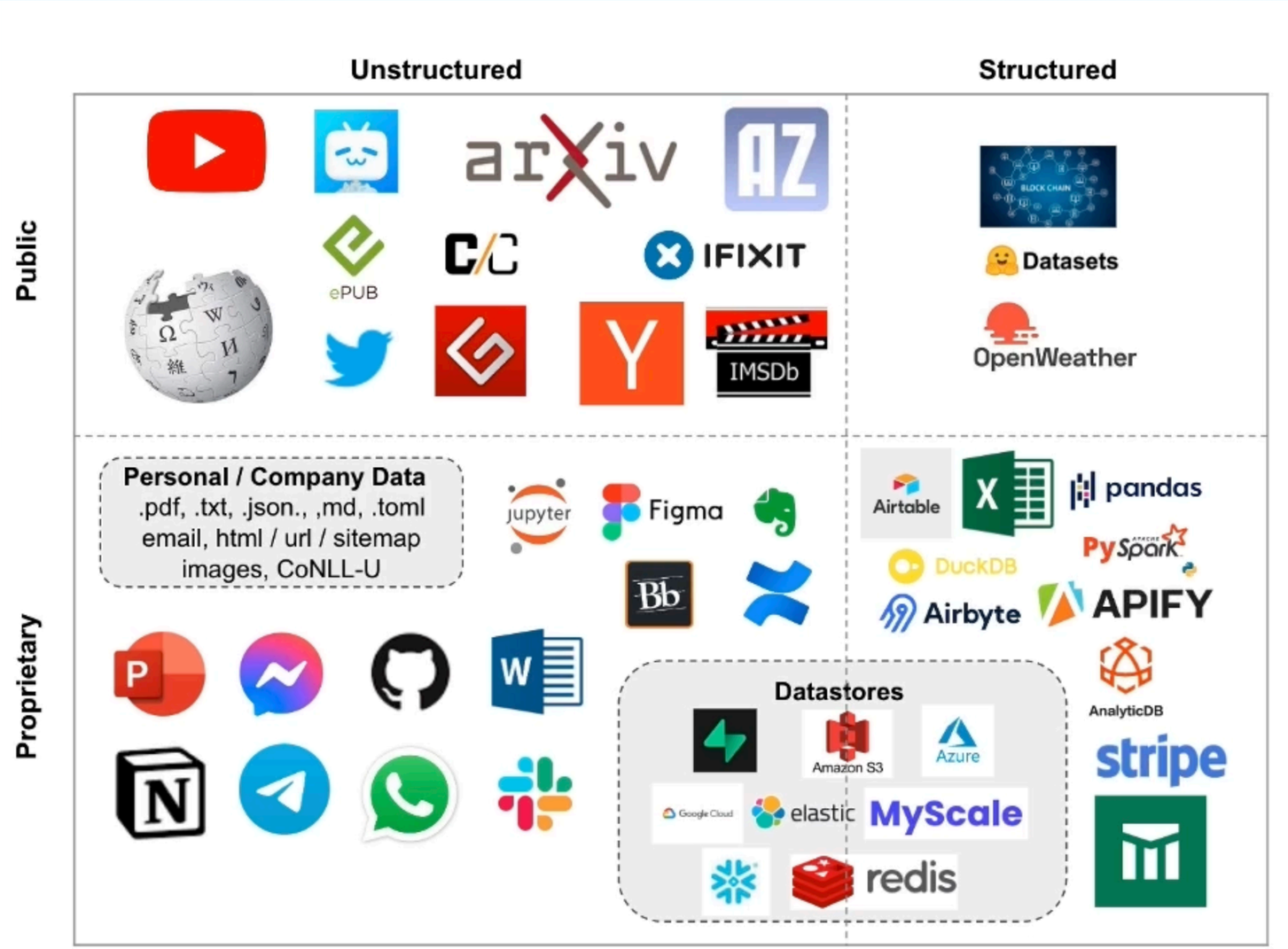
- Document loading
- Splitting
- Storage
- Retrival
- Output

Document loading

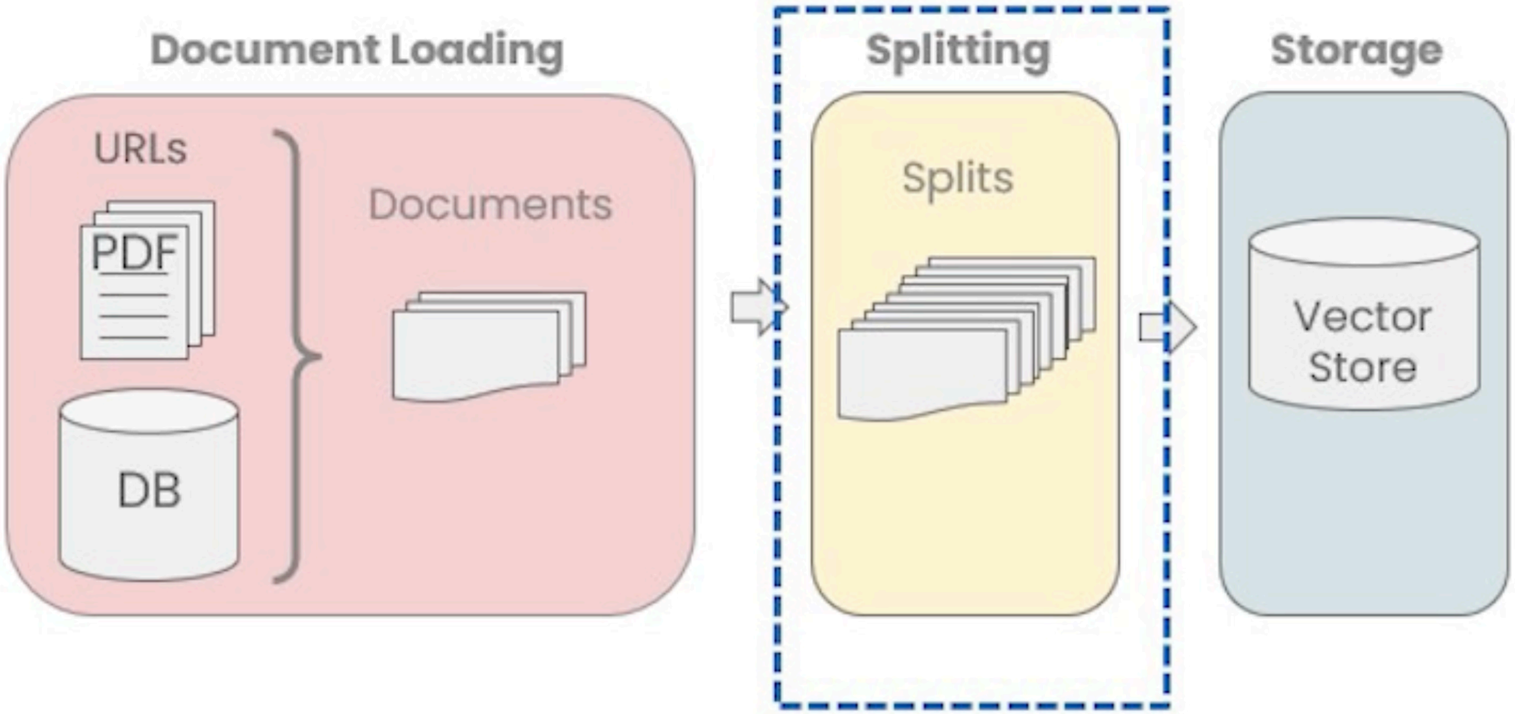
Loaders

- Loaders deal with the specifics of accessing and converting data
- Accessing websites, data bases, YouTube…
- Data types like PDF, HTML, JSON…
- Returns a list of ‘Document’ objects

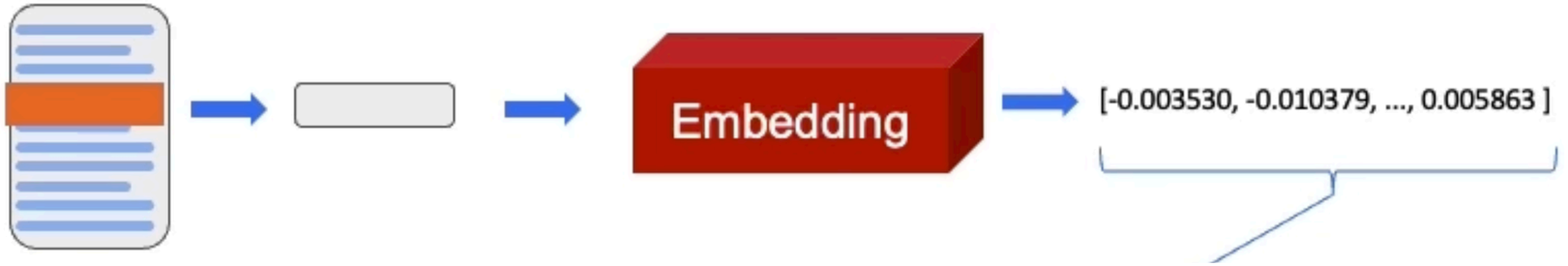
Document loaders



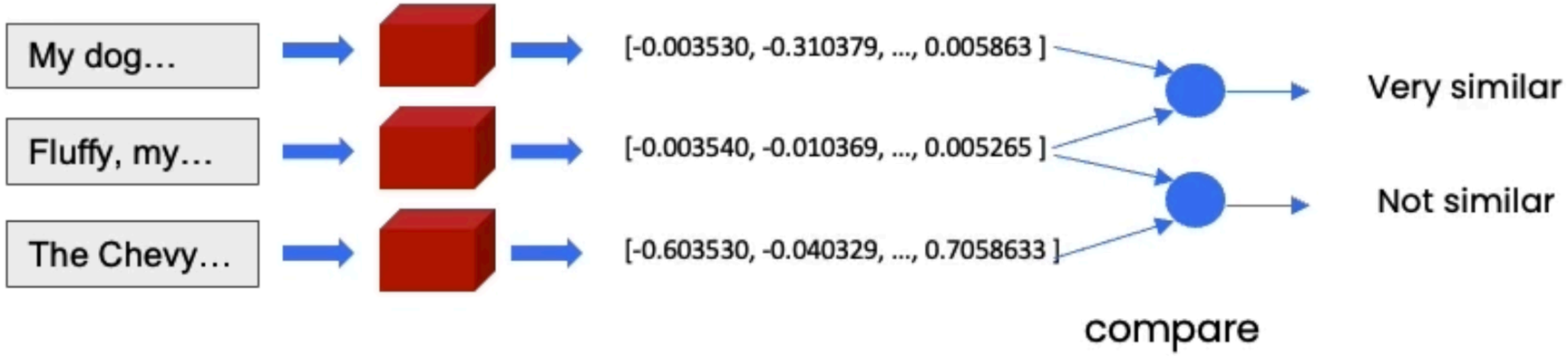
Document Spliting



Vectors and embedding



1) My dog Rover likes to chase squirrels.
2) Fluffy, my cat, refuses to eat from a can.
3) The Chevy Bolt accelerates to 60 mph in 6.7 seconds.

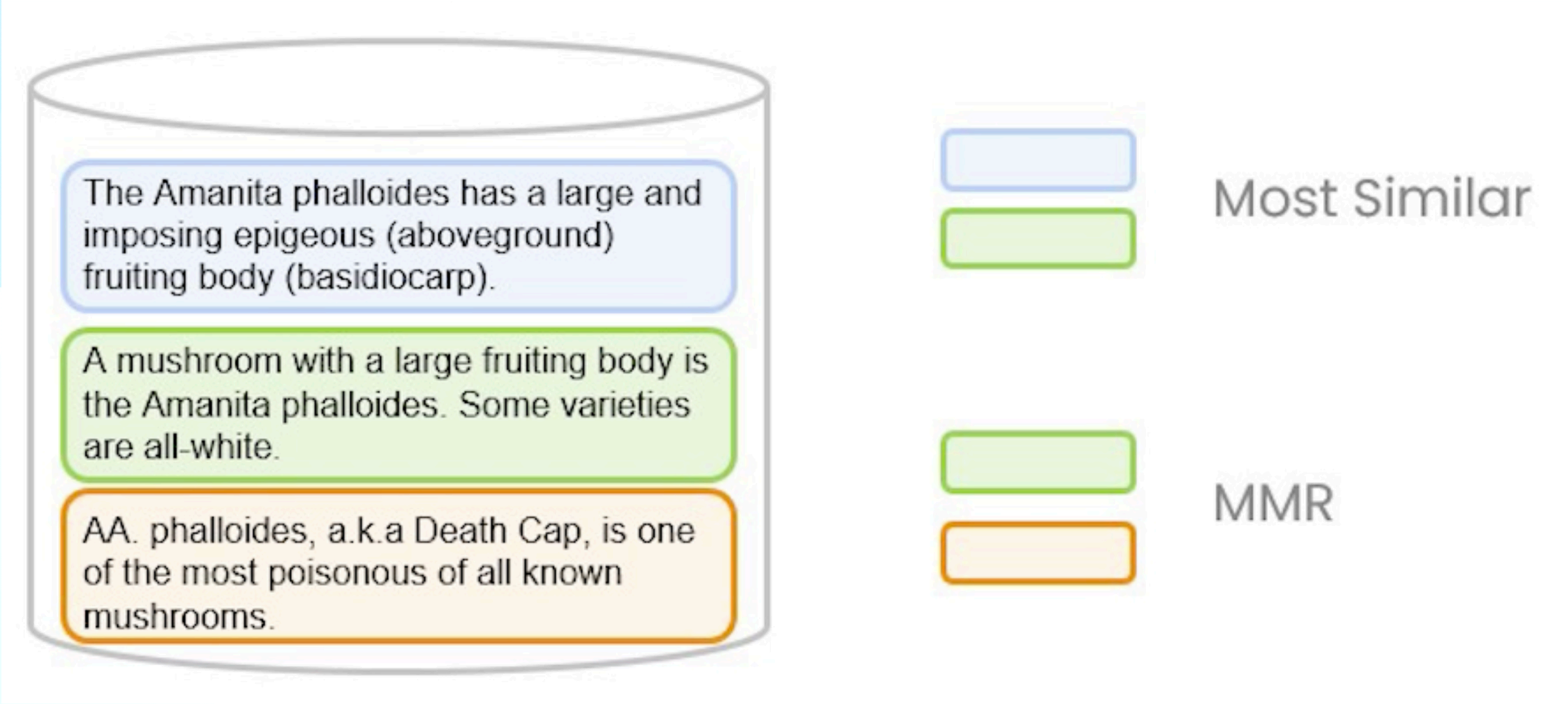


Retrival

- Accessing/indexing the data in the vector store (basic semantic similarity, Maximum marginal relevance, Including metadata)
- LLM Aided Retrieval

Retrival

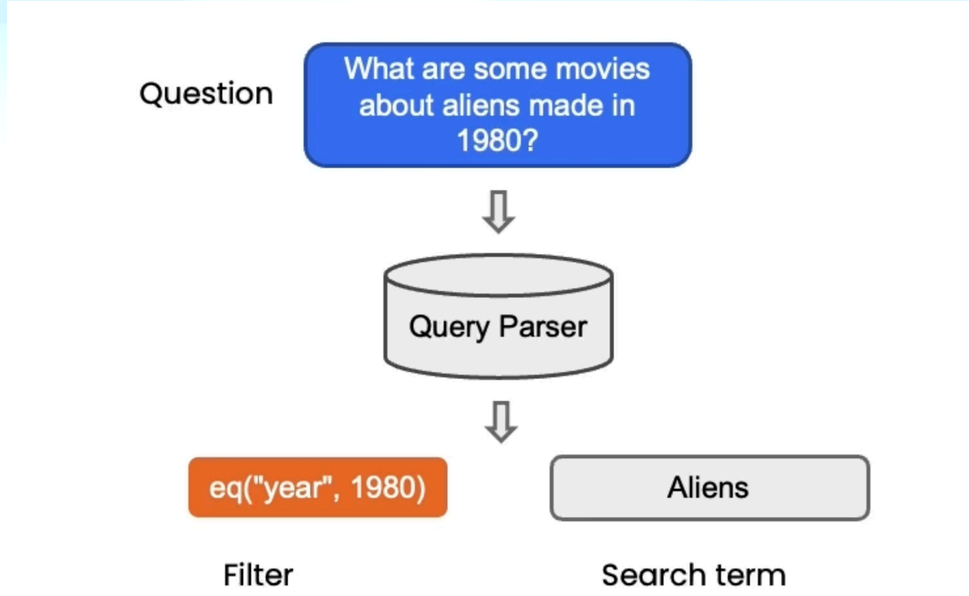
Maximum marginal relevance (MMR)



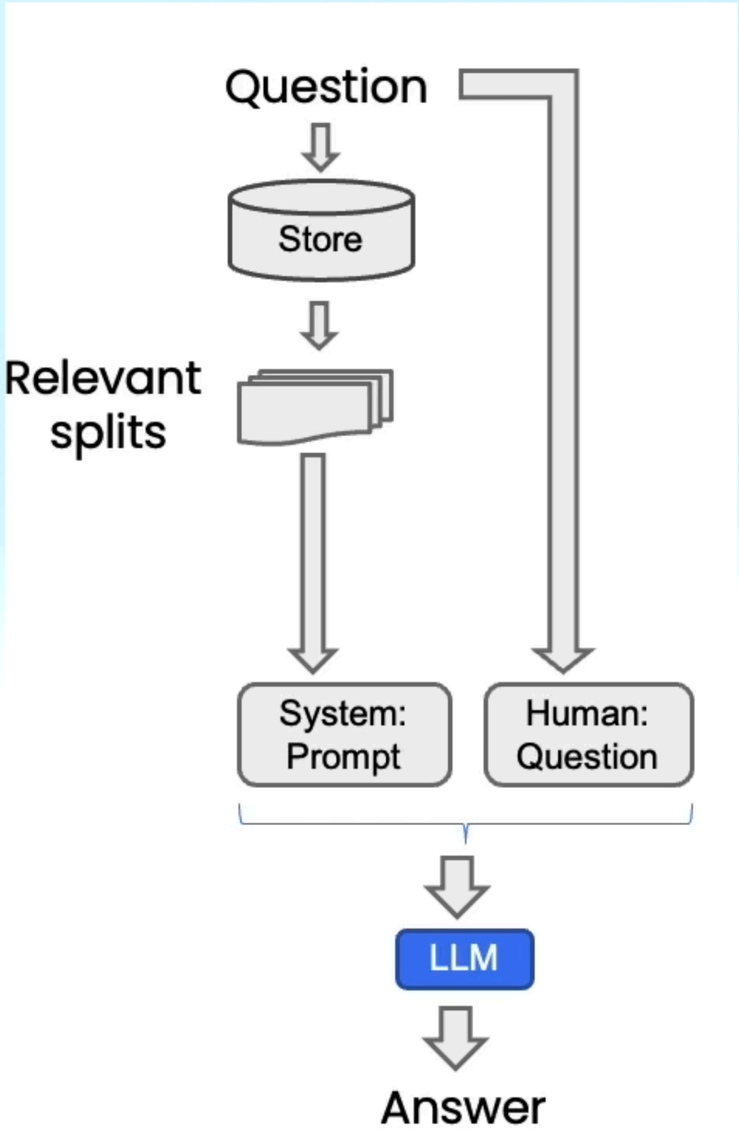
Retrival

LLM Aided Retrieval

- There are citations where QUERY to the DB is more then just the question asked
- We use LLM to convert question to the query



RetrievalQA chain



RetrievalQA chain

3 additional methods