

LangChain Chat with Your Data

Overview

- Open-source development framework for LLM applications
- Python and Javascript (TypeScript) packages
- Focused on composition and modularity
- Key value adds:
 - Modular components
 - Use cases: Common ways to combine components

Components

Models

- LLMs: 20+ integrations
- Chat models
- Text embedding models: 10+ integrations

Prompts

- Prompt templates
- Output parsers: 5+ implementations
- Example selectors: 5+ implementations

Indexes

- Document loaders: 50+ implementations
- Text Splitters: 10+ implementations
- Vector stores: 10+ implementations
- Retrievers: 5+ implementations/integrations

Chains

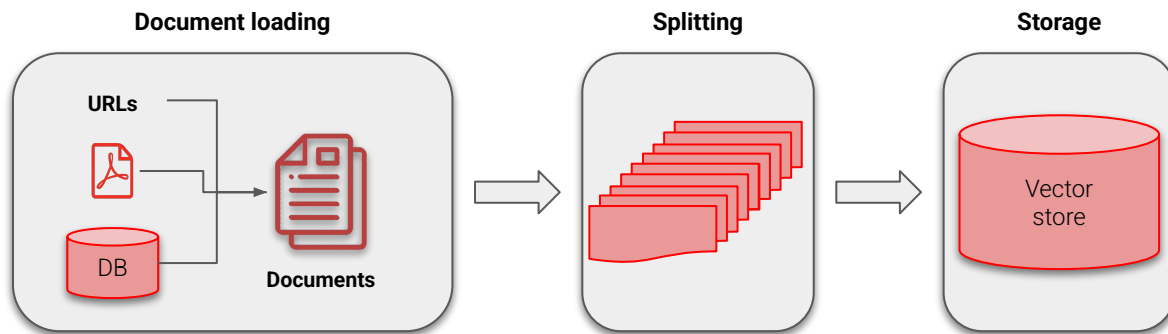
- Prompt + LLM + Output parsing
- Can be used as a building blocks for longer chains
- More application specific chains: 20+ types

Agents

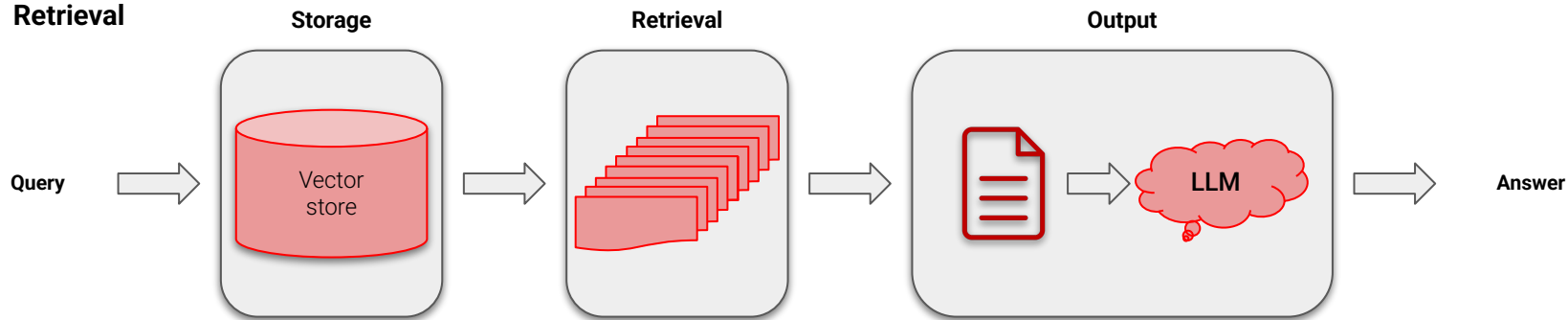
- Agents types: 5+ types
 - Algorithms for getting LLMs to use tools
- Agent toolkits: 10+ implementations
 - Agents armed with specific tools for a specific applications

Retrieval Augmented Generation

Vector store loading



Retrieval



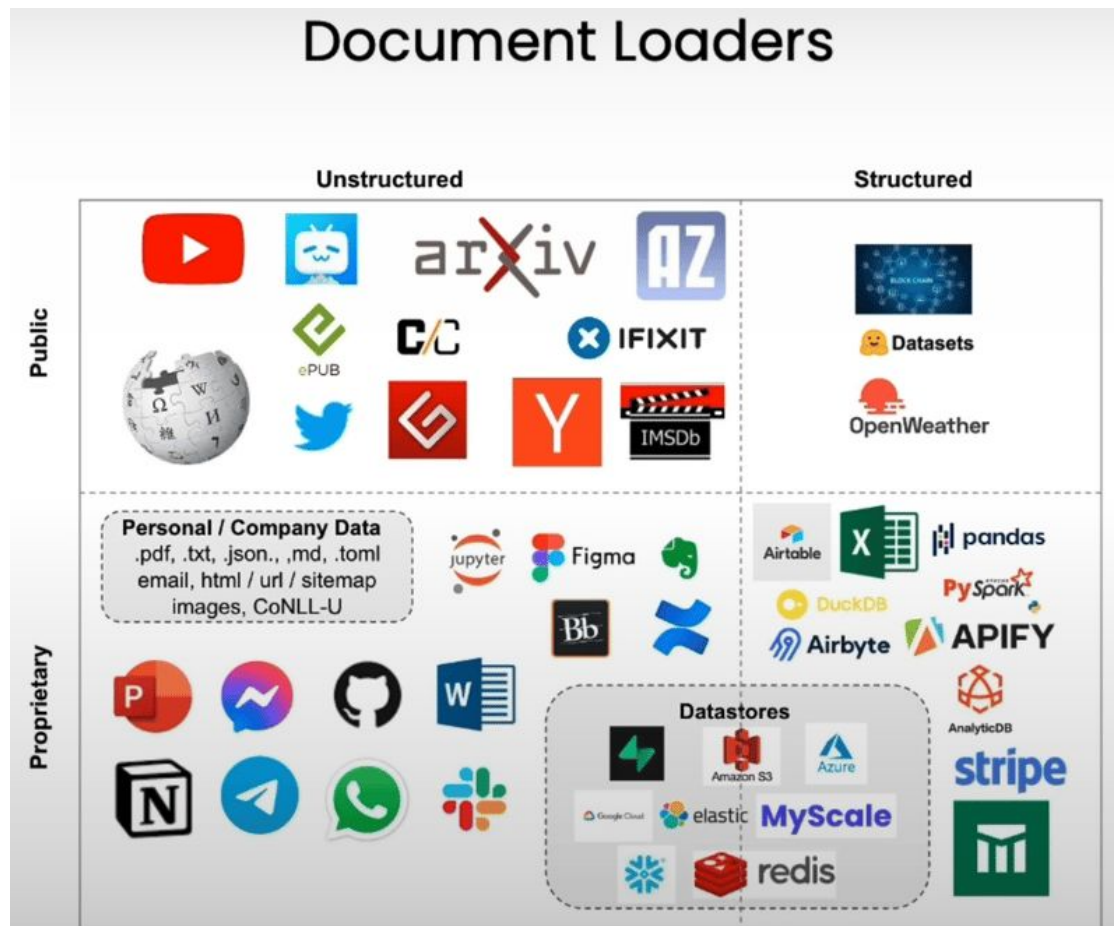
Loaders

Loaders deal with the specifics of accessing and converting data

- **Accessing**
 - Web sites
 - Databases
 - Youtube
 - arXiv
 - and many more
- **Data types**
 - PDF
 - HTML
 - JSON
 - Word
 - Powerpoint
 - and many more

```
[  
  Document(page_content="MachineLearning-Lecture01 \nInstructor (Andrew Ng): Okay.  
  Good morning. Welcome to CS229....",  
  metadata={'source': 'documents/MachineLearning-Lecture01.pdf', 'page': 0})  
  .....  
]
```

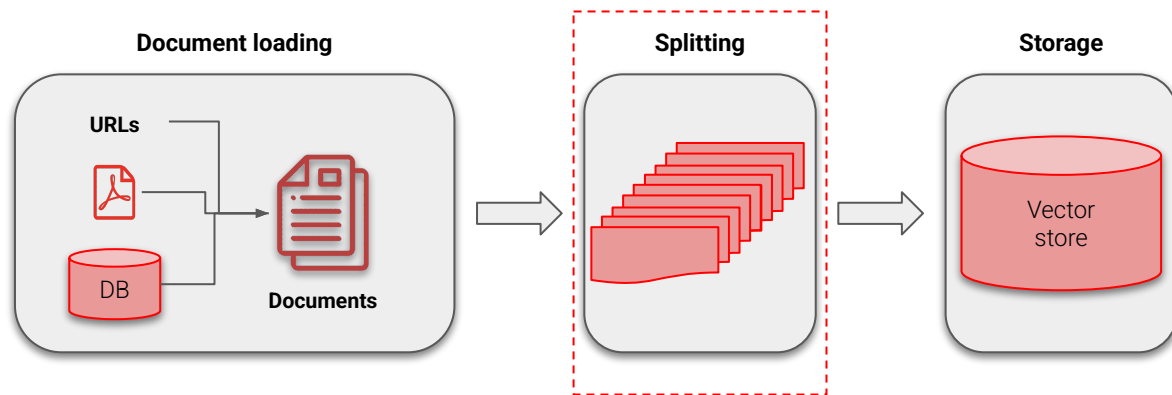
Document Loaders



Document Splitting

Splitting documents into smaller chunks

- Retaining meaningful relationships!



...

on this model. The Toyota Camry has a head-snapping 80 HP and an eight-speed automatic transmission that will

...

Chunk 1: on this model. The Toyota Camry has a head-snapping

Chunk 2: 80 HP and an eight-speed automatic transmission that will

Question: What are the specifications of the Camry?

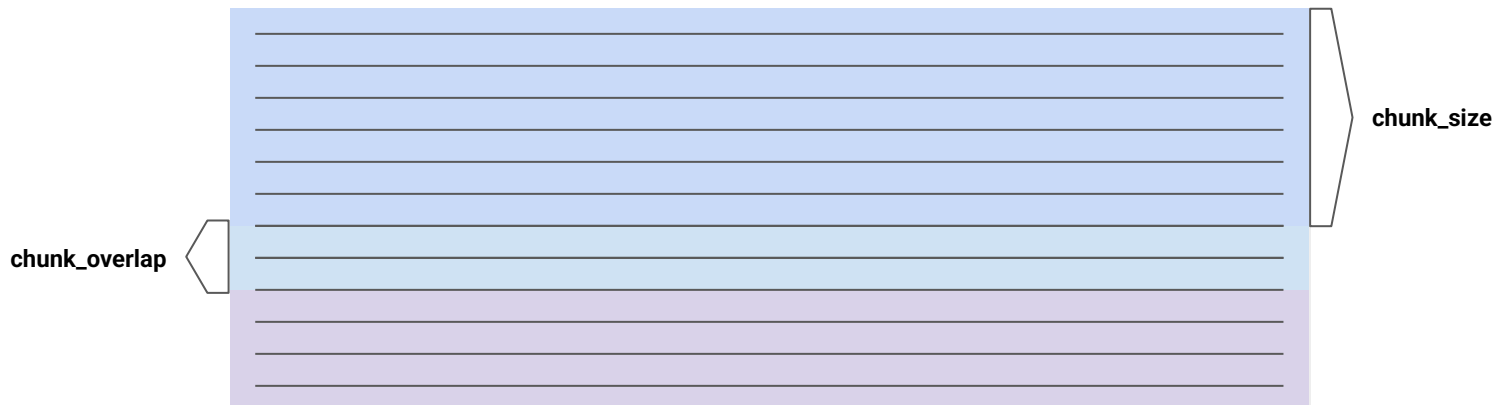
Example Splitter

```
langchain.text_splitter.CharacterTextSplitter(  
    separator: str = "\n\n",  
    chunk_size=4000,  
    chunk_overlap=200,  
    length_function=<builtin function len>,  
)
```

Methods:

create_documents() - create documents from a list of texts.

split_documents() - split documents

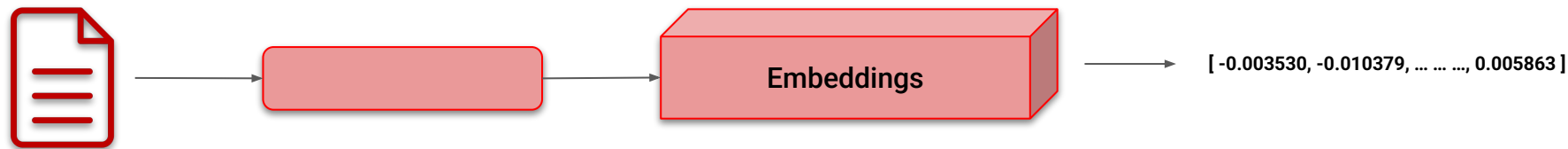


Types of Splitters

langchain.text_splitter.

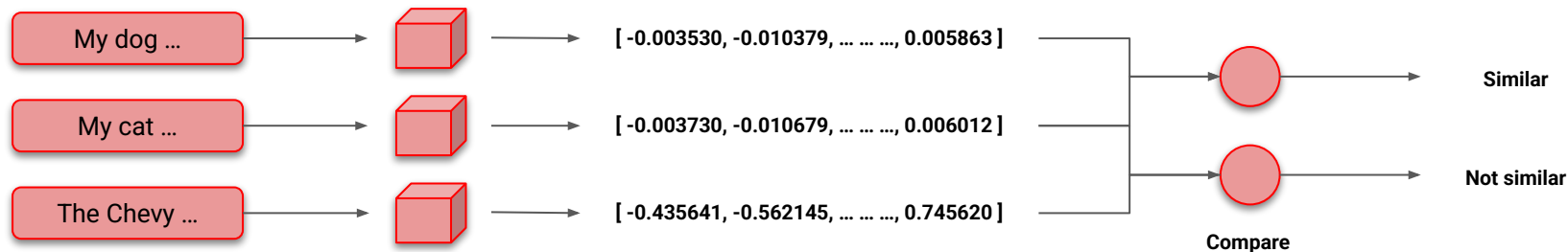
- `CharacterTextSplitter()` - Implementation of splitting text that looks at characters.
- `MarkdownHeaderTextSplitter()` - Implementation of splitting markdown files based on specified headers.
- `TokenTextSplitter()` - Implementation of splitting text that looks at tokens.
- `SentencesTransformerTokenTextSplitter()` - Implementation of splitting text that looks at tokens
- `RecursiveCharacterTextSplitter()` - Implementation of splitting text that looks at characters. Recursively tried to split by different characters to find one that works.
- `Language()` - for CPP, Python, Ruby, Markdown etc
- `NLTKTextSplitter()` - Implementation of splitting text that looks at sentences using NLTK (Natural Language Tool Kit)
- `SpacyTextSplitter()` - Implementation of splitting text that looks at sentences using Spacy.

Embeddings

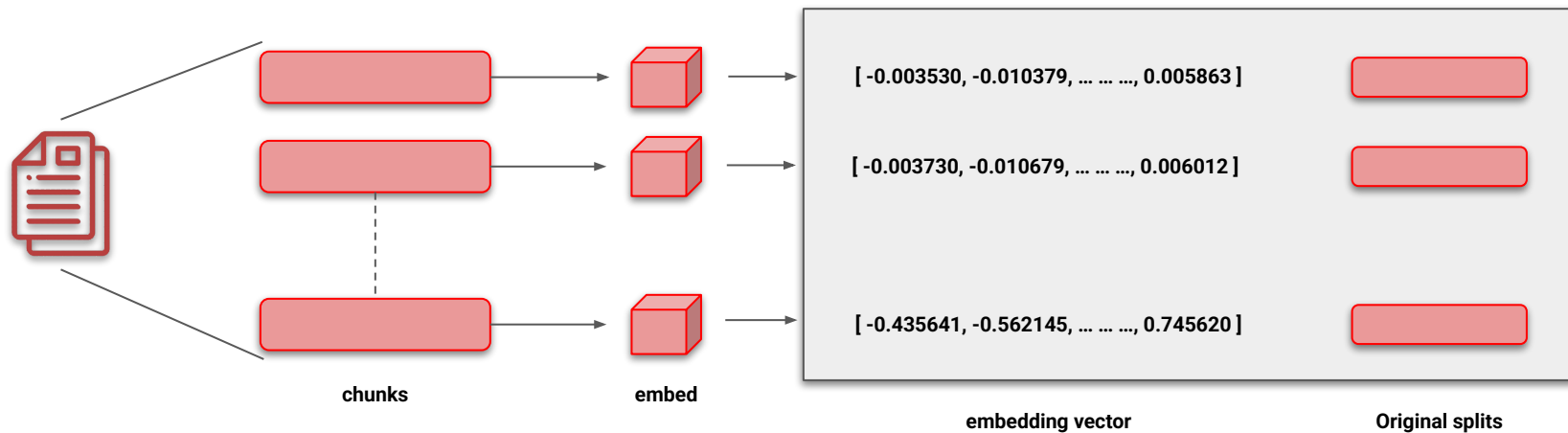


- **Embedding vector captures content / meaning**
- **Text with similar content will have similar vectors**

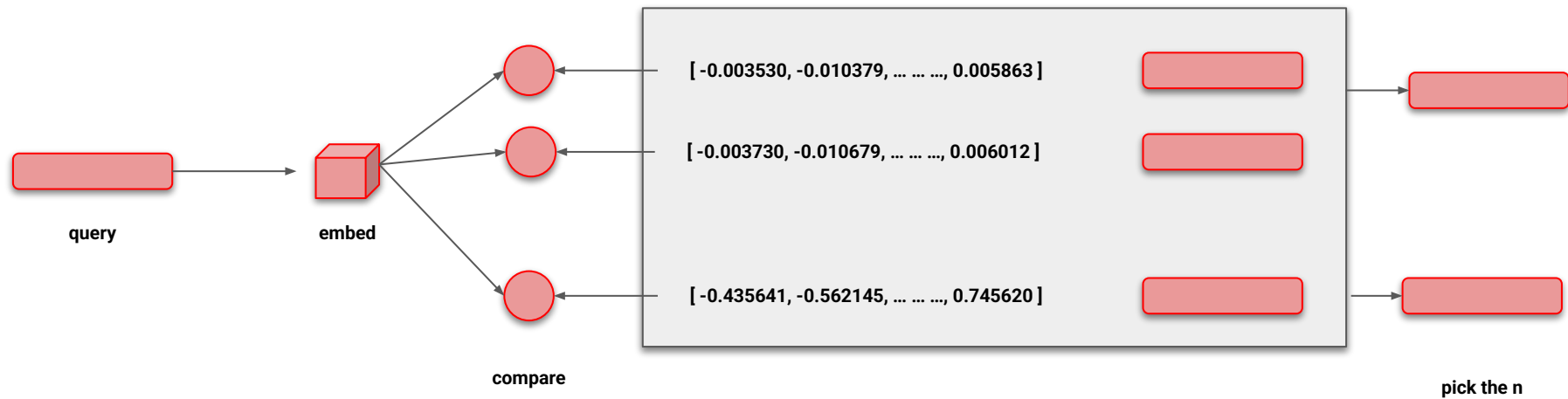
- 1) My dog likes to chase squirrels
- 2) My cat refuses to eat from a can
- 3) The Chevy bolt accelerates to 60 mph in 6.7 seconds



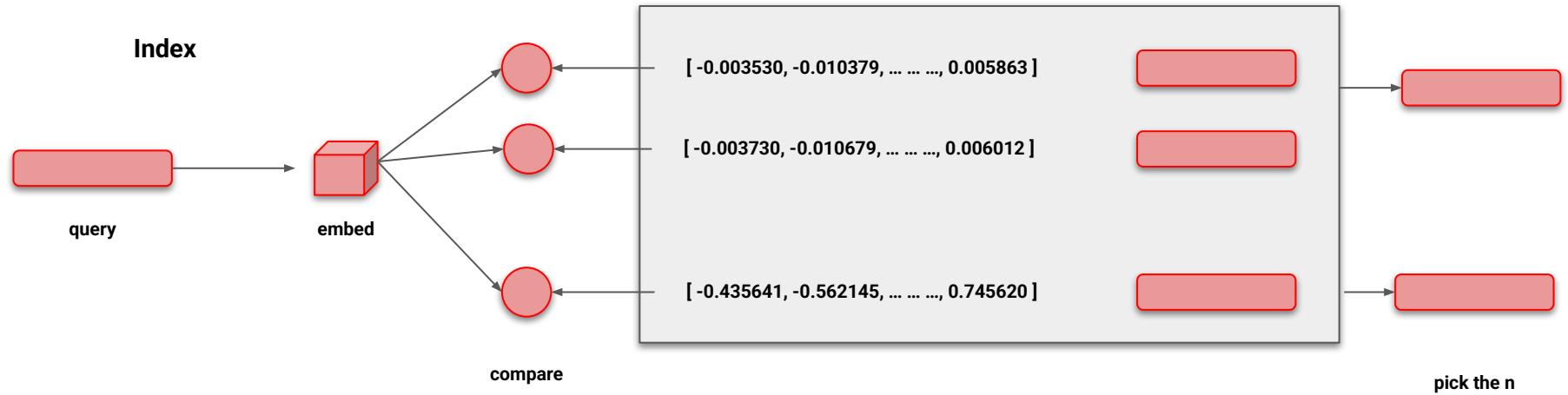
Vector Database (Create)



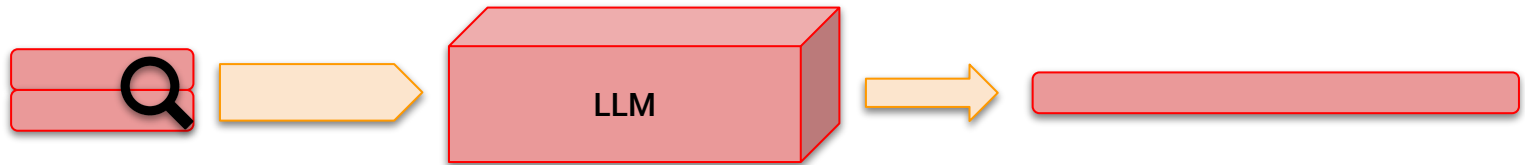
Vector Database (Index)



Vector Store / Database

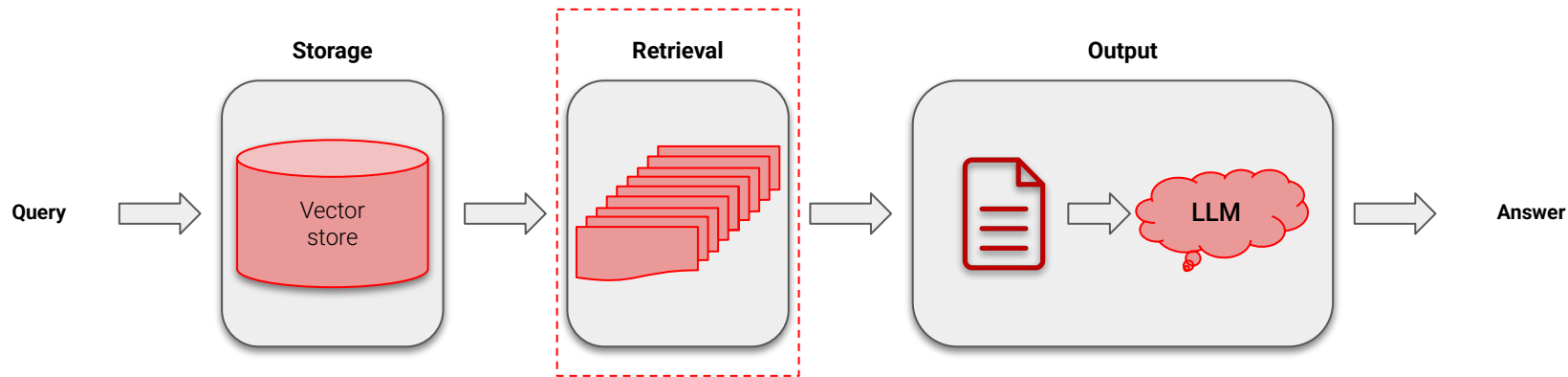


Process with LLM



The returned values can now fit into LLM context

Retrieval



Accessing / indexing the data into vector store

- Basic semantic similarity
- Maximum marginal relevance
- Including metadata

LLM aided retrieval

Maximum Marginal Relevance (MMR)

You may not always want to choose the most similar responses

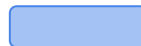


Tell me about all-white mushrooms with large fruiting bodies

The Amanita phalloides has a large and imposing epigeous (aboveground) fruiting body (basidiocarp).

A mushroom with a large fruiting body is the Amanita phalloides. Some varieties are all-white.

AA. phalloides, a.k.a Death Cap, is one of the most poisonous of all known mushrooms.



Most similar



MMR



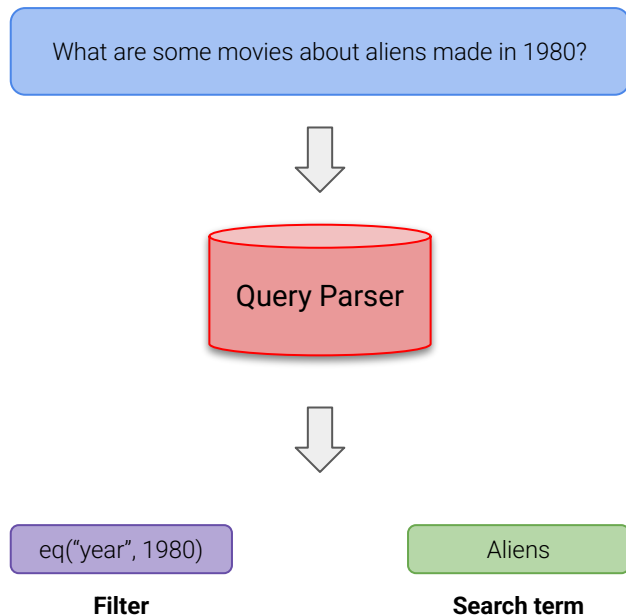
MMR Algorithm

- Query the vector store
- Choose the 'fetch_k' most similar responses
- Within those responses choose the 'k' most diverse



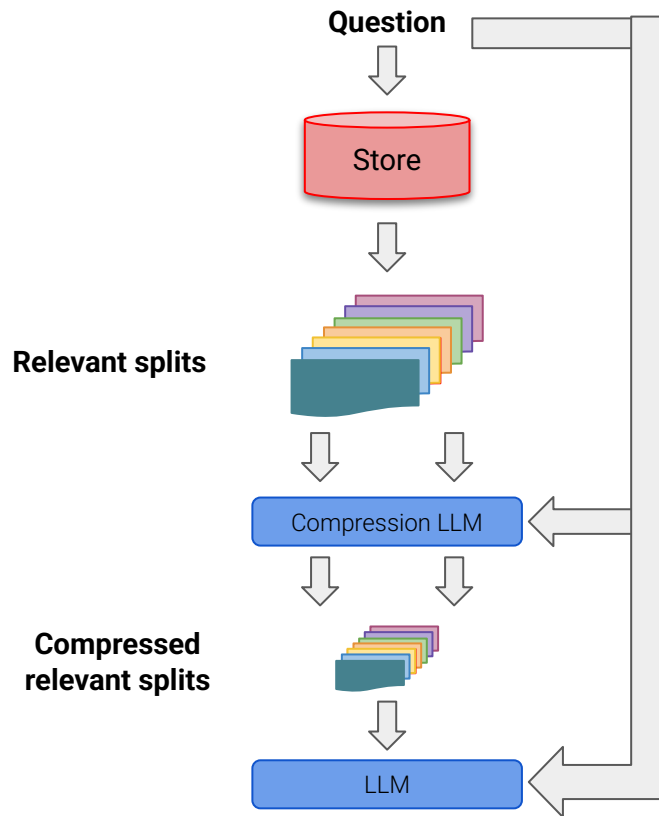
LLM Aided Retrieval

- There are several situations where the Query applied to the DB is more than just the Question asked
- One is SelfQuery, where we use an LLM to convert the user question into a query

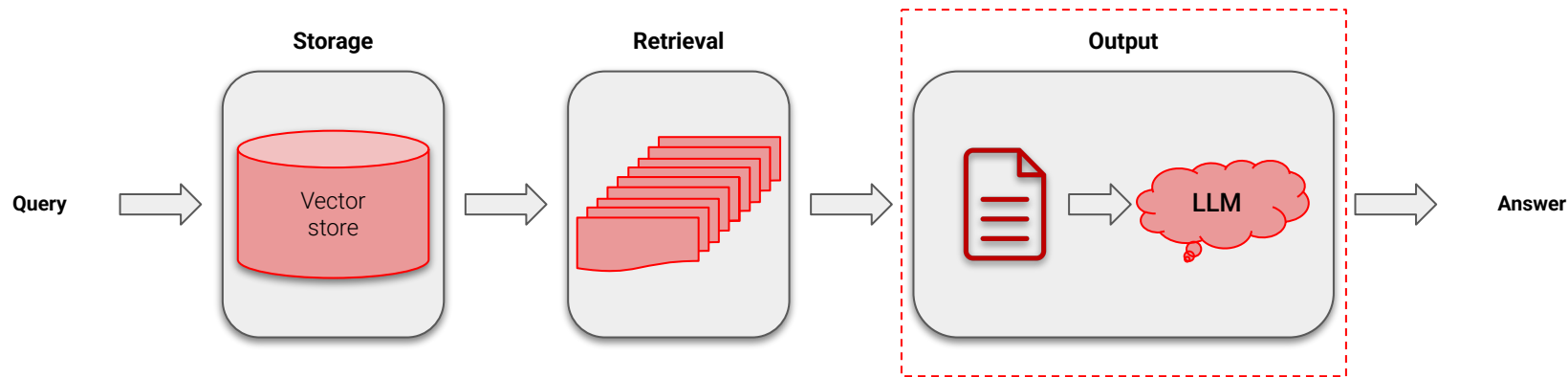


Compression

- Increase the number of results you can put in the context by shrinking the responses to only the relevant information



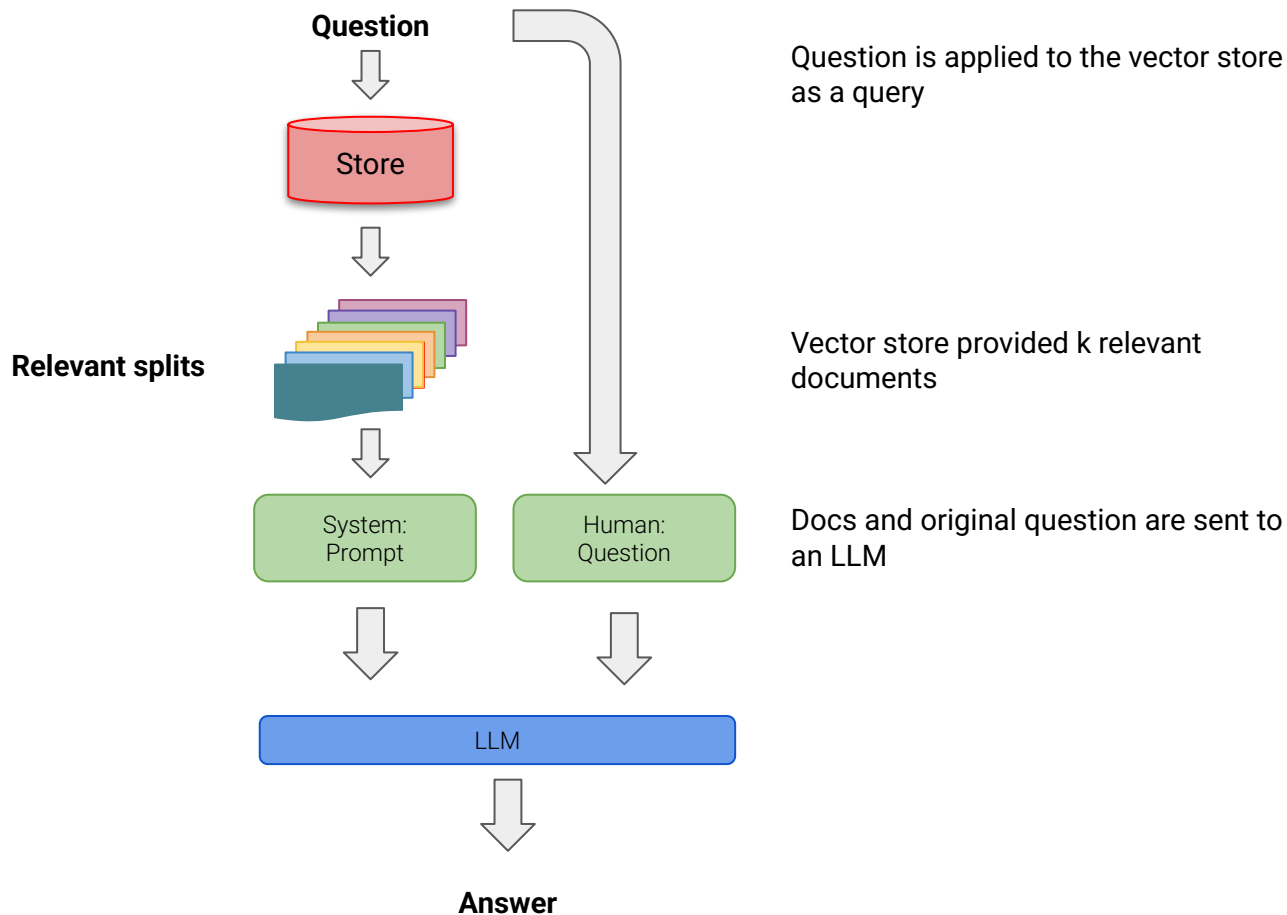
Question Answering



- Multiple relevant documents have been retrieved from the vector store
- Potentially compress the relevant splits to fit into LLM context
- Send the information along with our question to an LLM to select and format an answer

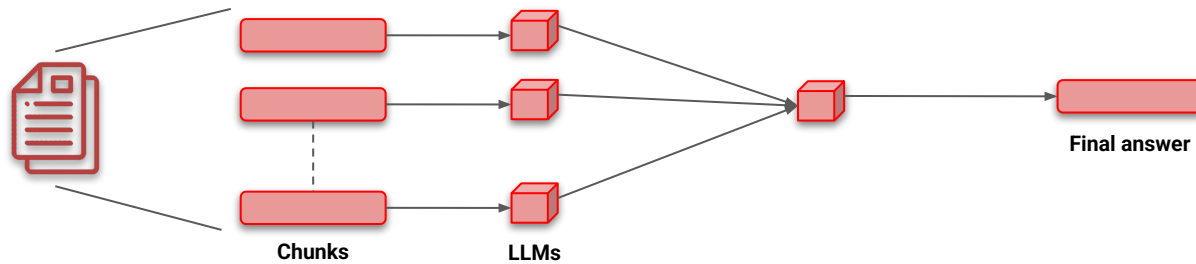
RetrievalQA Chain

Stuff Method

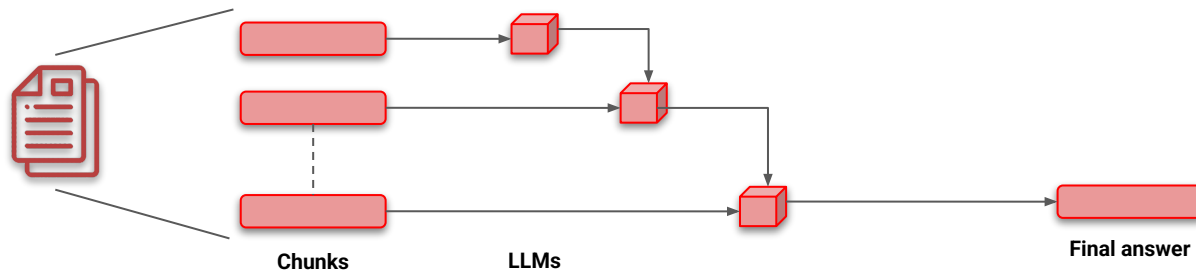


Additional Methods

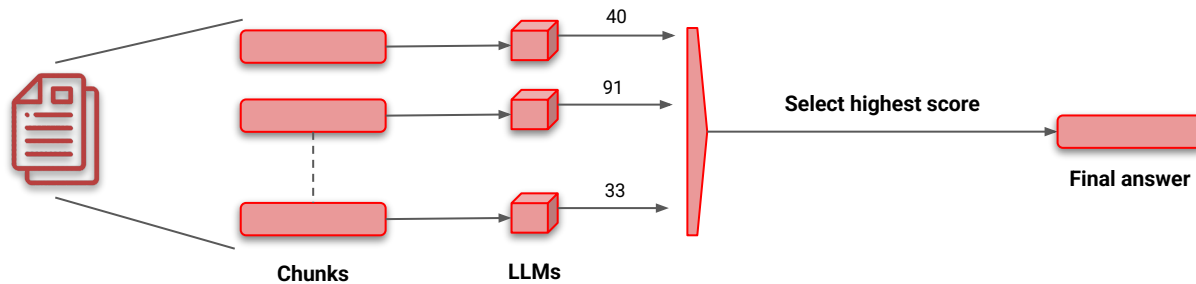
Map reduce



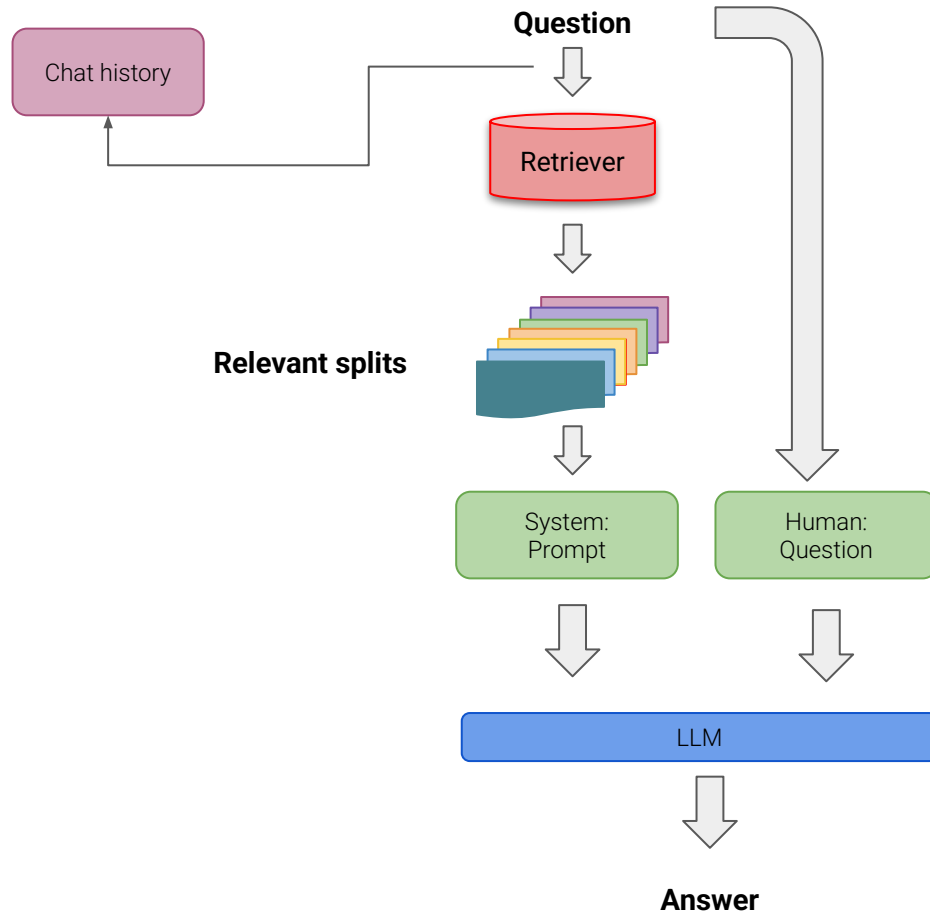
Refine



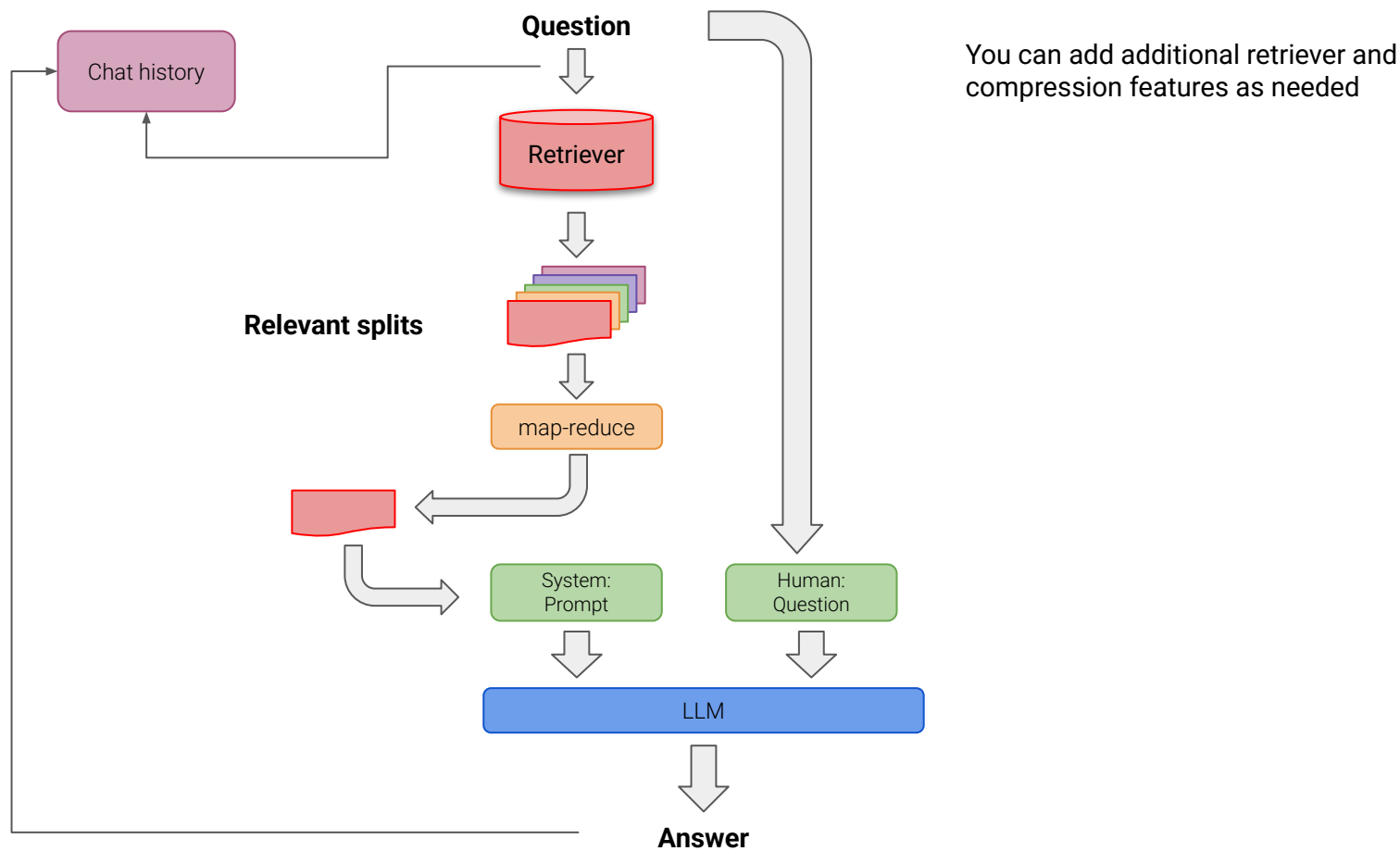
Map rerank



Conversational Retrieval Chain with Memory



Conversational Retrieval Chain with Memory



THANK YOU