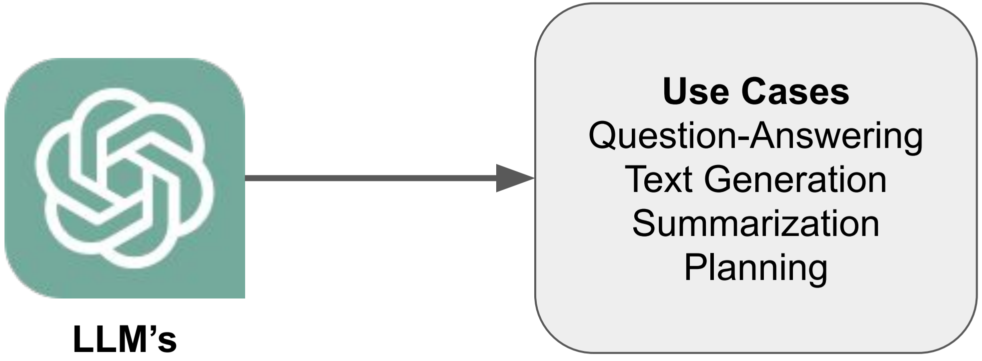
Using LlamaIndex to build Advanced Retrieval in your LLM App

Jerry Liu, LlamaIndex co-founder/CEO

RAG

Context

- LLMs are a phenomenal piece of technology for knowledge generation and reasoning. They are pre-trained on large amounts of **publicly available data**.

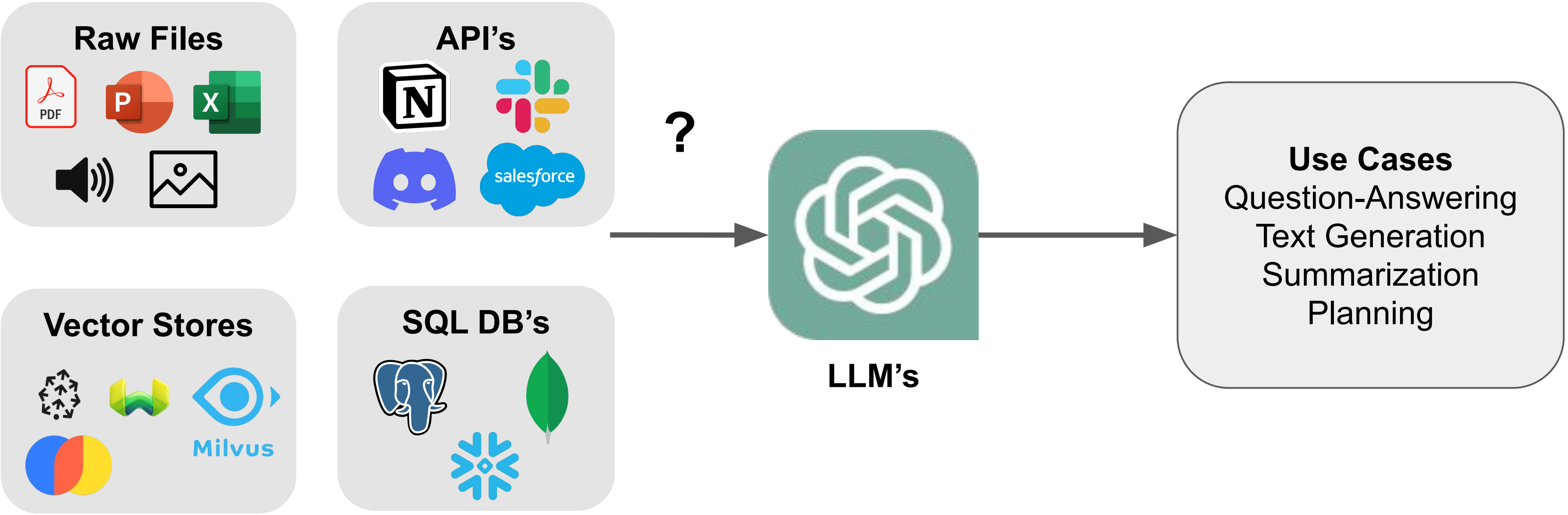


LLM’s

Use Cases
Question-Answering
Text Generation
Summarization
Planning

Context

- How do we best augment LLMs with our own **private data**?



Raw Files

API's

Vector Stores

SQL DB's

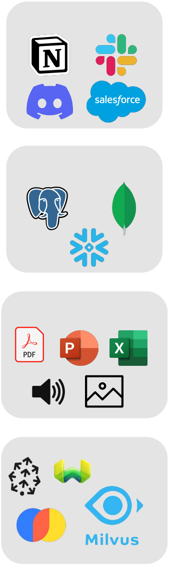
?

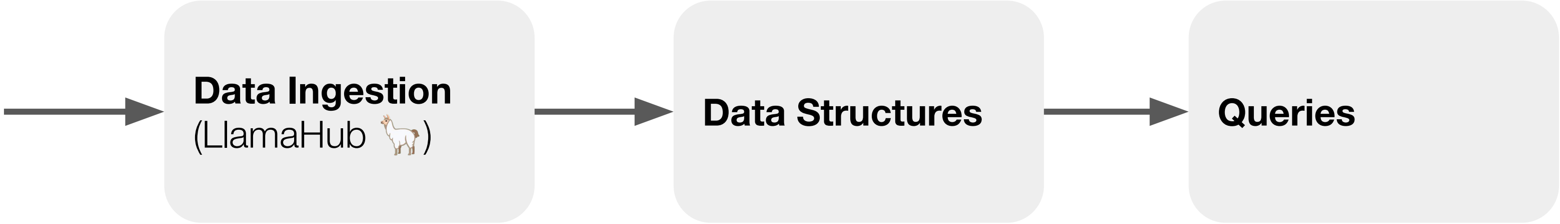
LLM's

Use Cases
Question-Answering
Text Generation
Summarization
Planning

LlamaIndex: A data framework for LLM applications

- Data Management and Query Engine for your LLM application
- Offers components across the data lifecycle: ingest, index, and query over data





Data Ingestion (LlamaHub 🦙)

Data Structures

Queries

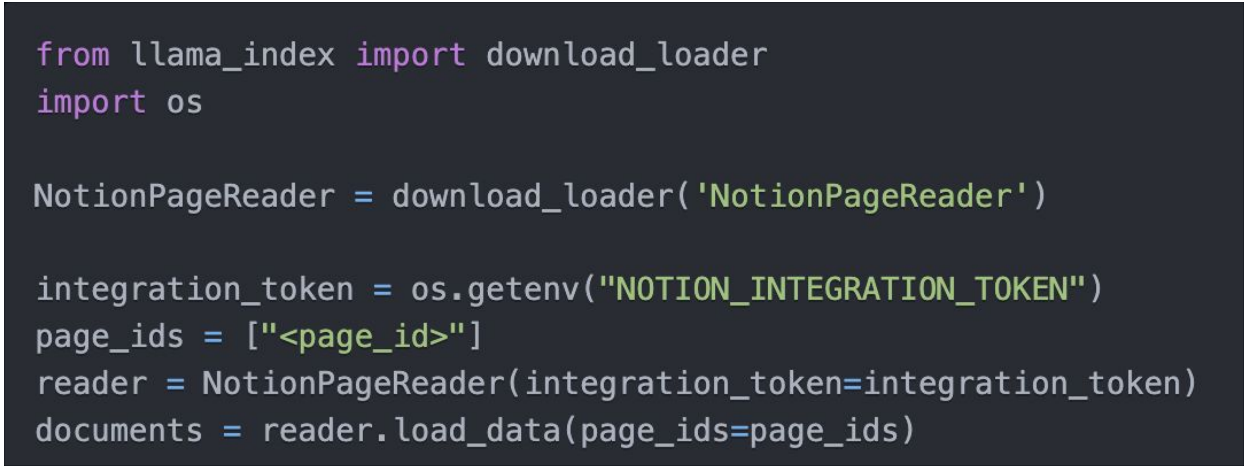
- Connect your existing data sources and data formats (API's, PDF's, docs, SQL, etc.)

- Store and index your data for different use cases. Integrate with different db's (vector db, graph db, kv db)

- Retrieve and query over data
- Includes: QA, Summarization, Agents, and more

Data Connectors: powered by LlamaHub 🦙

- Easily ingest any kind of data, from anywhere
  - into *unified* document containers
- Powered by community-driven hub
  - rapidly growing (100+ loaders and counting!)
- Growing support for multimodal documents (e.g. with inline images)

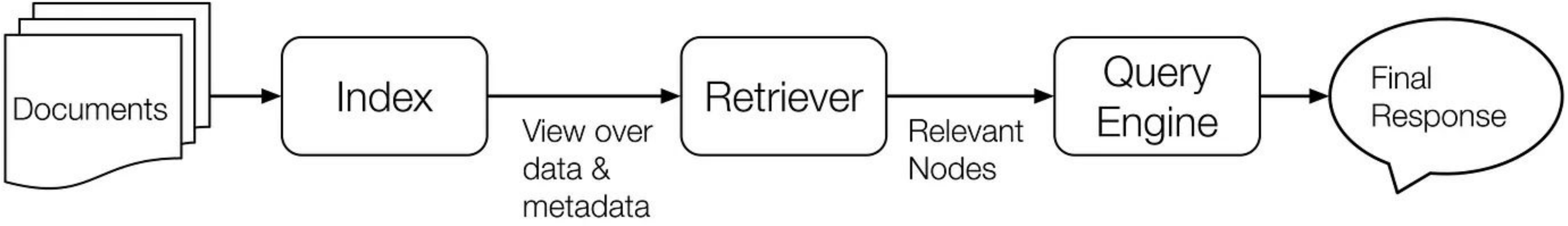


```
from llama_index import download_loader
import os

NotionPageReader = download_loader('NotionPageReader')

integration_token = os.getenv("NOTION_INTEGRATION_TOKEN")
page_ids = ["<page_id>"]
reader = NotionPageReader(integration_token=integration_token)
documents = reader.load_data(page_ids=page_ids)
```

<10 lines of code to ingest from Notion

Doc

Data Indices + Query Interface



Your **source documents** are stored in a data collection

In-memory, MongoDB

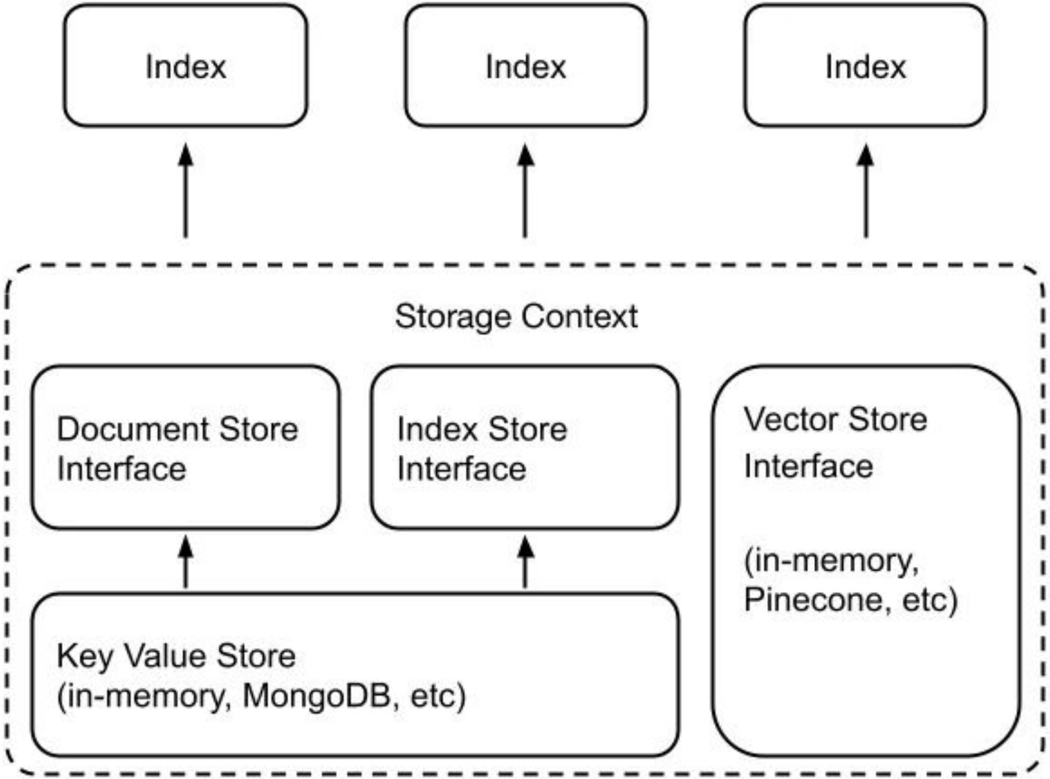
Our **data indices** help to provide a view of your raw data

Vectors, knowledge graphs, keywords

A **retriever** helps to retrieve relevant documents for your query

A **query engine** manages retrieval and synthesis given the query.

Storage Abstractions

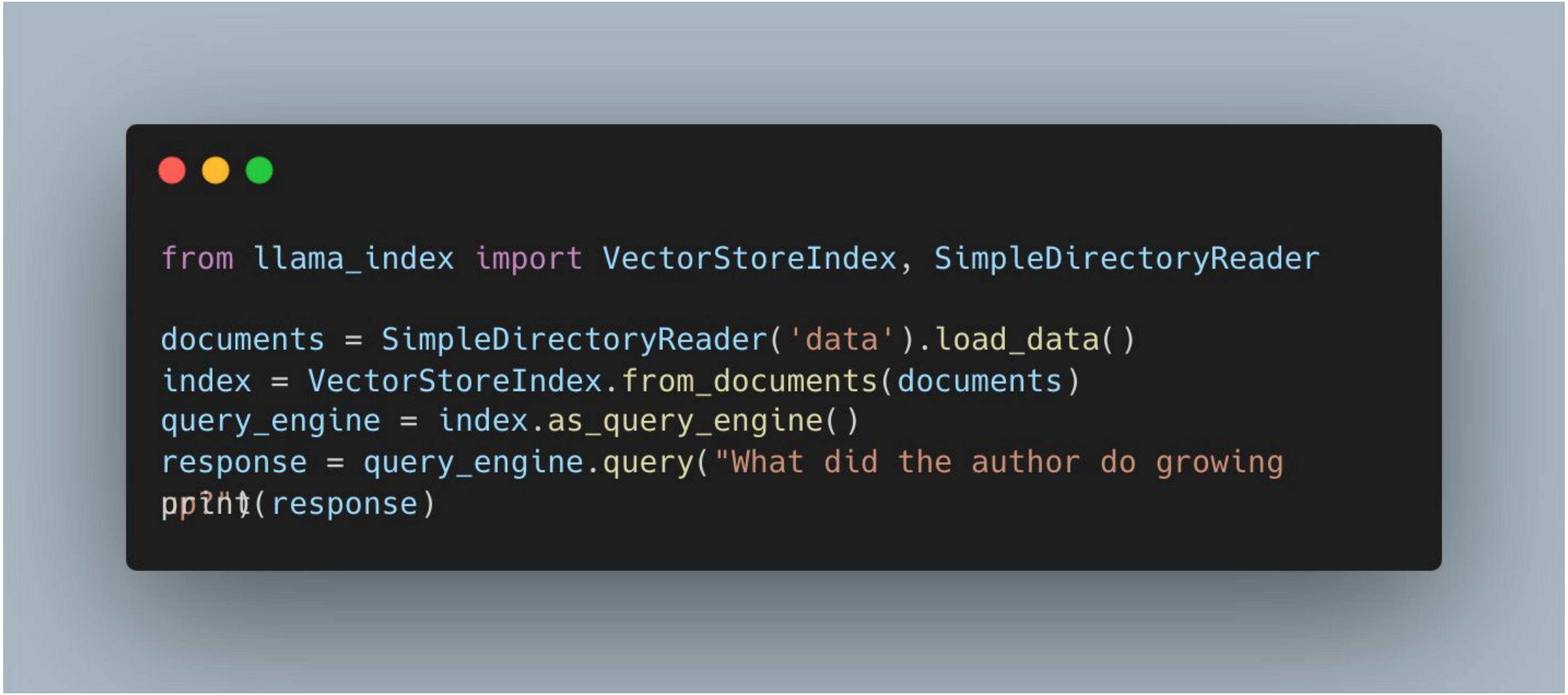


KV Stores:

- In-memory
- MongoDB
- S3

Vector Stores:

- Pinecone
- Weaviate
- Chroma
- Milvus
- Faiss
- Qdrant
- Redis
- Deeplake
- Metal
- DynamoDB
- LanceDB
- Opensearch
- etc.

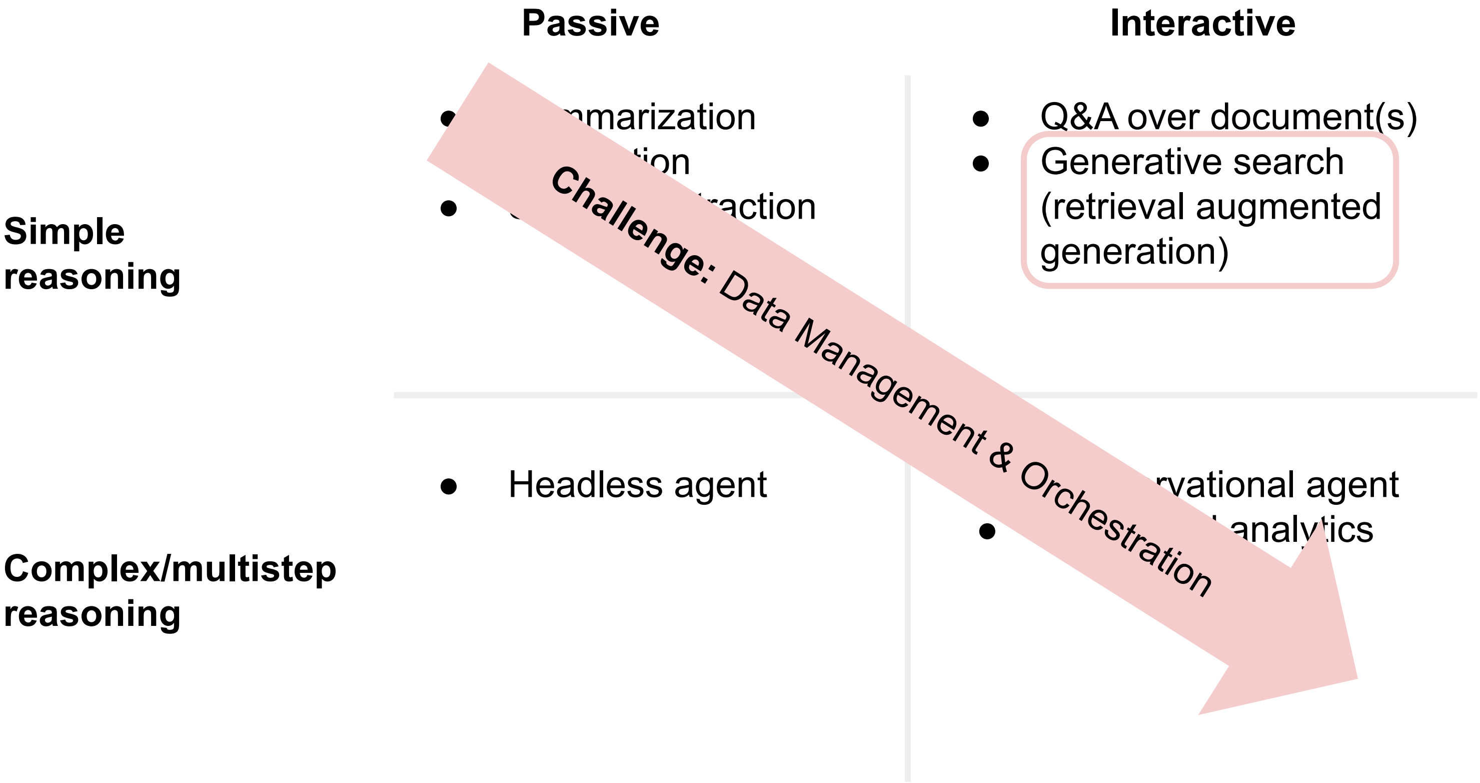


```
from llama_index import VectorStoreIndex, SimpleDirectoryReader

documents = SimpleDirectoryReader('data').load_data()
index = VectorStoreIndex.from_documents(documents)
query_engine = index.as_query_engine()
response = query_engine.query("What did the author do growing
up?")
print(response)
```

LLM Apps over Data (Use Cases)

	Passive	Interactive
Simple reasoning	● Summarization ● Translation ● Schema extraction	● Q&A over document(s) ● Generative search (retrieval augmented generation)
Complex/multistep reasoning	● Headless agent	● Conservational agent ● Structured analytics

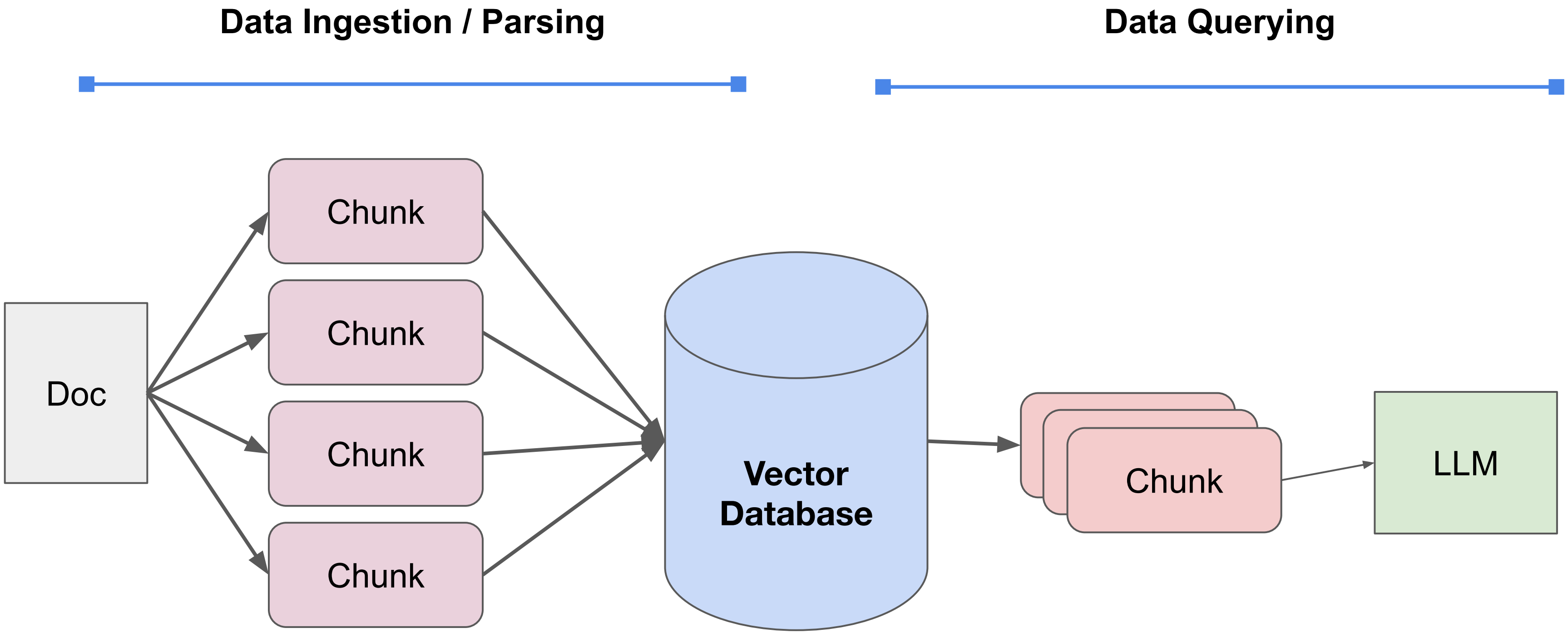
LLM Apps over Data (Use Cases)



	Passive	**Interactive**
Simple reasoning	● mmarization tion ● raction	● Q&A over document(s) ● Generative search (retrieval augmented generation)
Complex/multistep reasoning	● Headless agent	rvational agent ● analy tics

Challenge: Data Management & Orchestration

Naive RAG Stack for building a QA System



Data Ingestion / Parsing

Data Querying

Doc

Chunk

Chunk

Chunk

Chunk

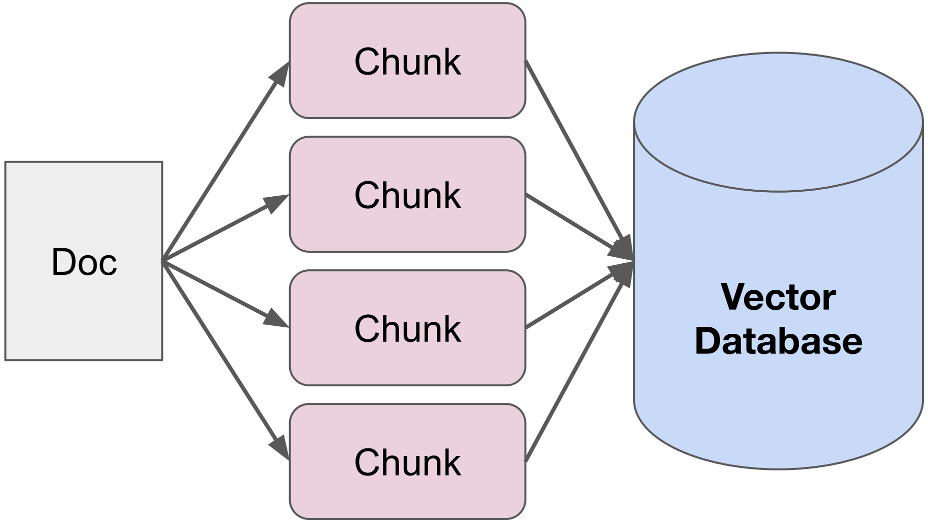
Vector Database

Chunk

LLM

5 Lines of Code in LlamaIndex!

Naive RAG Stack (Data Ingestion/Parsing)



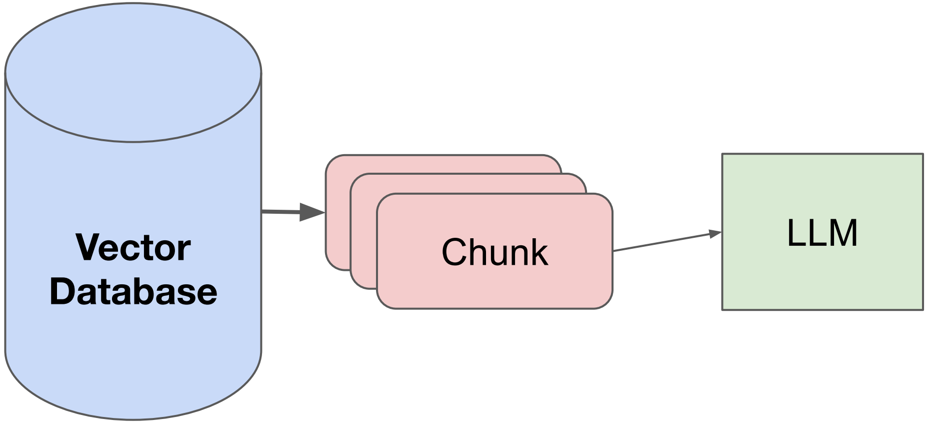
Naive State:

- Split up document(s) into even chunks.
- Each chunk is a piece of raw text.
- All chunks are stored in the same collection in a vector database.

Naive RAG Stack (Querying)

Naive State:

- Find top-k most similar chunks from vector database collection
- Plug into LLM response synthesis module



Challenges with Naive RAG

- **Failure Modes**
    - **Quality-Related (Hallucination, Accuracy)**
    - **Non-Quality-Related (Latency, Cost, Syncing)**

Challenges with Naive RAG (Response Quality)

- The most common reason for bad response quality is bad retrieval
  - If the retrieved results are bad, there’s no way the LLM can synthesize a proper response without hallucinating!

Challenges with Naive RAG (Response Quality)

- Aspects of Bad Retrieval
    - **Low Precision:** Not all chunks in retrieved set are relevant
        - Hallucination + Lost in the Middle Problems
    - **Low Recall:** Now all relevant chunks are retrieved.
        - Lacks enough context for LLM to synthesize an answer
    - **Outdated information:** The data is redundant or out of date.

Challenges with Naive RAG (Other)

- How do you deal with updates in the source document?
  - How do you update stored chunks in the vector database?
- How do you ingest hundreds/thousands of documents?

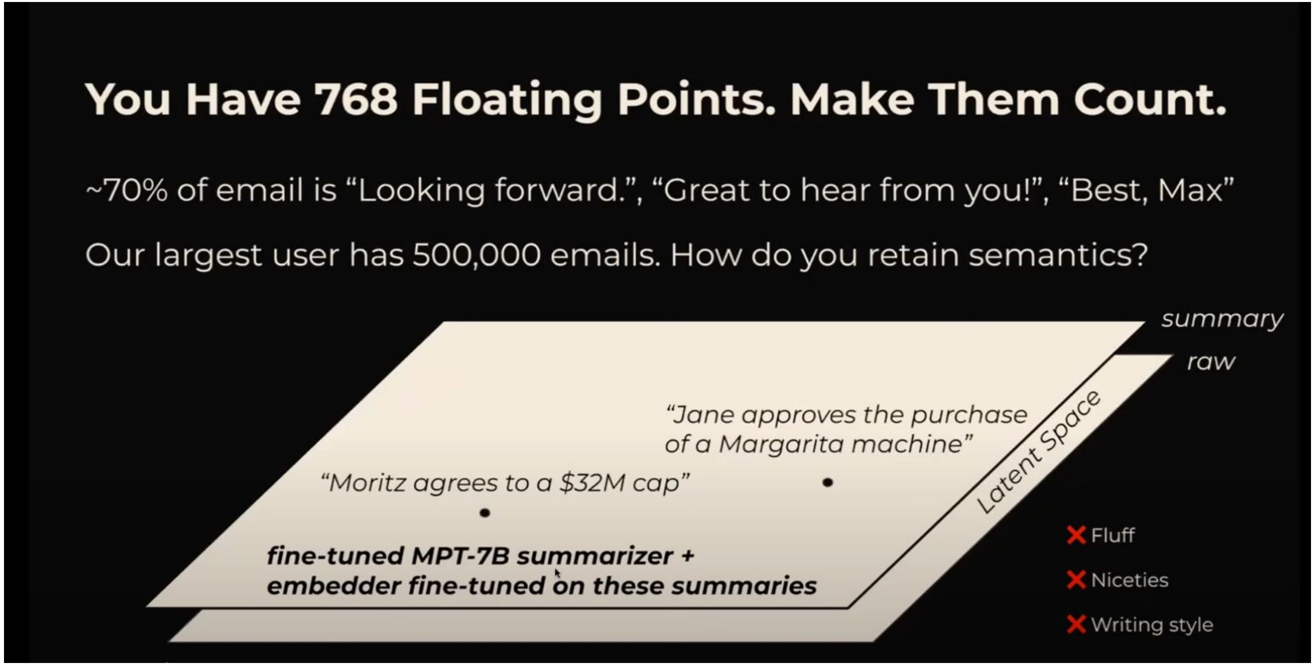
What do we do?

- **Data:** Can we store additional information beyond raw text chunks?
- **Embeddings:** Can we optimize our embedding representations?
- **Retrieval:** Can we do better than top-k embedding lookup?
- **Synthesis:** Can we use LLMs for more than generation?

Techniques for Better Performing RAG

Decouple Embeddings from Raw Text Chunks

Raw text chunks can bias your embedding representation with filler content (Max Rumpf, sid.ai)



You Have 768 Floating Points. Make Them Count.

~70% of email is “Looking forward.”, “Great to hear from you!”, “Best, Max”

Our largest user has 500,000 emails. How do you retain semantics?

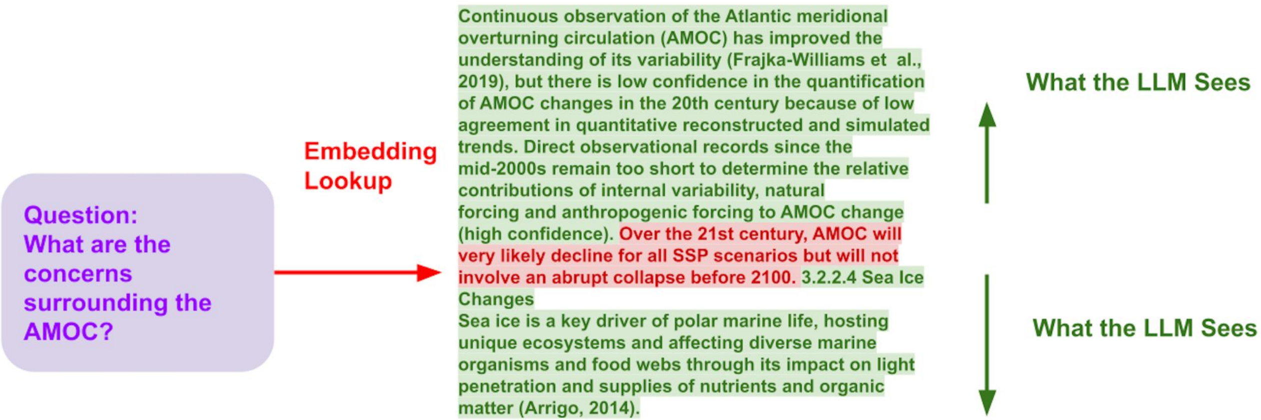
summary
raw
Latent Space
“Jane approves the purchase of a Margarita machine”
“Moritz agrees to a $32M cap”
fine-tuned MPT-7B summarizer + embedder fine-tuned on these summaries
✗ Fluff
✗ Niceties
✗ Writing style

Decouple Embeddings from Raw Text Chunks

Solutions:

- Embed text at the sentence-level - then **expand** that window during LLM synthesis

Embed Sentence → Link to Expanded Window



Question: What are the concerns surrounding the AMOC?

Embedding Lookup

Continuous observation of the Atlantic meridional overturning circulation (AMOC) has improved the understanding of its variability (Frajka-Williams et al., 2019), but there is low confidence in the quantification of AMOC changes in the 20th century because of low agreement in quantitative reconstructed and simulated trends. Direct observational records since the mid-2000s remain too short to determine the relative contributions of internal variability, natural forcing and anthropogenic forcing to AMOC change (high confidence). Over the 21st century, AMOC will very likely decline for all SSP scenarios but will not involve an abrupt collapse before 2100. 3.2.2.4 Sea Ice Changes

Sea ice is a key driver of polar marine life, hosting unique ecosystems and affecting diverse marine organisms and food webs through its impact on light penetration and supplies of nutrients and organic matter (Arrigo, 2014).

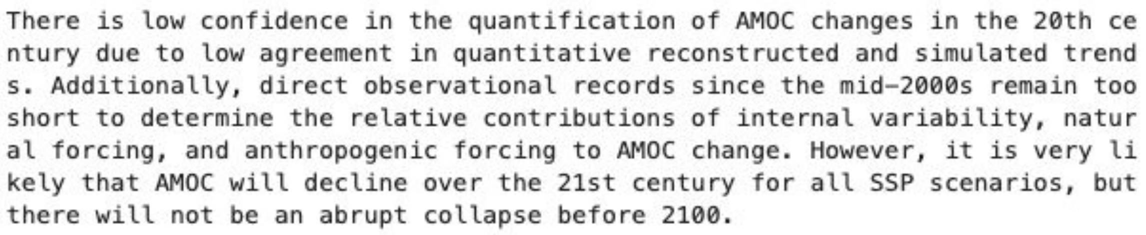
What the LLM Sees

What the LLM Sees

Decouple Embeddings from Raw Text Chunks

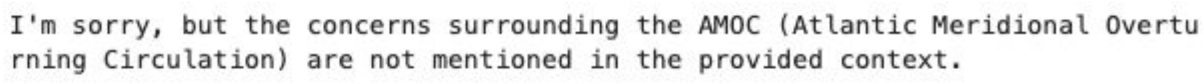
Solutions:

- Embed text at the sentence-level - then **expand** that window during LLM synthesis



```
There is low confidence in the quantification of AMOC changes in the 20th ce
ntury due to low agreement in quantitative reconstructed and simulated trend
s. Additionally, direct observational records since the mid-2000s remain too
short to determine the relative contributions of internal variability, natur
al forcing, and anthropogenic forcing to AMOC change. However, it is very li
kely that AMOC will decline over the 21st century for all SSP scenarios, but
there will not be an abrupt collapse before 2100.
```

Sentence Window Retrieval (k=2)



```
I'm sorry, but the concerns surrounding the AMOC (Atlantic Meridional Overtu
rning Circulation) are not mentioned in the provided context.
```

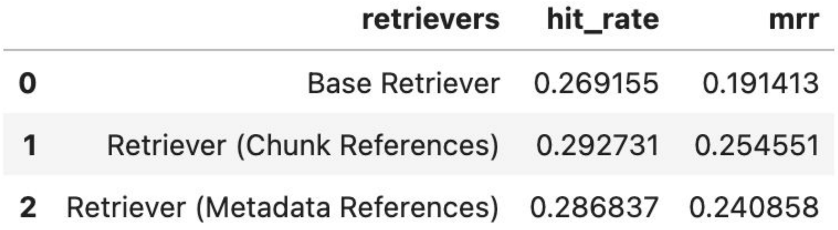
Naive Retrieval (k=5)

Only one out of the 5 chunks is relevant - "lost in the middle" problem

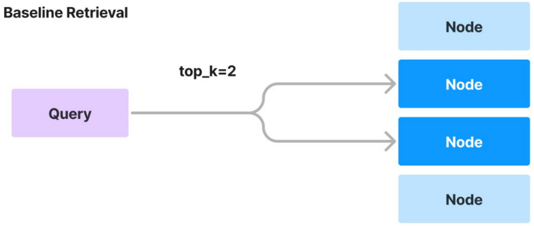
Decouple Embeddings from Raw Text Chunks

Solutions:

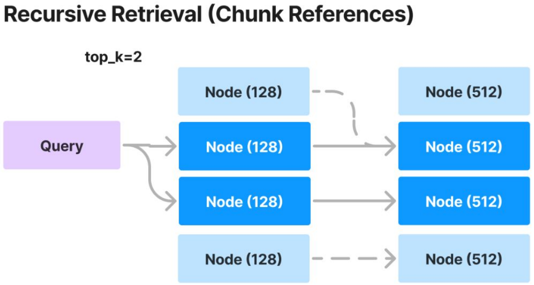
- Embed “**references**” to text chunks instead of the text chunks directly.
- Examples
  - Smaller Chunks
  - Metadata
  - Summaries
- Retrieve those references first, then the text chunks.



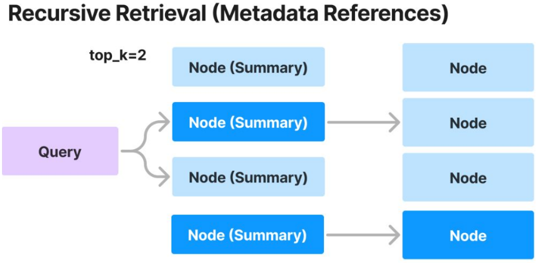
	retrievers	hit_rate	mrr
0	Base Retriever	0.269155	0.191413
1	Retriever (Chunk References)	0.292731	0.254551
2	Retriever (Metadata References)	0.286837	0.240858



Baseline Retrieval



Recursive Retrieval (Chunk References)

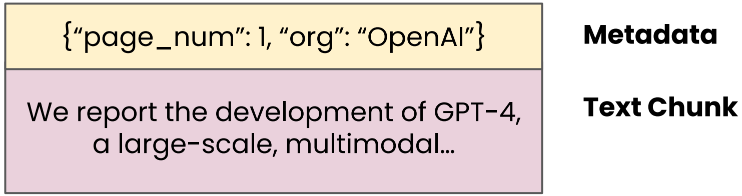


Recursive Retrieval (Metadata References)

Organize your data for more structured retrieval (Metadata)

- **Metadata:** context you can inject into each text chunk
- Examples
  - Page number
  - Document title
  - Summary of adjacent chunks
  - Questions that chunk can answer (reverse HyDE)
- **Benefits**
  - **Can Help Retrieval**
  - **Can Augment Response Quality**
  - **Integrates with Vector DB Metadata Filters**

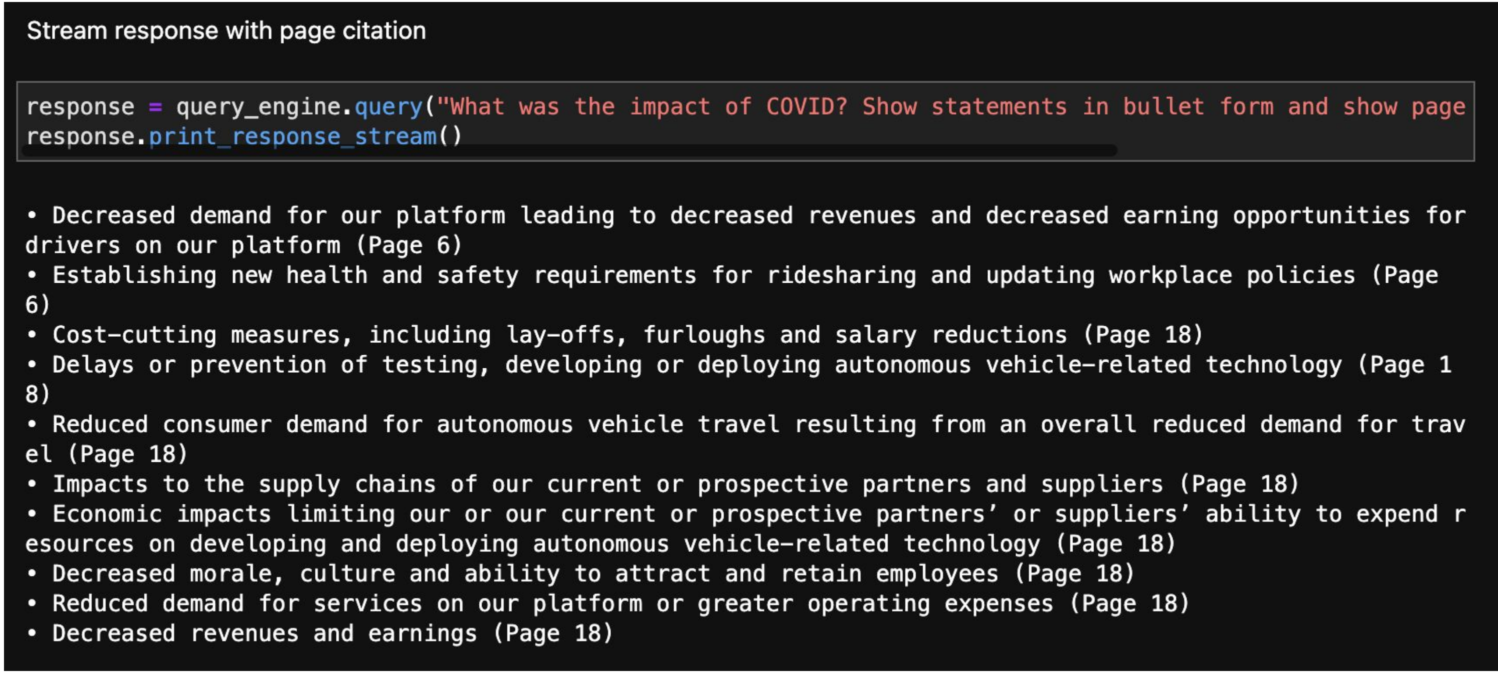
Example of Metadata



{“page_num”: 1, “org”: “OpenAI”}

We report the development of GPT-4, a large-scale, multimodal…

Simple use case: adding page numbers to PDF’s allows for in-line citations

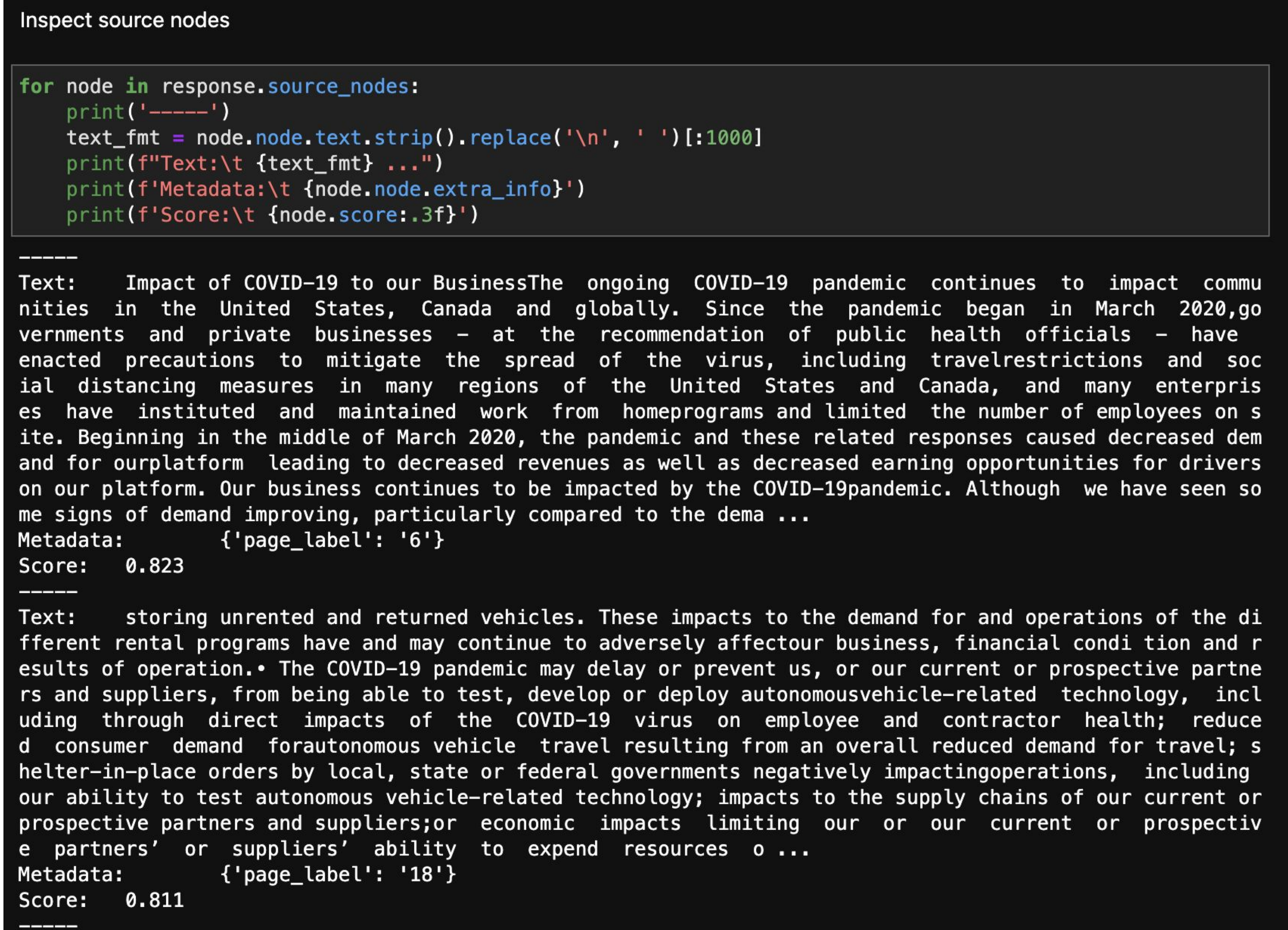


Stream response with page citation

```
response = query_engine.query("What was the impact of COVID? Show statements in bullet form and show page
response.print_response_stream()
```

```
• Decreased demand for our platform leading to decreased revenues and decreased earning opportunities for
drivers on our platform (Page 6)
• Establishing new health and safety requirements for ridesharing and updating workplace policies (Page
6)
• Cost-cutting measures, including lay-offs, furloughs and salary reductions (Page 18)
• Delays or prevention of testing, developing or deploying autonomous vehicle-related technology (Page 1
8)
• Reduced consumer demand for autonomous vehicle travel resulting from an overall reduced demand for trav
el (Page 18)
• Impacts to the supply chains of our current or prospective partners and suppliers (Page 18)
• Economic impacts limiting our or our current or prospective partners’ or suppliers’ ability to expend r
esources on developing and deploying autonomous vehicle-related technology (Page 18)
• Decreased morale, culture and ability to attract and retain employees (Page 18)
• Reduced demand for services on our platform or greater operating expenses (Page 18)
• Decreased revenues and earnings (Page 18)
```

Simple use case: adding page numbers to PDF's allows for in-line citations

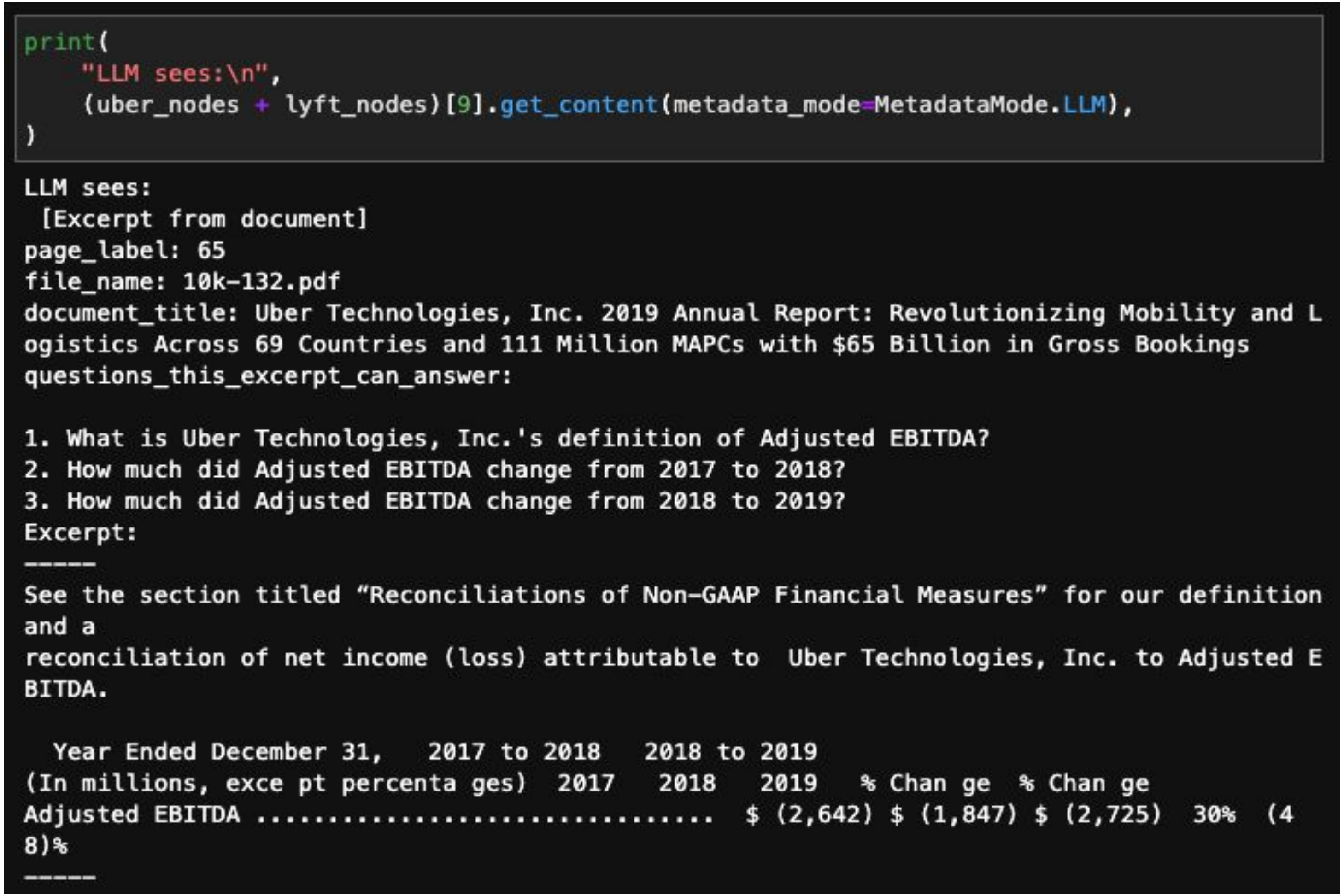


Inspect source nodes

```python
for node in response.source_nodes:
    print('-----')
    text_fmt = node.node.text.strip().replace('\n', ' ')[:1000]
    print(f"Text:\t {text_fmt} ...")
    print(f'Metadata:\t {node.node.extra_info}')
    print(f'Score:\t {node.score:.3f}')
```

```
-----
Text:    Impact of COVID-19 to our BusinessThe ongoing COVID-19 pandemic continues to impact commu
nities in the United States, Canada and globally. Since the pandemic began in March 2020,go
vernments and private businesses - at the recommendation of public health officials - have
enacted precautions to mitigate the spread of the virus, including travelrestrictions and soc
ial distancing measures in many regions of the United States and Canada, and many enterpris
es have instituted and maintained work from homeprograms and limited the number of employees on s
ite. Beginning in the middle of March 2020, the pandemic and these related responses caused decreased dem
and for ourplatform leading to decreased revenues as well as decreased earning opportunities for drivers
on our platform. Our business continues to be impacted by the COVID-19pandemic. Although we have seen so
me signs of demand improving, particularly compared to the dema ...
Metadata:        {'page_label': '6'}
Score:   0.823
-----
Text:    storing unrented and returned vehicles. These impacts to the demand for and operations of the di
fferent rental programs have and may continue to adversely affectour business, financial condi tion and r
esults of operation.• The COVID-19 pandemic may delay or prevent us, or our current or prospective partne
rs and suppliers, from being able to test, develop or deploy autonomousvehicle-related technology, incl
uding through direct impacts of the COVID-19 virus on employee and contractor health; reduce
d consumer demand forautonomous vehicle travel resulting from an overall reduced demand for travel; s
helter-in-place orders by local, state or federal governments negatively impactingoperations, including
our ability to test autonomous vehicle-related technology; impacts to the supply chains of our current or
prospective partners and suppliers;or economic impacts limiting our or our current or prospectiv
e partners' or suppliers' ability to expend resources o ...
Metadata:        {'page_label': '18'}
Score:   0.811
-----
```

Using LLMs for Automatic Metadata Extraction



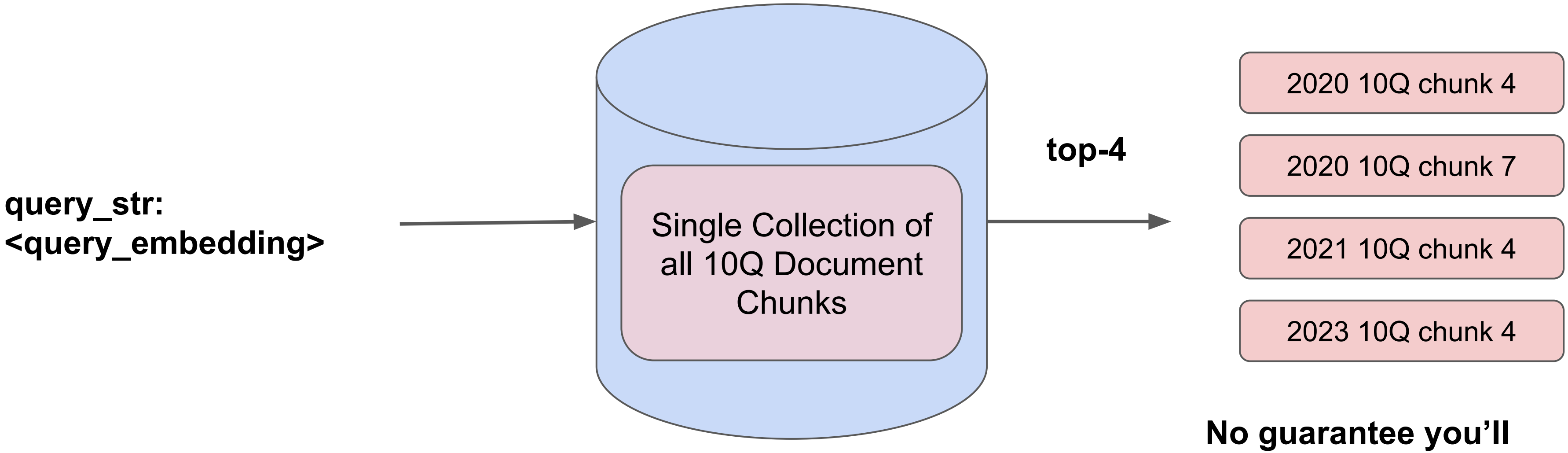
```
print(
    "LLM sees:\n",
    (uber_nodes + lyft_nodes)[9].get_content(metadata_mode=MetadataMode.LLM),
)
```

```
LLM sees:
 [Excerpt from document]
page_label: 65
file_name: 10k-132.pdf
document_title: Uber Technologies, Inc. 2019 Annual Report: Revolutionizing Mobility and L
ogistics Across 69 Countries and 111 Million MAPCs with $65 Billion in Gross Bookings
questions_this_excerpt_can_answer:

1. What is Uber Technologies, Inc.'s definition of Adjusted EBITDA?
2. How much did Adjusted EBITDA change from 2017 to 2018?
3. How much did Adjusted EBITDA change from 2018 to 2019?
Excerpt:
-----
See the section titled “Reconciliations of Non-GAAP Financial Measures” for our definition
and a
reconciliation of net income (loss) attributable to  Uber Technologies, Inc. to Adjusted E
BITDA.

  Year Ended December 31,   2017 to 2018   2018 to 2019
(In millions, exce pt percenta ges)  2017   2018   2019   % Chan ge  % Chan ge
Adjusted EBITDA ................................  $ (2,642) $ (1,847) $ (2,725)  30%  (4
8)%
-----
```

Organize your data for more structured retrieval (Metadata Filters)

Question: “Can you tell me about Google’s R&D initiatives from 2020 to 2023?”

- Dumping chunks to a single collection doesn’t work.

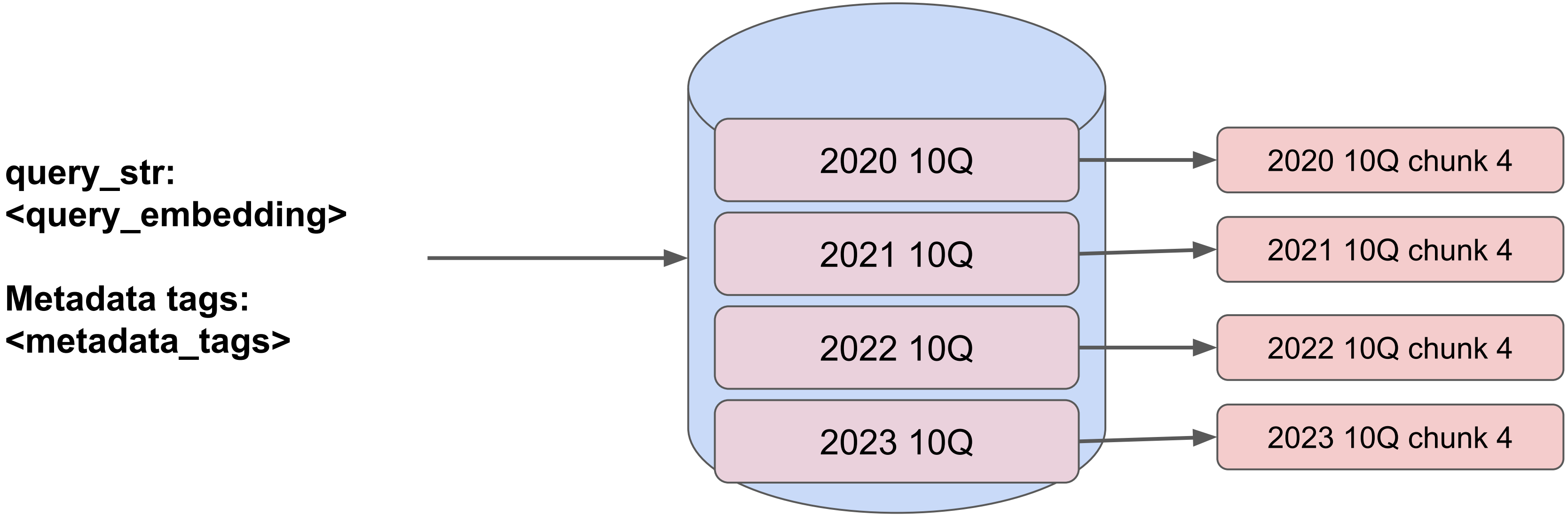


No guarantee you’ll return the relevant document chunks!

Organize your data for more structured retrieval (Metadata Filters)

Question: “Can you tell me about Google’s R&D initiatives from 2020 to 2023?”

- Here, we separate and tag the documents with **metadata filters**.
- During query-time, we can *infer* these metadata filters in addition to semantic query.



query_str: <query_embedding>

Metadata tags: <metadata_tags>

2020 10Q

2021 10Q

2022 10Q

2023 10Q

2020 10Q chunk 4

2021 10Q chunk 4

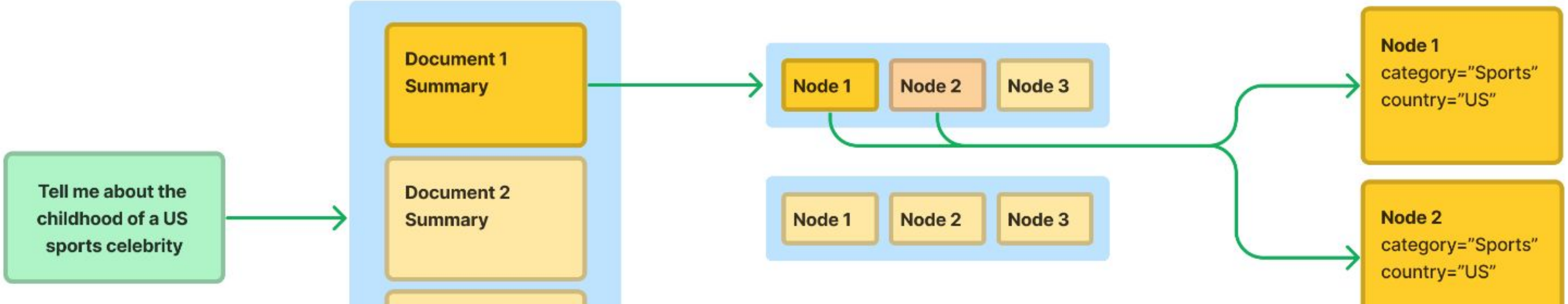
2022 10Q chunk 4

2023 10Q chunk 4

Organize your data for more structured retrieval (Recursive Retrieval)

- Organize your data **hierarchically**
  - Summaries → documents
  - Documents → embedded objects (Tables/Graphs)

Document Hierarchies (Summaries + Raw Chunks) + Recursive Retrieval

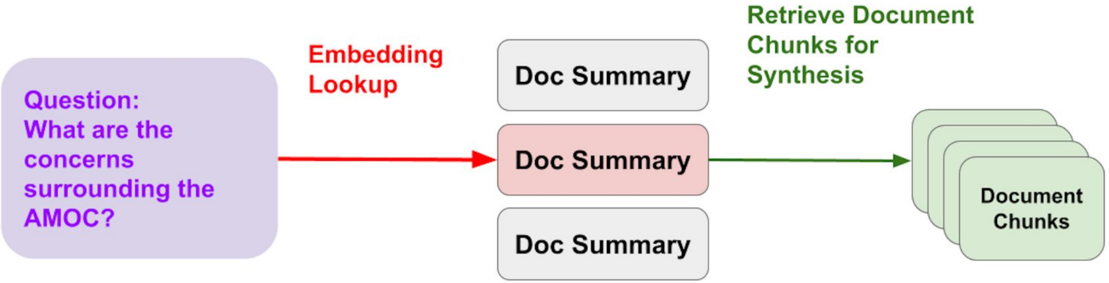

Tell me about the childhood of a US sports celebrity
Document 1 Summary
Document 2 Summary
Node 1
Node 2
Node 3
Node 1
Node 2
Node 3
Node 1 category="Sports" country="US"
Node 2 category="Sports" country="US"

Organize your data for more structured retrieval (Recursive Retrieval)

Summaries → documents

- Embed larger documents via **summaries.** First retrieve documents by summaries, then retrieve chunks within those documents

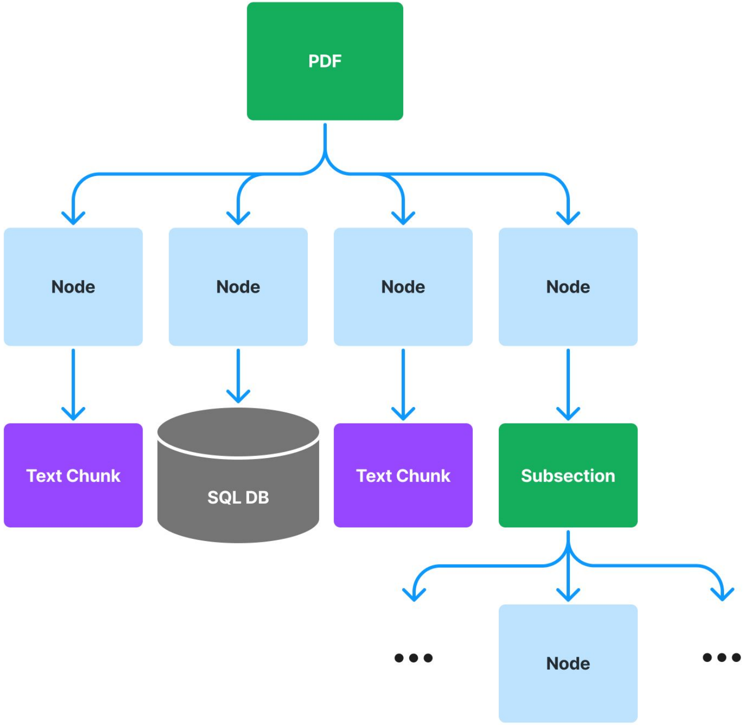
Embed Summary → Link to Additional Documents


Question: What are the concerns surrounding the AMOC?
Embedding Lookup
Doc Summary
Doc Summary
Doc Summary
Retrieve Document Chunks for Synthesis
Document Chunks

Organize your data for more structured retrieval (Recursive Retrieval)

Documents → Embedded Objects

- If you have embedded objects in your PDF documents (tables, graphs), first retrieve entities by a **reference object,** then query the underlying object.



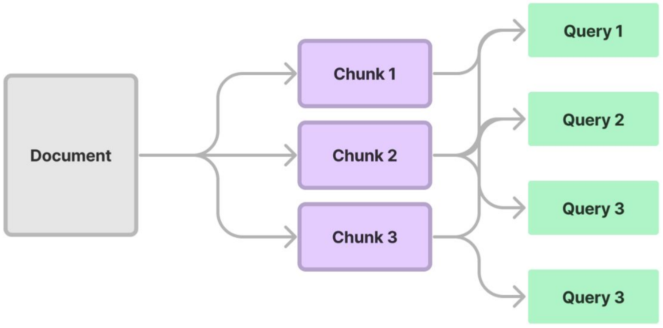
Production RAG Guide

https://gpt-index.readthedocs.io/en/latest/end_to_end_tutorials/dev_practices/production_rag.html

Evaluation

Synthetic Dataset Generation for Retrieval Evals

1. Parse / chunk up text corpus
2. Prompt GPT-4 to generate questions from each chunk (or subset of chunks)
3. Each (question, chunk) is now your evaluation pair!
4. Run question through retriever. Compare against ground-truth with ranking metrics.



LLM-generated Synthetic Queries

Query 1	Ground-Truth: 1, 2
Query 2	Ground-Truth: 2, 3
Query 3	Ground-Truth: 2
Query 3	Ground-Truth: 3

Dataset (for Evaluation or Finetuning)

Fine-Tuning

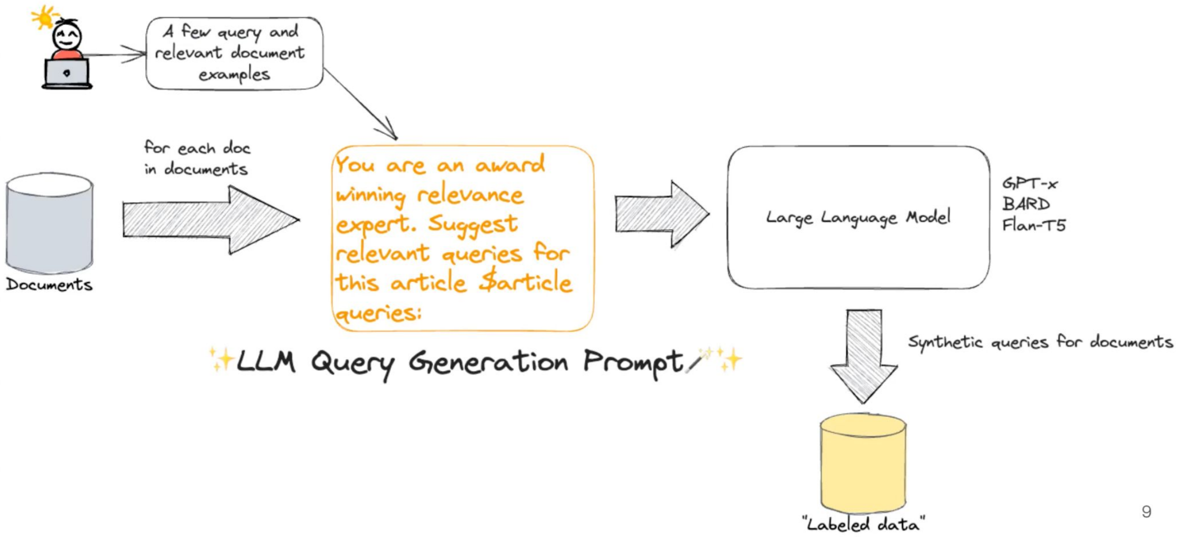
Fine-tuning

You can choose to fine-tune the **embeddings** or the **LLM**

Fine-tuning (Embeddings)

Generate a synthetic query dataset from raw text chunks using LLMs

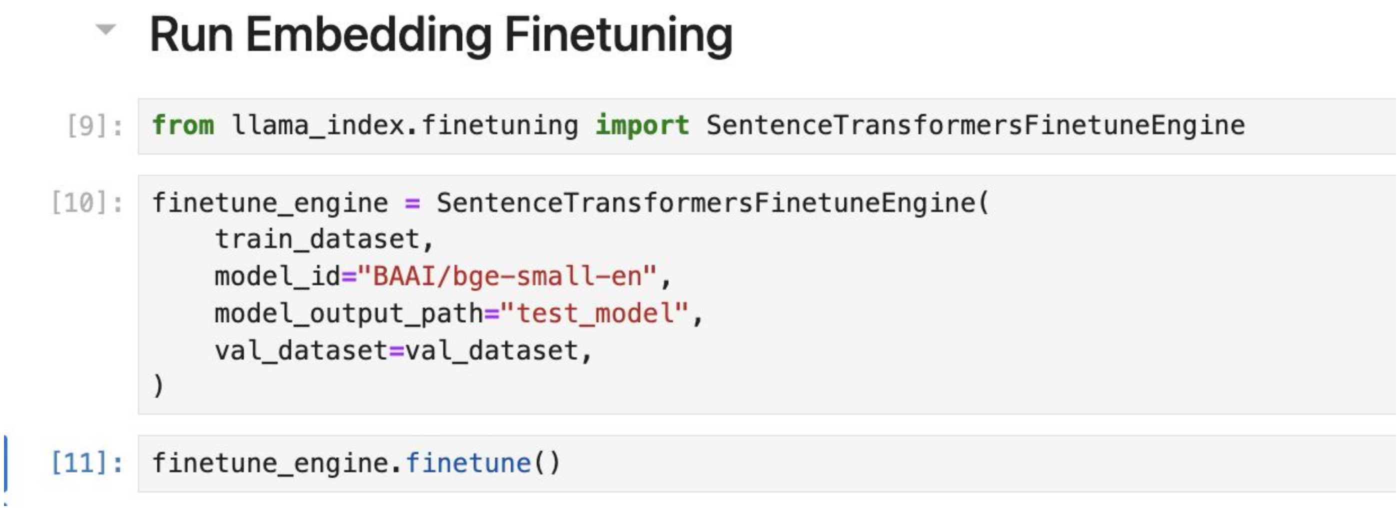
The gist of using LLMs to generate labeled data


A few query and relevant document examples
for each doc in documents
Documents
You are an award winning relevance expert. Suggest relevant queries for this article $article queries:
✨LLM Query Generation Prompt🪄✨
Large Language Model
GPT-x
BARD
Flan-T5
Synthetic queries for documents
"Labeled data"

Fine-tuning (Embeddings)

Use this synthetic dataset to finetune an embedding model.

- Directly finetune sentence_transformers model
- Finetune a black-box adapter (linear, NN, any neural network)



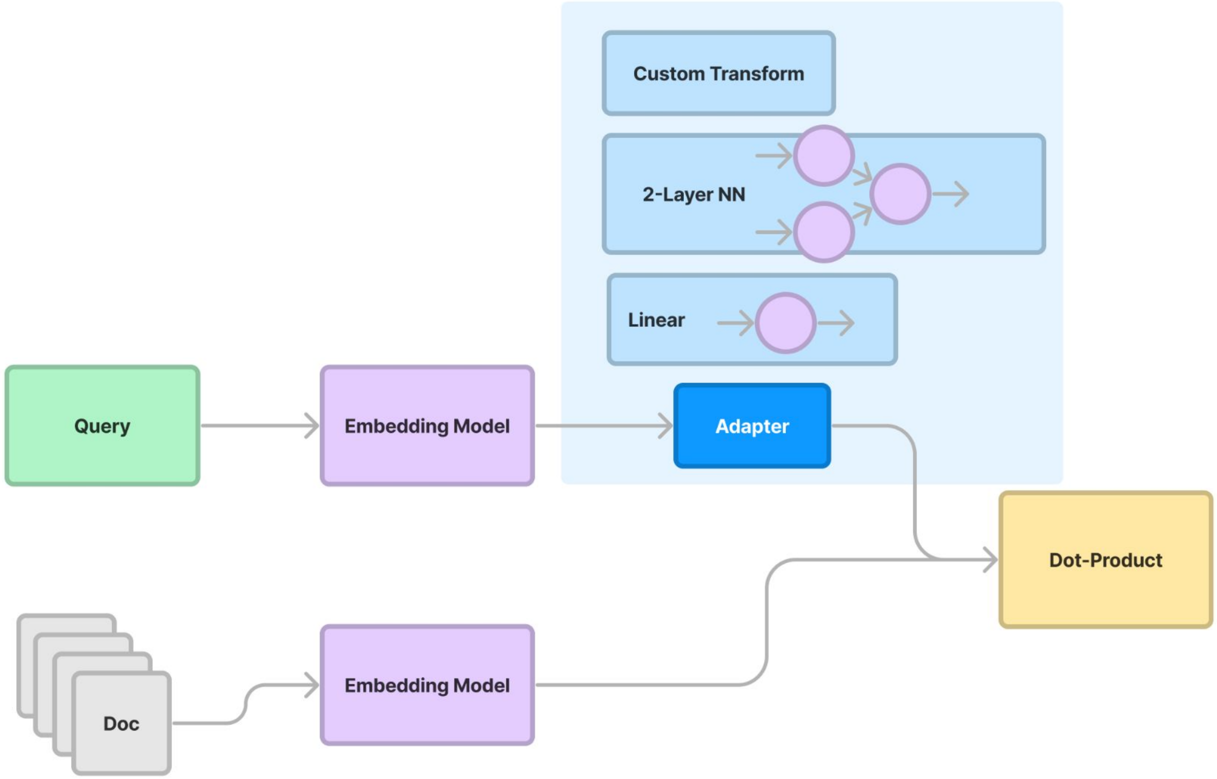
Run Embedding Finetuning

```
[9]: from llama_index.finetuning import SentenceTransformersFinetuneEngine
```

```
[10]: finetune_engine = SentenceTransformersFinetuneEngine(
          train_dataset,
          model_id="BAAI/bge-small-en",
          model_output_path="test_model",
          val_dataset=val_dataset,
      )
```

```
[11]: finetune_engine.finetune()
```

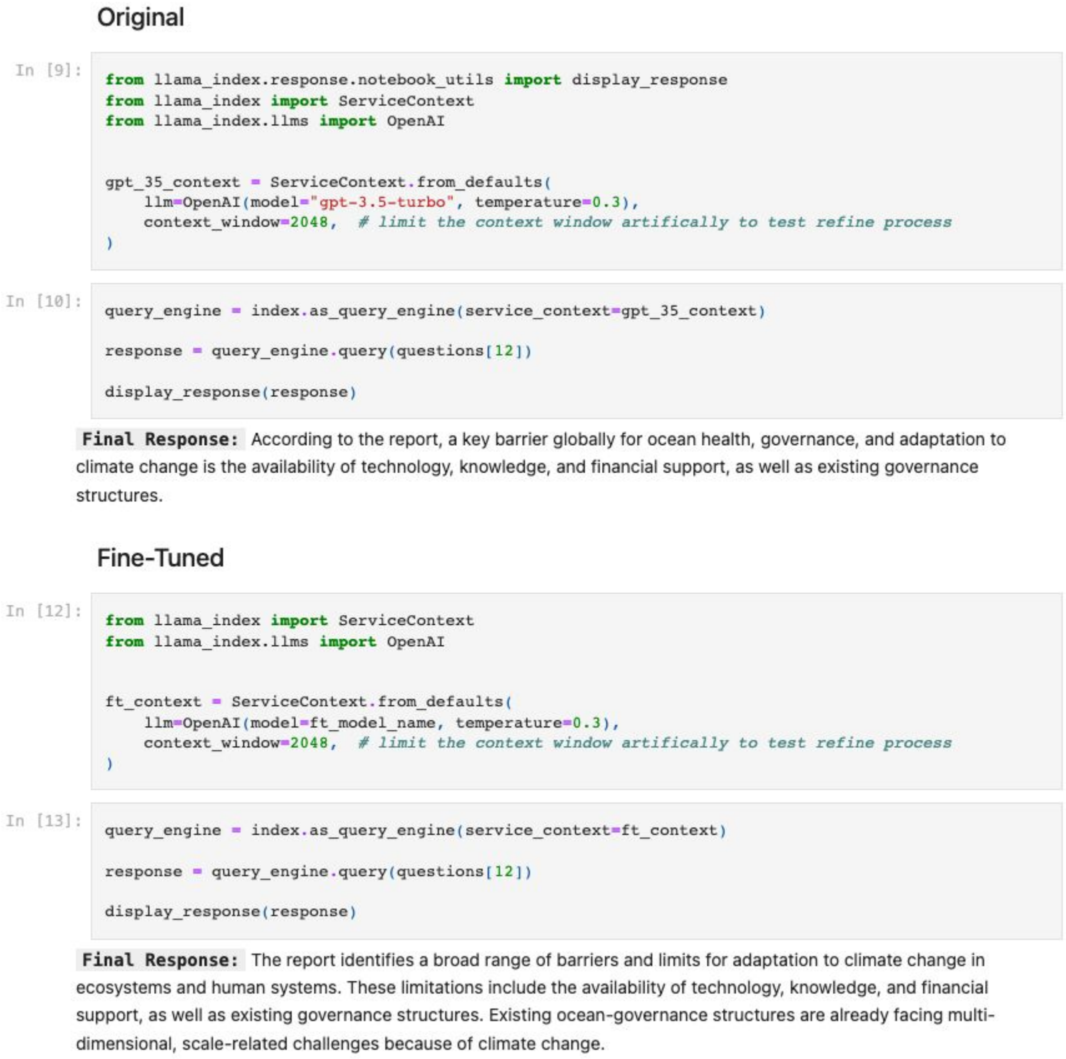
Fine-tuning a Black-box Adapter



Fine-tuning (LLMs)

Use OpenAI to distill GPT-4 to gpt-3.5-turbo

- Final response generation
- Agent intermediate reasoning



Original

```
from llama_index.response.notebook_utils import display_response
from llama_index import ServiceContext
from llama_index.llms import OpenAI


gpt_35_context = ServiceContext.from_defaults(
    llm=OpenAI(model="gpt-3.5-turbo", temperature=0.3),
    context_window=2048,  # limit the context window artifically to test refine process
)
```

```
query_engine = index.as_query_engine(service_context=gpt_35_context)

response = query_engine.query(questions[12])

display_response(response)
```

Final Response: According to the report, a key barrier globally for ocean health, governance, and adaptation to climate change is the availability of technology, knowledge, and financial support, as well as existing governance structures.

Fine-Tuned

```
from llama_index import ServiceContext
from llama_index.llms import OpenAI


ft_context = ServiceContext.from_defaults(
    llm=OpenAI(model=ft_model_name, temperature=0.3),
    context_window=2048,  # limit the context window artifically to test refine process
)
```

```
query_engine = index.as_query_engine(service_context=ft_context)

response = query_engine.query(questions[12])

display_response(response)
```

Final Response: The report identifies a broad range of barriers and limits for adaptation to climate change in ecosystems and human systems. These limitations include the availability of technology, knowledge, and financial support, as well as existing governance structures. Existing ocean-governance structures are already facing multi-dimensional, scale-related challenges because of climate change.

Finetuning Abstractions in LlamaIndex

https://gpt-index.readthedocs.io/en/latest/end_to_end_tutorials/finetuning.html

Notebook Walkthroughs

Quickstart Walkthrough

https://colab.research.google.com/drive/1knQpGJLHj-LTTHqlZhgcjDH5F_nJIiY0?usp=sharing

Advanced Retrieval: Node References

https://colab.research.google.com/drive/1JazWHjk-_KWm-_o0pcRtpwtJ8TwFu2aH?usp=sharing