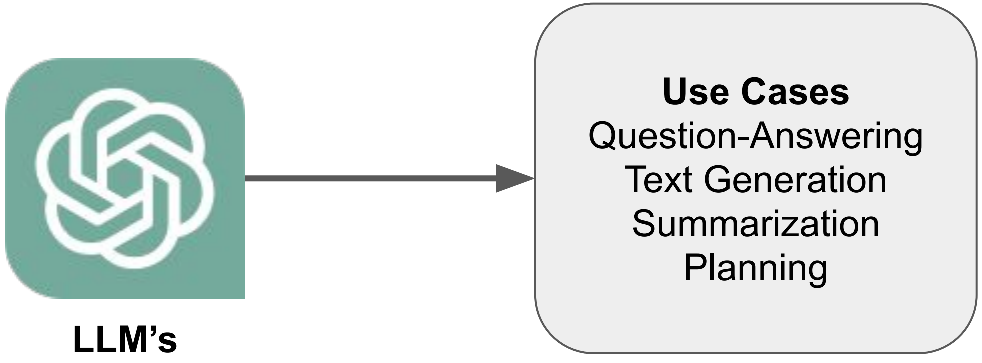
Building RAG with LlamaIndex (+Tips/Tricks!)

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**.

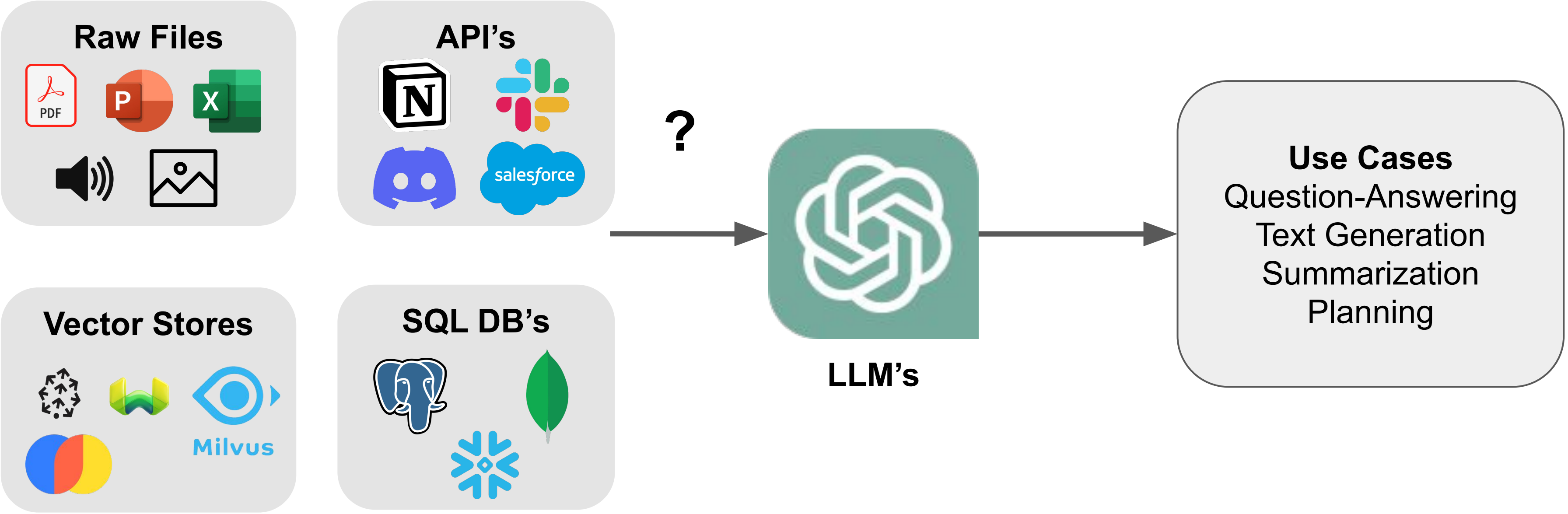


LLM’s

Use Cases
Question-Answering
Text Generation
Summarization
Planning

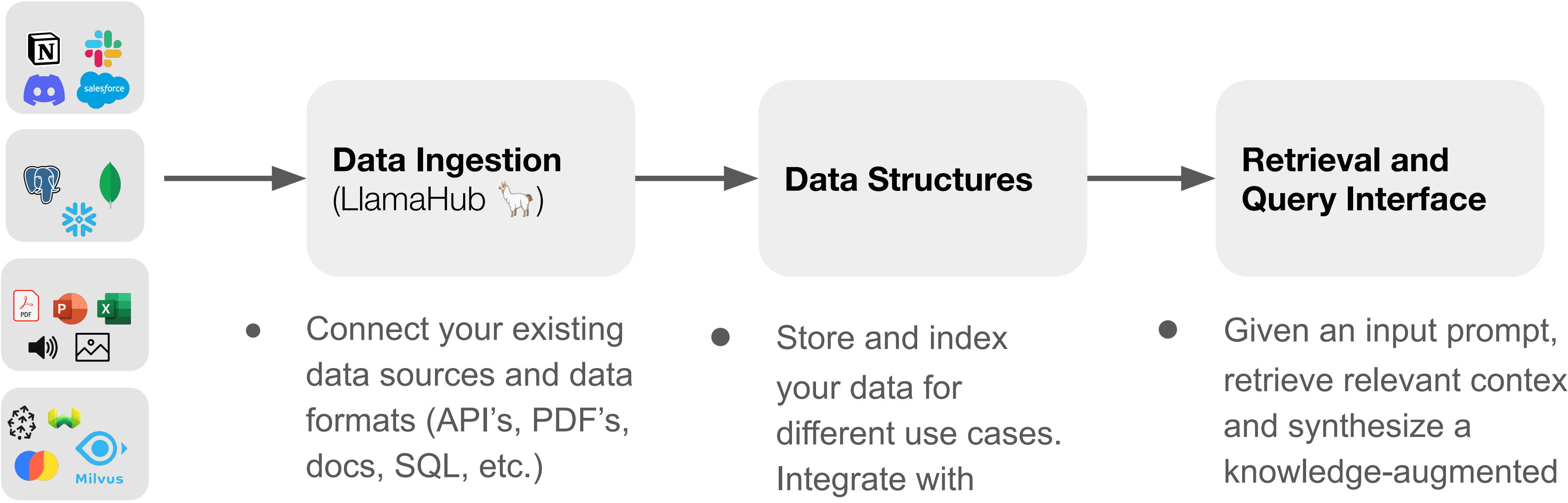
Context

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



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 🦙)

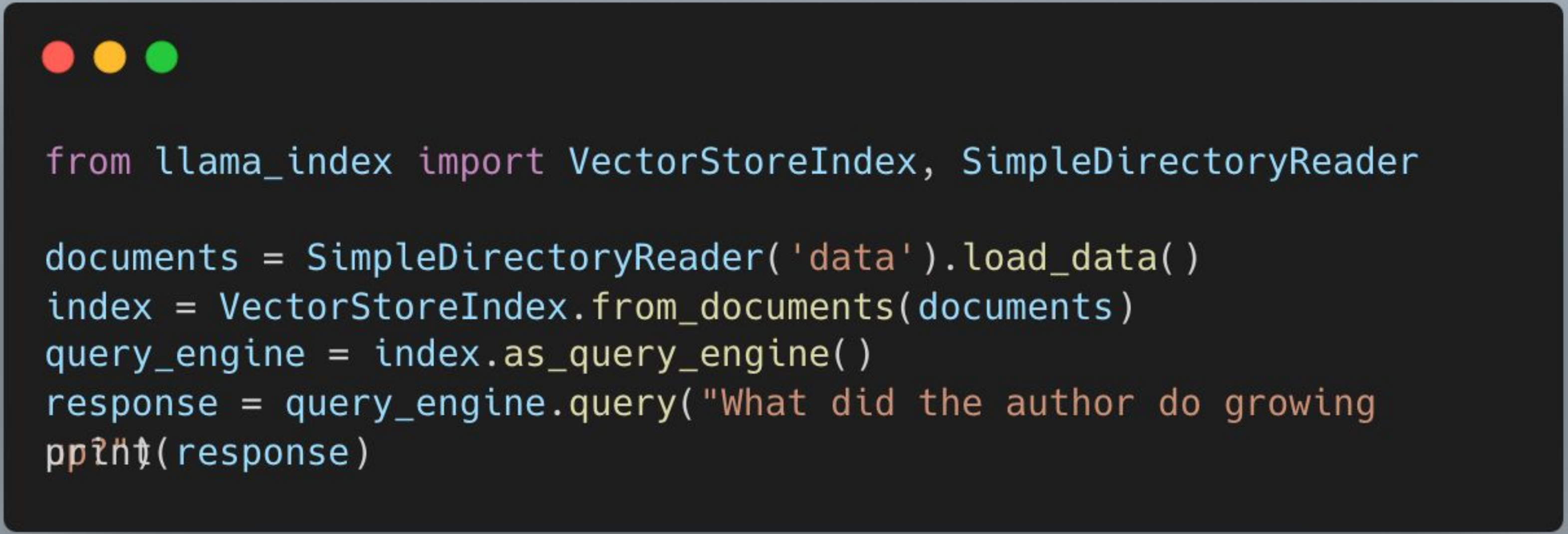
Data Structures

Retrieval and Query Interface

- 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.

- Given an input prompt, retrieve relevant context and synthesize a knowledge-augmented output.



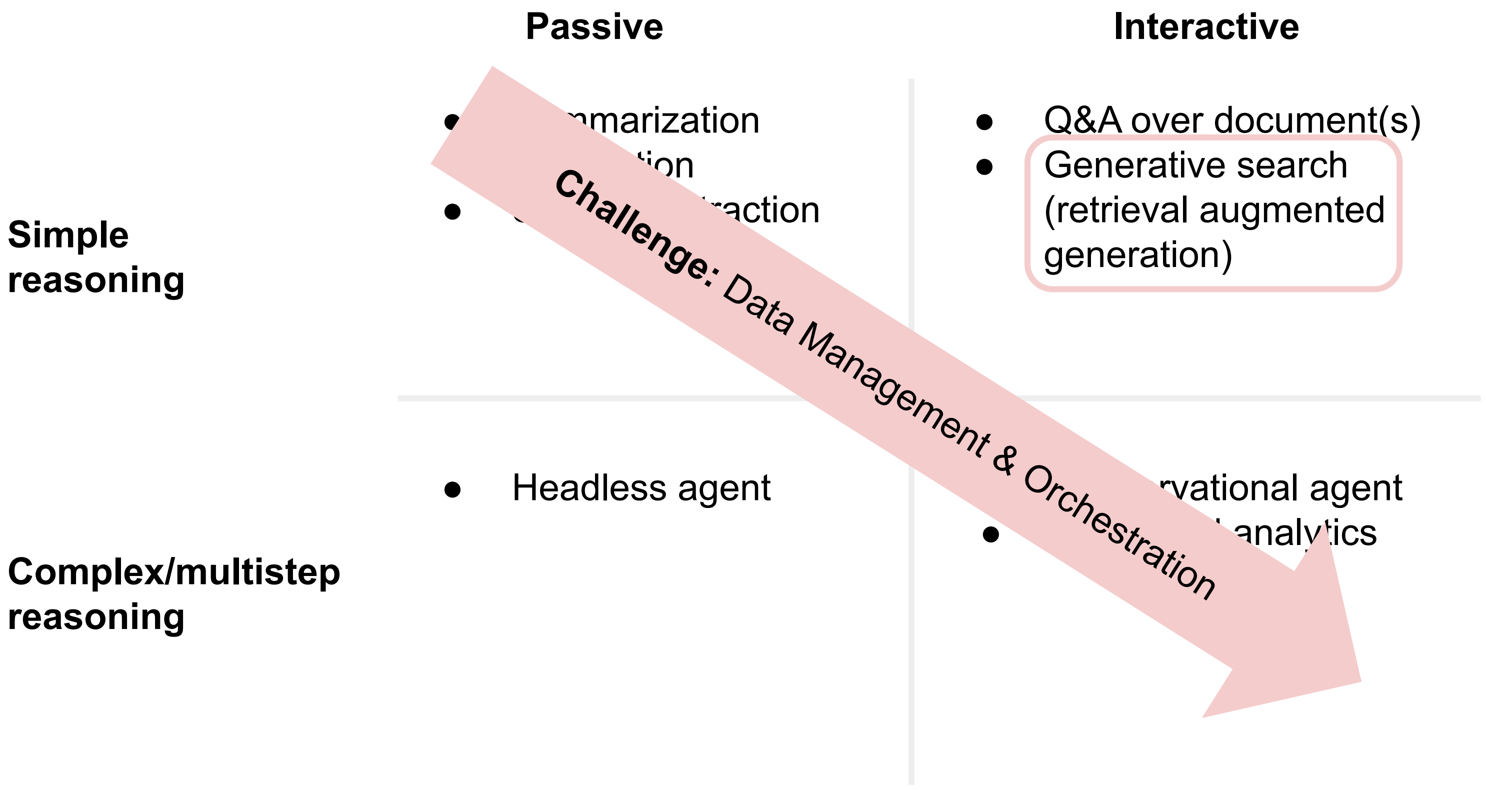
```
from llama_index import VectorStoreIndex, SimpleDirectoryReader

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 App 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 App Use Cases



Passive

Interactive

Simple reasoning

- mmarization
  tion
- raction

- Q&A over document(s)
- Generative search (retrieval augmented generation)

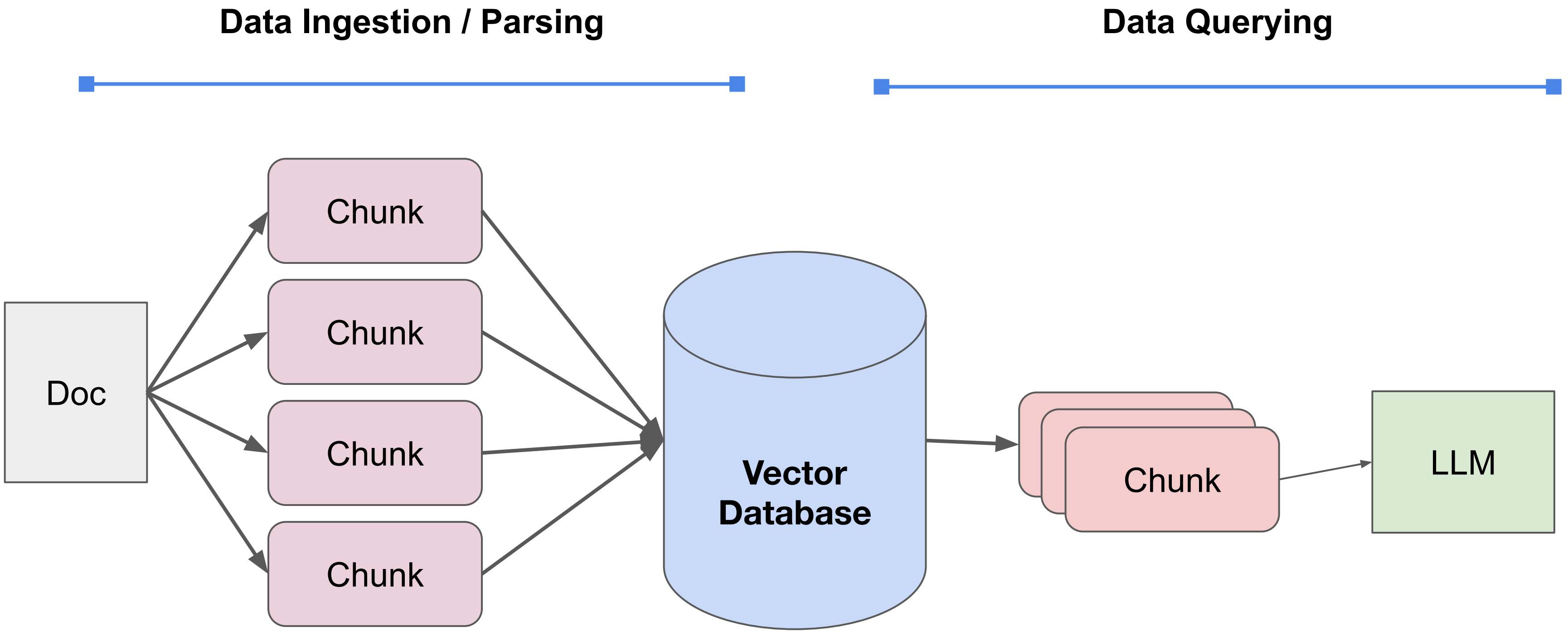
Complex/multistep reasoning

- Headless agent

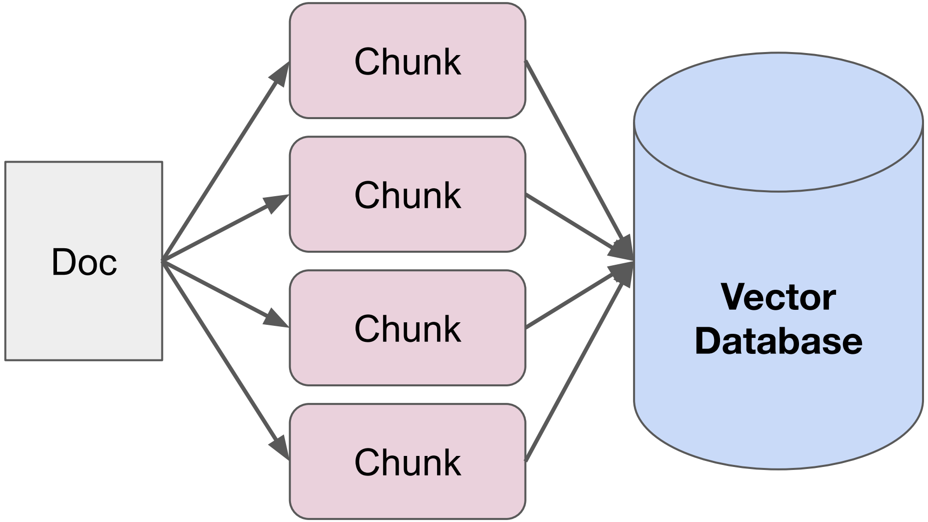
- rvational agent
- l analytics

Challenge: Data Management & Orchestration

Naive RAG Stack for building a QA System



Current RAG Stack (Data Ingestion/Parsing)



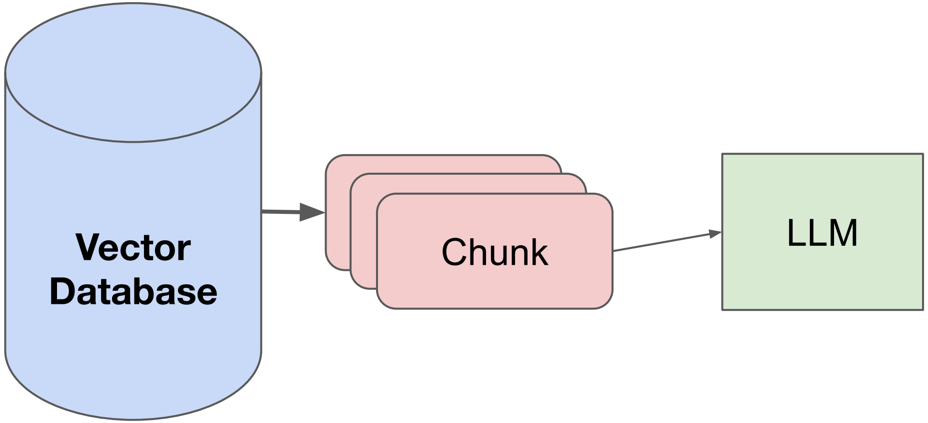
Naive State:

- Split up document(s) into even chunks.
- Each chunk does not contain parent context.
- All chunks are stored in the same collection in a vector database.

Current 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 (Response Quality)

- When RAG fails, the most common reason is bad retrieval
    - If the retrieved results are bad, there’s no way the LLM can synthesize a proper response without hallucinating!
- The most common retrieval method is top-k embedding lookup

Challenges with Naive RAG (Response Quality)

- Causes of bad retrieval quality
  - Each chunk does not have awareness of parent context or related context
  - The query assumes a certain traversal structure that top-k embedding lookup doesn’t utilize.
  - The data is redundant or out of date

Challenges with Naive RAG (System Concerns)

There are also system-level considerations with this stack

- How do you deal with updates in the source document?
    - How do you update stored chunks in the vector database?

Key Lessons

To improve your RAG stack,

Improve the way you define **state**, not just the retrieval algorithm!

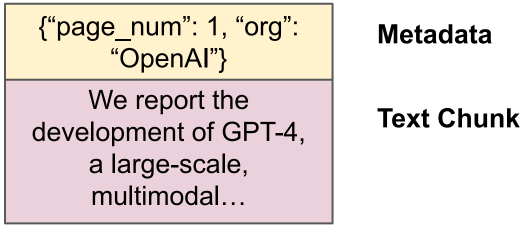
Data Tips/Tricks for Better Performing RAG

Augmenting Chunks with Context

- One of the reasons embedding retrieval fails is that relevant context chunks do not match the query embedding

Different Context Augmentation Strategies

Injecting Metadata



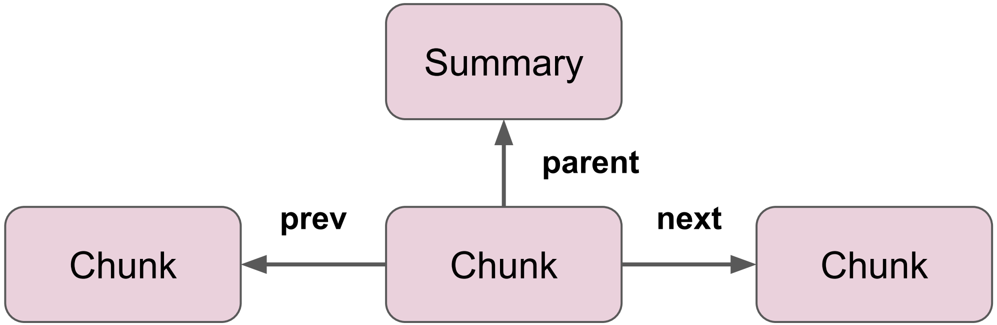
{"page_num": 1, "org": "OpenAI"}

Metadata

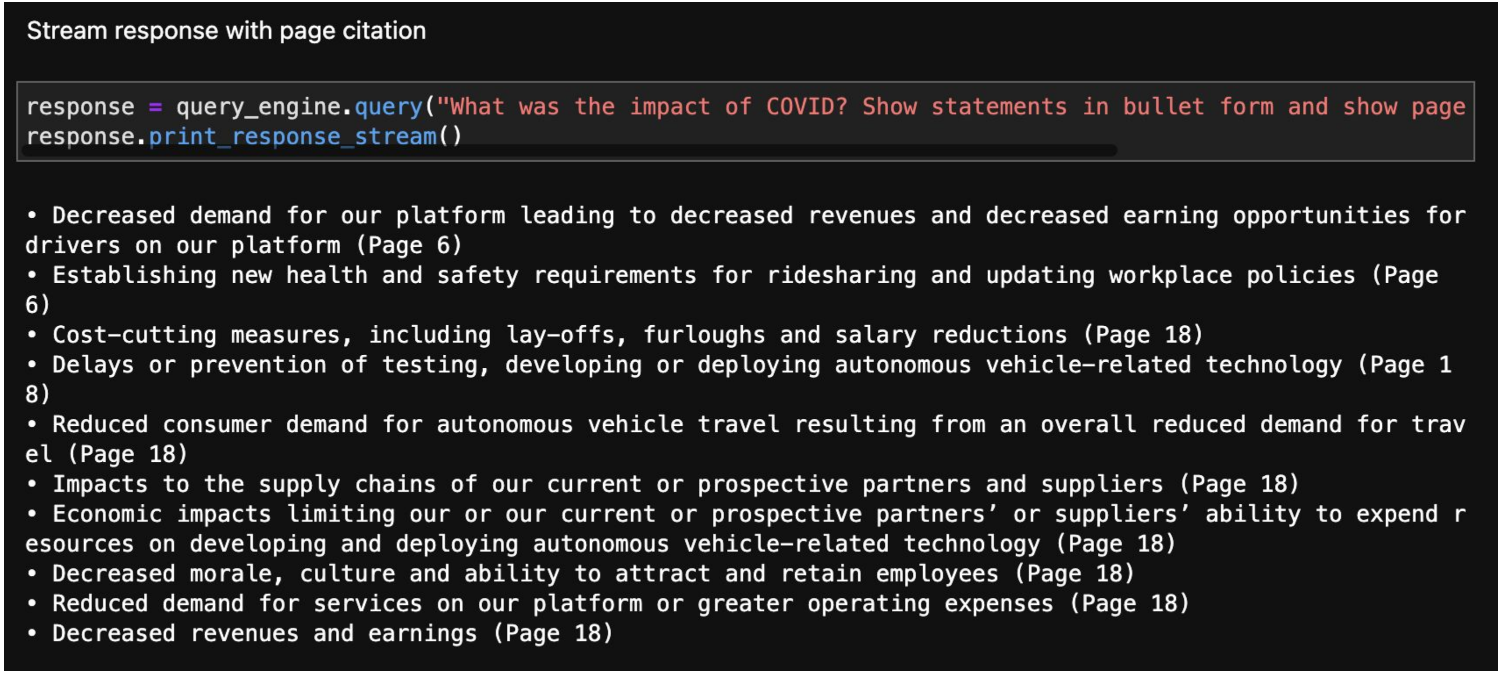
We report the development of GPT-4, a large-scale, multimodal…

Text Chunk

Defining Node Relationships



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

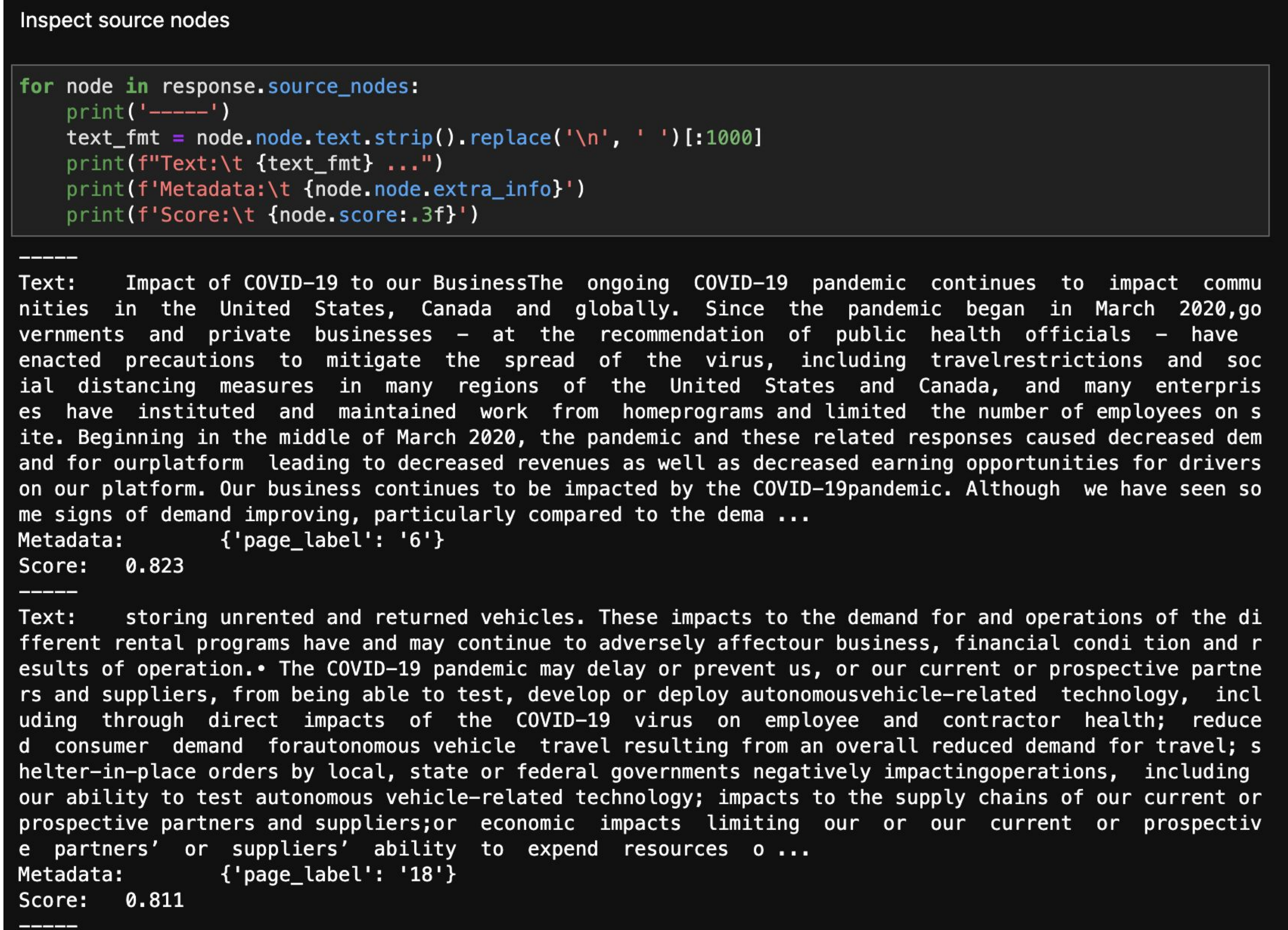


Stream response with page citation

```
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

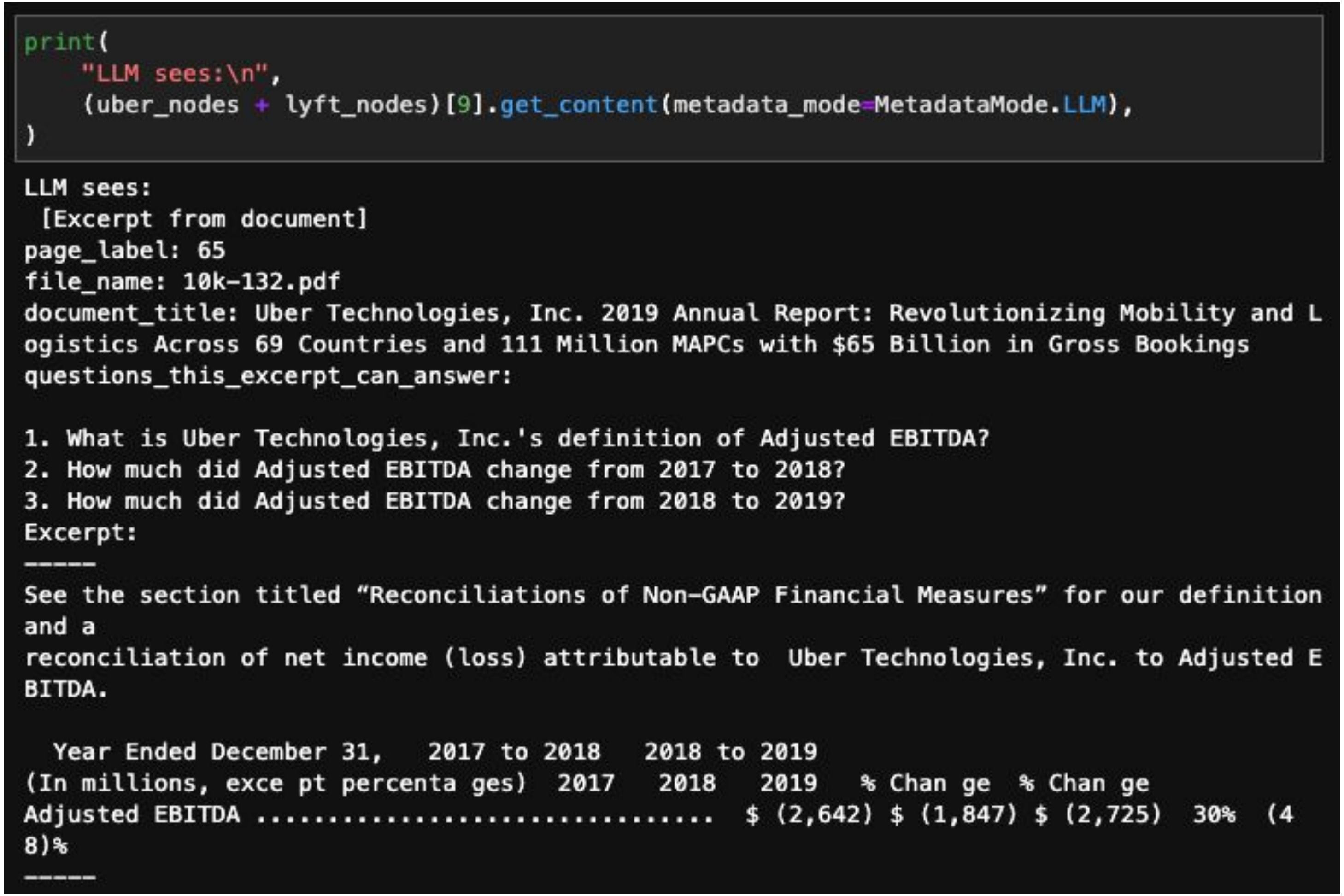


Inspect source nodes

```
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



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

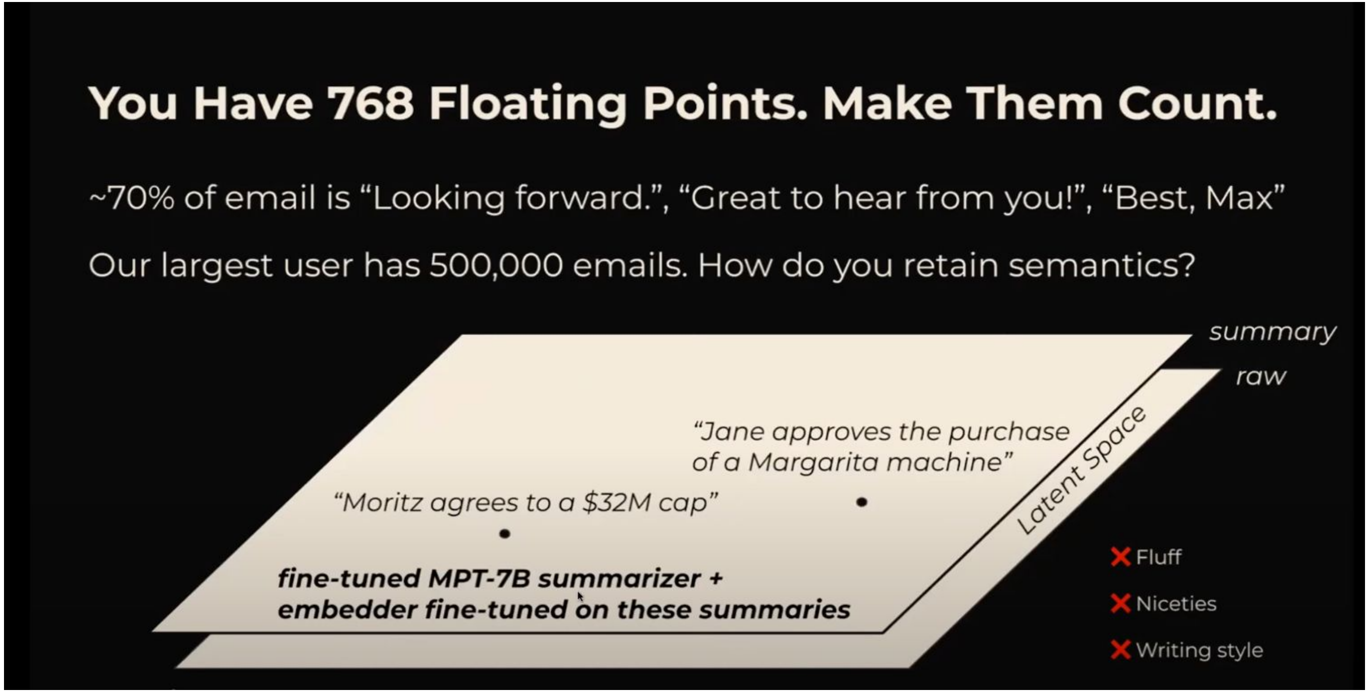
```
LLM sees:
 [Excerpt from document]
page_label: 65
file_name: 10k-132.pdf
document_title: Uber Technologies, Inc. 2019 Annual Report: Revolutionizing Mobility and L
ogistics Across 69 Countries and 111 Million MAPCs with $65 Billion in Gross Bookings
questions_this_excerpt_can_answer:

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)%
-----
```

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?

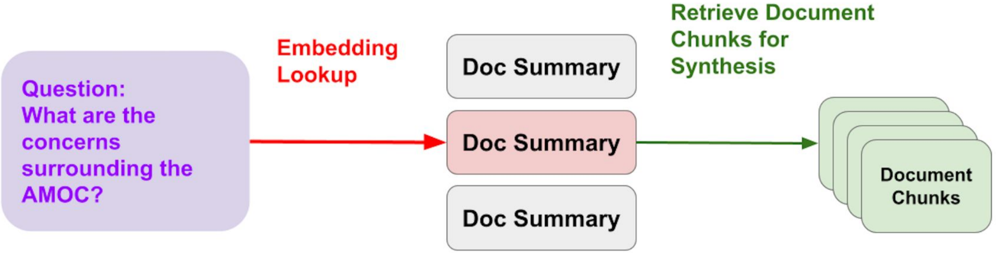
summary
raw
Latent Space
“Jane approves the purchase of a Margarita machine”
“Moritz agrees to a $32M cap”
fine-tuned MPT-7B summarizer + embedder fine-tuned on these summaries
✗ Fluff
✗ Niceties
✗ Writing style

Decouple Embeddings from Raw Text Chunks

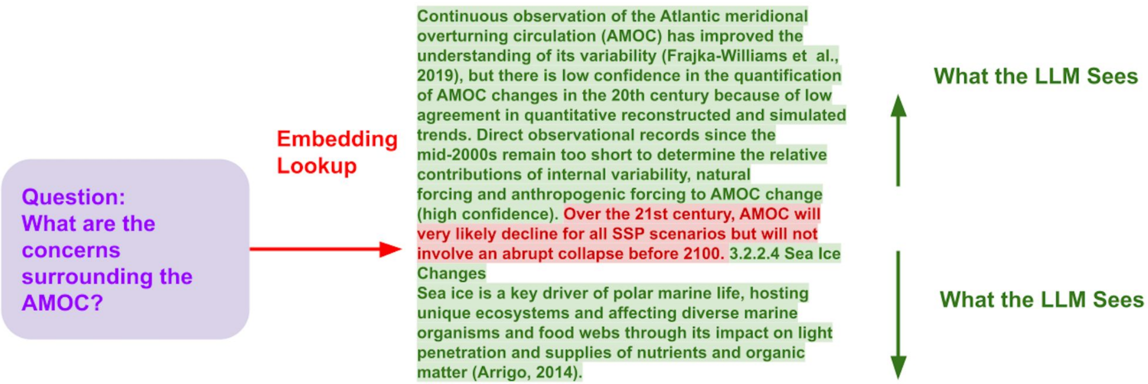
Solutions:

- Embed larger documents via summaries
- Embed text at the sentence-level - then **expand** that window during LLM synthesis
- Finetune embeddings over a specific corpus

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

Embed Sentence → Link to Expanded Window



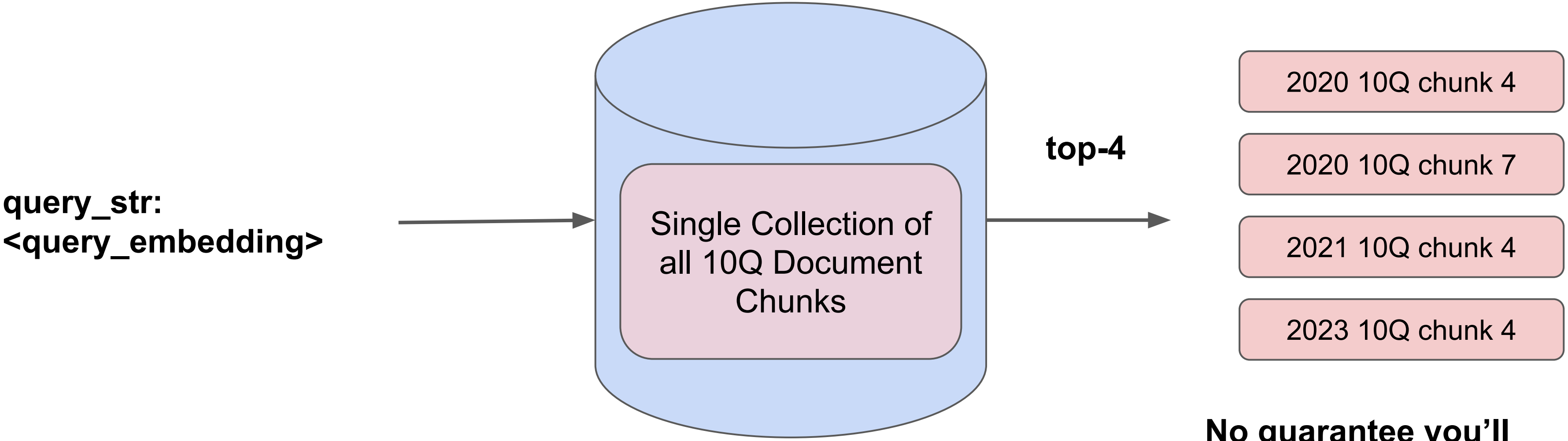
Continuous observation of the Atlantic meridional overturning circulation (AMOC) has improved the understanding of its variability (Frajka-Williams et al., 2019), but there is low confidence in the quantification of AMOC changes in the 20th century because of low agreement in quantitative reconstructed 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).

Organize your data for more structured retrieval

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

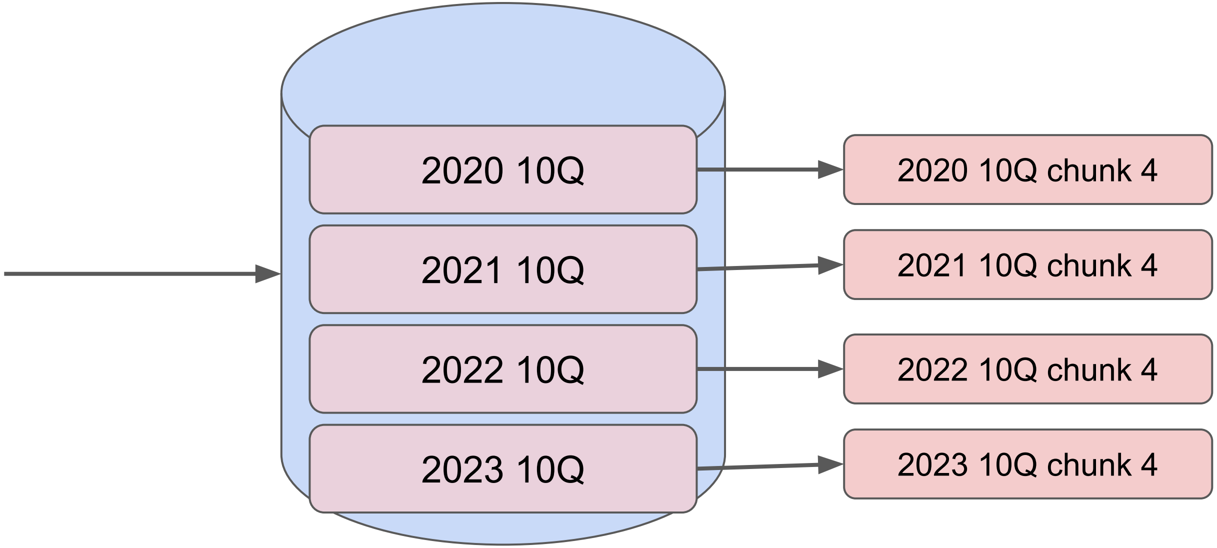
Question: “Can you tell me about Google’s R&D initiatives from 2020 to 2023?”

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>

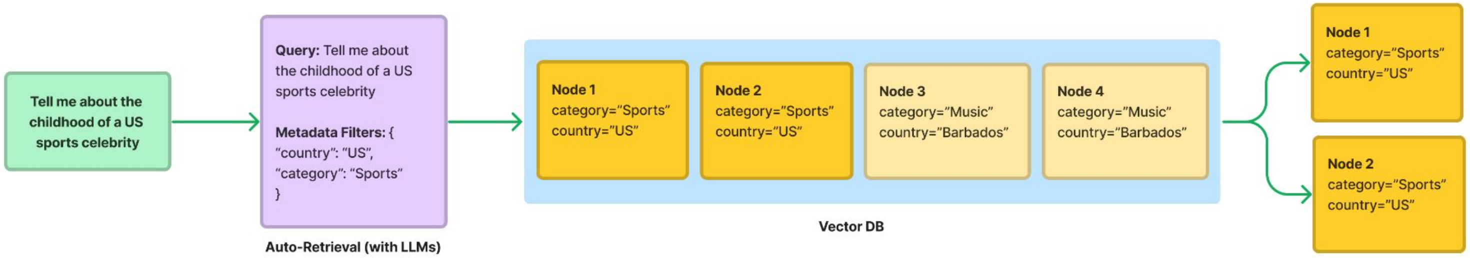
Metadata tags:
<metadata_tags>



Organize your data for more structured retrieval

Two main approaches here

Metadata Filters + Auto-Retrieval



Tell me about the childhood of a US sports celebrity

Query: Tell me about the childhood of a US sports celebrity

Metadata Filters: {
“country”: “US”,
“category”: “Sports”
}

Auto-Retrieval (with LLMs)

Node 1
category=”Sports”
country=”US”

Node 2
category=”Sports”
country=”US”

Node 3
category=”Music”
country=”Barbados”

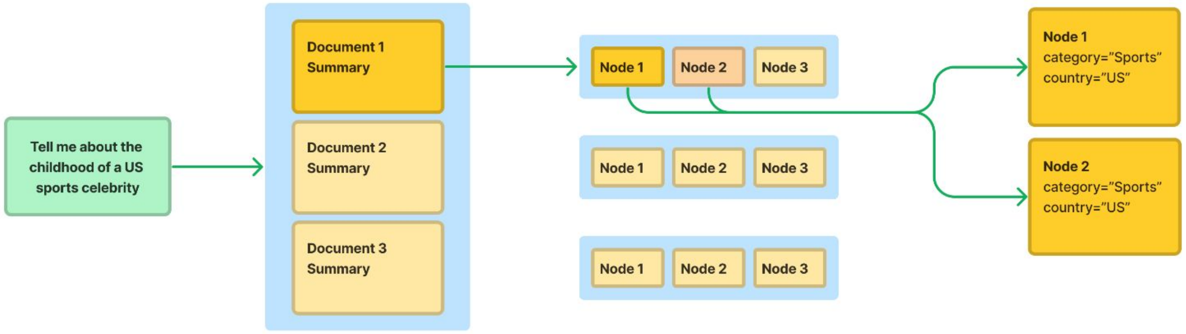
Node 4
category=”Music”
country=”Barbados”

Vector DB

Node 1
category=”Sports”
country=”US”

Node 2
category=”Sports”
country=”US”

Document Hierarchies (Summaries + Raw Chunks) + Recursive Retrieval



Tell me about the childhood of a US sports celebrity

Document 1 Summary

Document 2 Summary

Document 3 Summary

Node 1 Node 2 Node 3

Node 1 Node 2 Node 3

Node 1 Node 2 Node 3

Node 1
category=”Sports”
country=”US”

Node 2
category=”Sports”
country=”US”

Data Solutions in LlamaIndex

Define/customize metadata: https://gpt-index.readthedocs.io/en/latest/how_to/customization/custom_documents.html

Automatic metadata extraction: https://gpt-index.readthedocs.io/en/latest/how_to/index/metadata_extraction.html

Document Comparisons:
https://gpt-index.readthedocs.io/en/latest/examples/query_engine/sub_question_query_engine.html

Comparing document structuring approaches:
https://gpt-index.readthedocs.io/en/latest/examples/retrievers/auto_vs_recursive_retriever.html

Sentence-level Retrieval + Expanded Context During LLM Synthesis:

https://gpt-index.readthedocs.io/en/latest/examples/node_postprocessor/MetadataReplacementDemo.html

Handling Document Updates:
https://gpt-index.readthedocs.io/en/latest/how_to/index/usage_pattern.html#handling-document-update