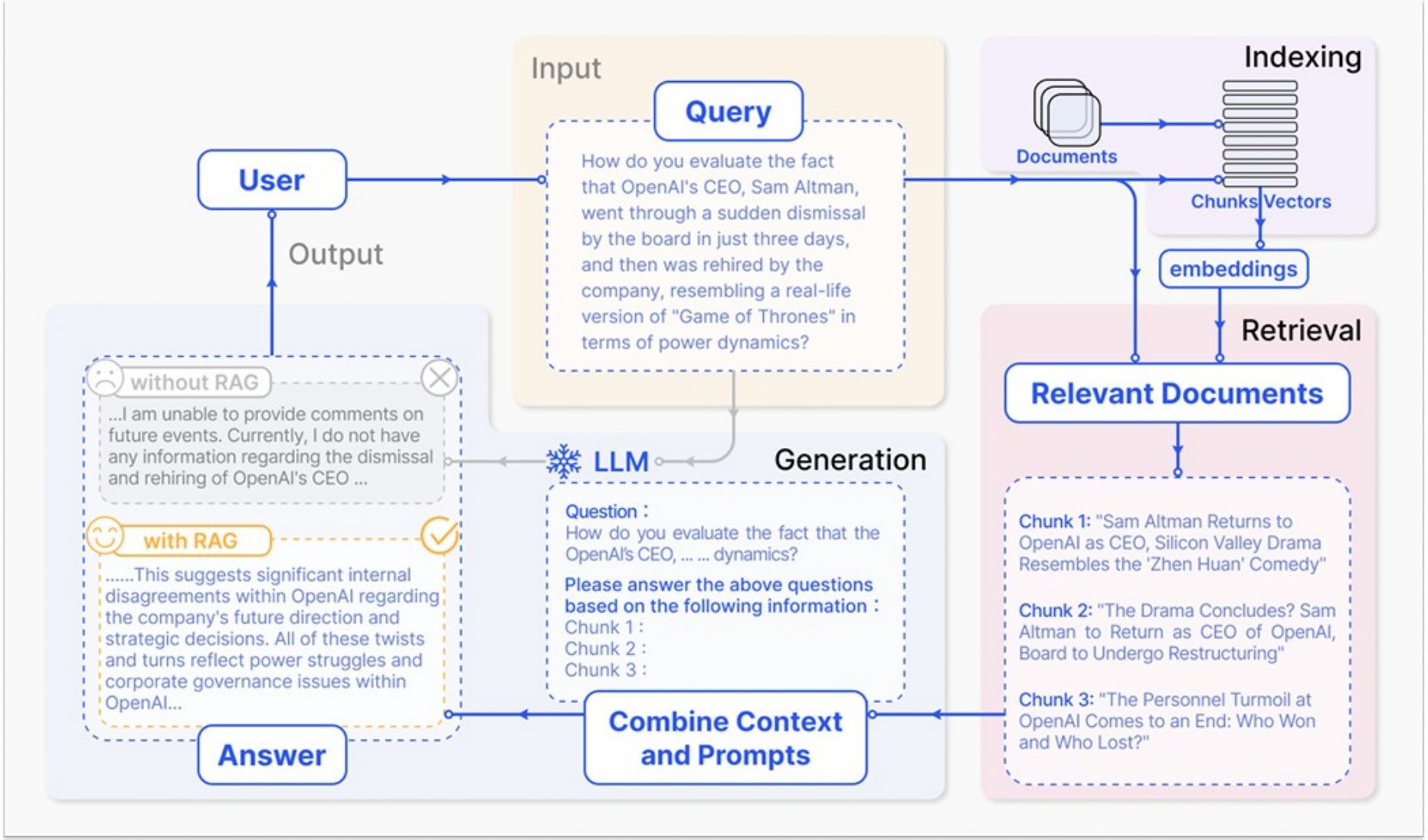
RAG

How to update LLM knowledge

- Update model parameters
  - Costly
  - Low frequency
- Retrieval Augmented Generation (RAG)
  - Information retriever + LLM
  - Other benefits: information provenance

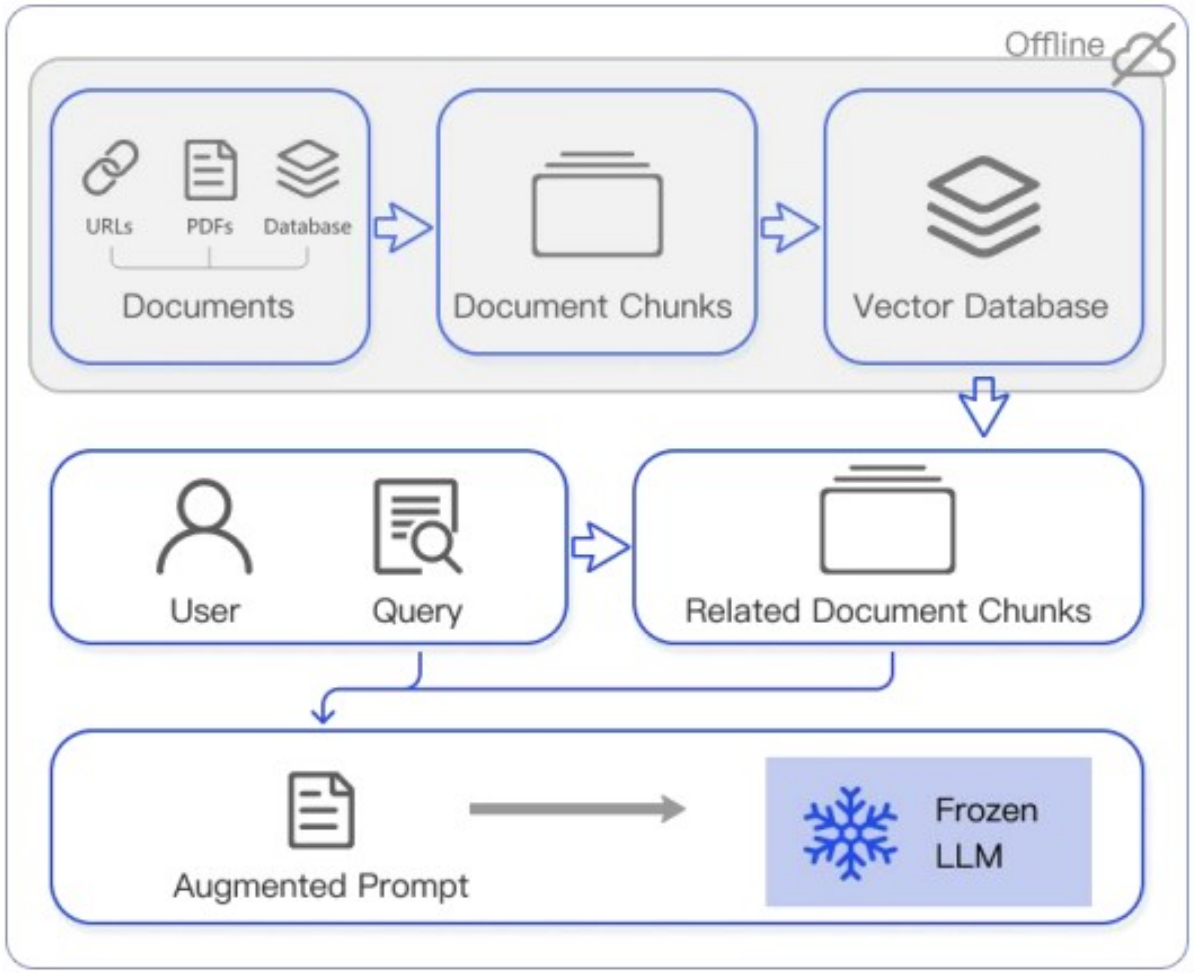
A RAG example



Gao, Yunfan, et al. "Retrieval-augmented generation for large language models: A survey." *arXiv preprint arXiv:2312.10997* (2023).

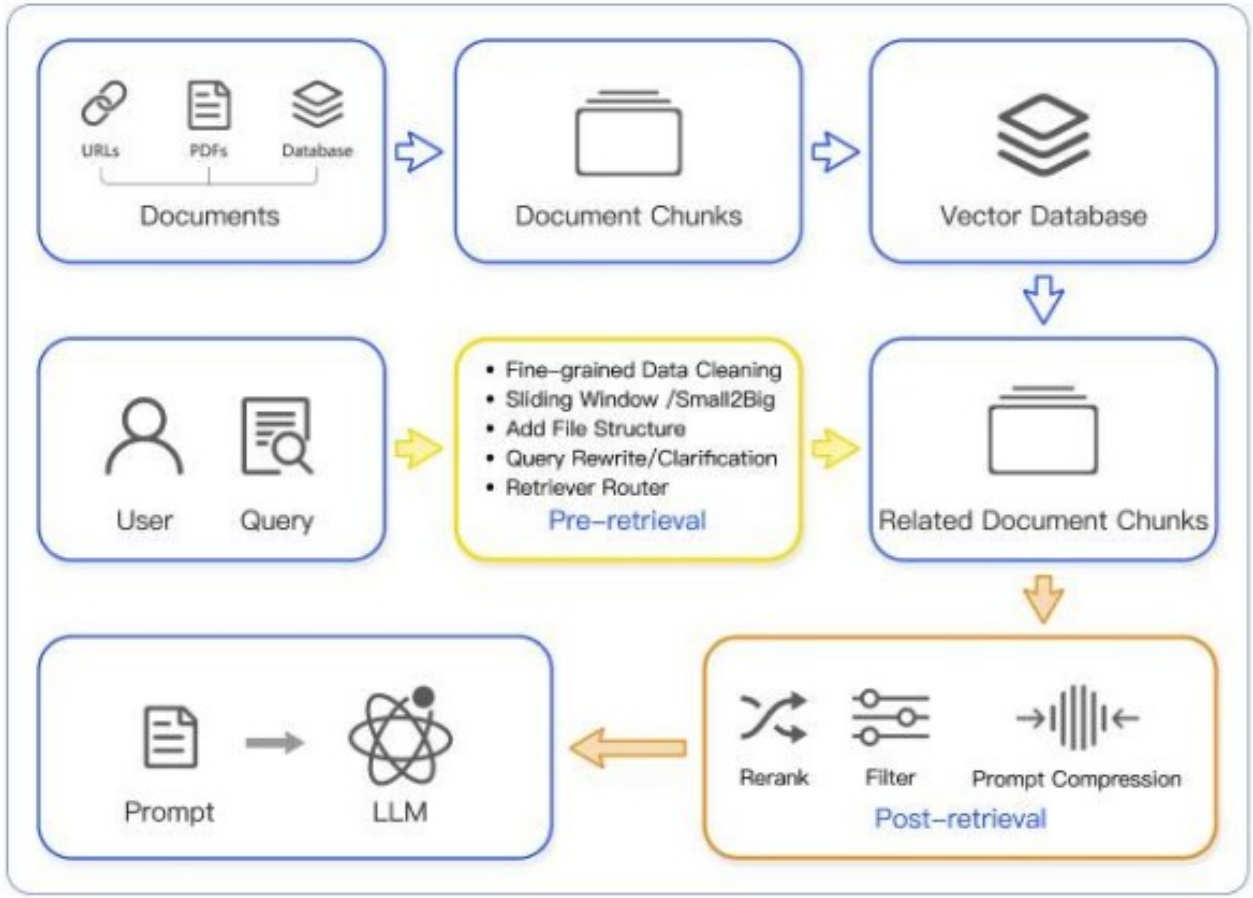
Naïve RAG

- Indexing
  - Divide the document into even chunks, each chunk being a piece of the original text.
  - Using the encoding model to generate an embedding for each chunk
  - Store the Embedding of each block in the vector database.
- Retrieval
- Generation

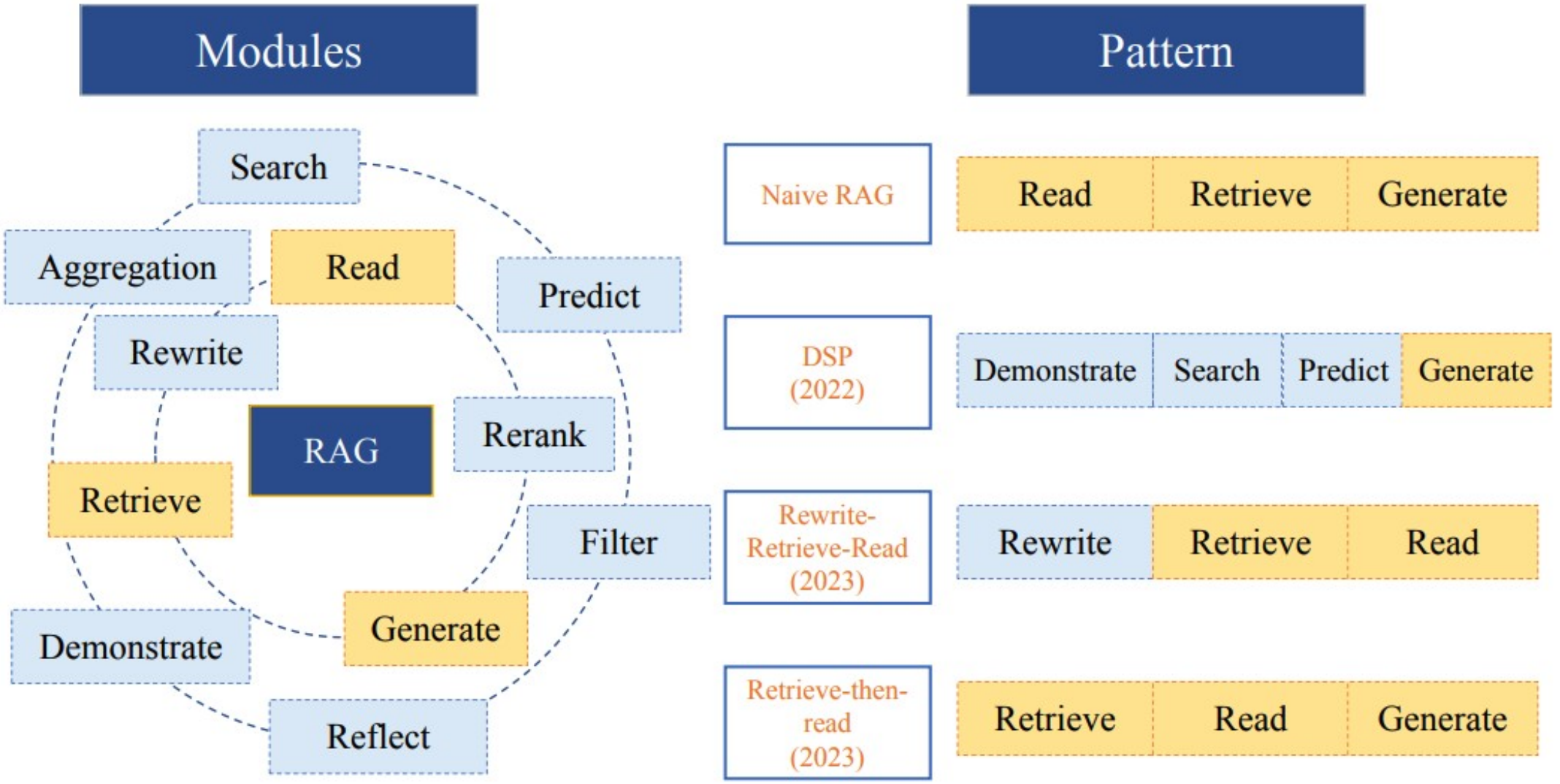


Advanced RAG

- Index Optimization
  - sliding window, fine-grained segmentation, adding metadata
- Pre-Retrieval Process
  - retrieve routes, summaries, rewriting, and confidence judgment
- Post-Retrieval Process
  - reorder, filter content retrieval



Modular RAG


Modules
Search
Aggregation
Read
Predict
Rewrite
RAG
Rerank
Retrieve
Filter
Generate
Demonstrate
Reflect
Pattern
Naive RAG
Read
Retrieve
Generate
DSP
(2022)
Demonstrate
Search
Predict
Generate
Rewrite-
Retrieve-Read
(2023)
Rewrite
Retrieve
Read
Retrieve-then-
read
(2023)
Retrieve
Read
Generate

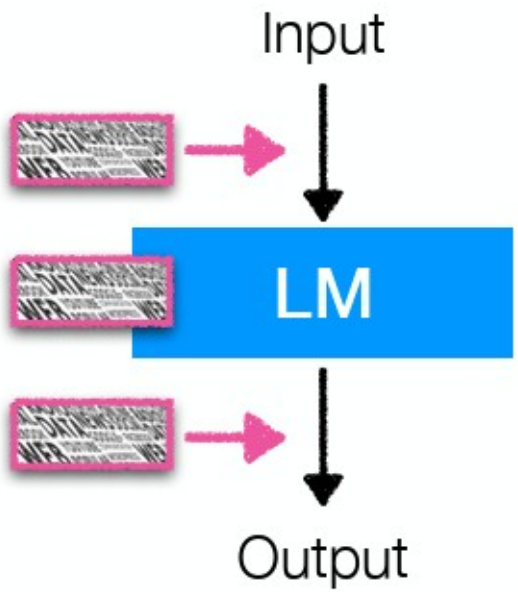
Key Questions of RAG

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

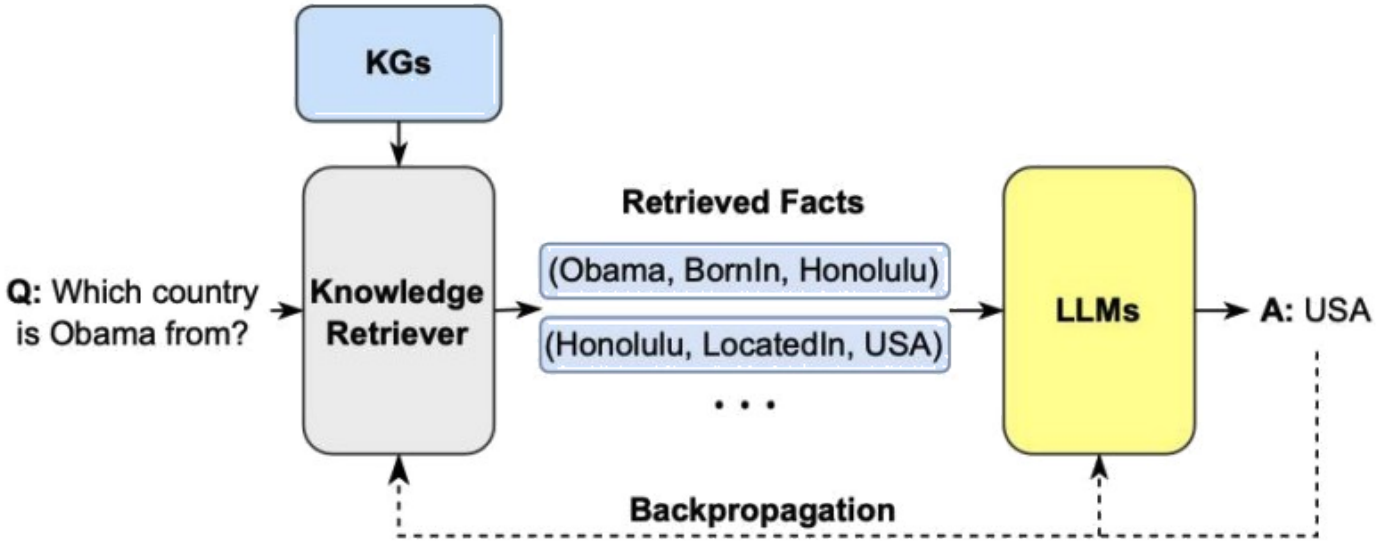
How to use retrieval?



When to retrieve?

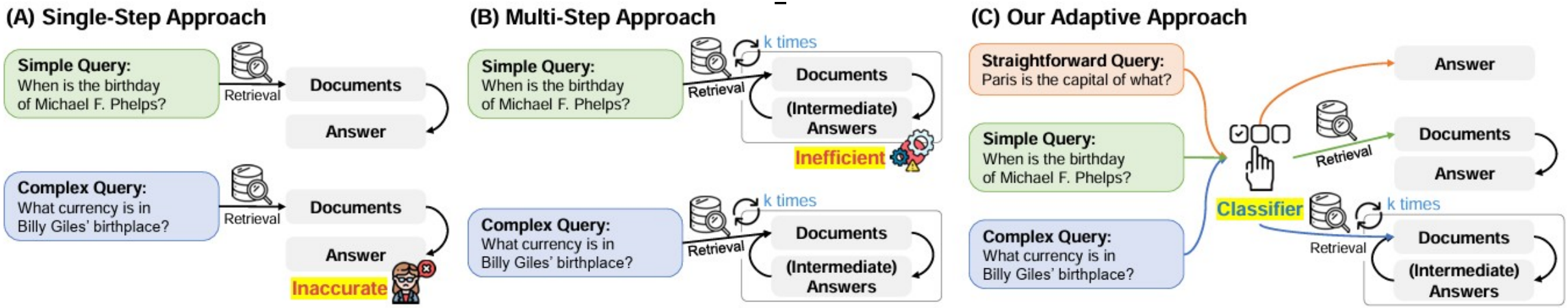
GraphRAG

- Use LLM (or other models) to extract key entities from the question
- Retrieve subgraphs based on entities, delving to a certain depth, such as 2 hops or even more
- Utilize the obtained context to generate answers through LLM



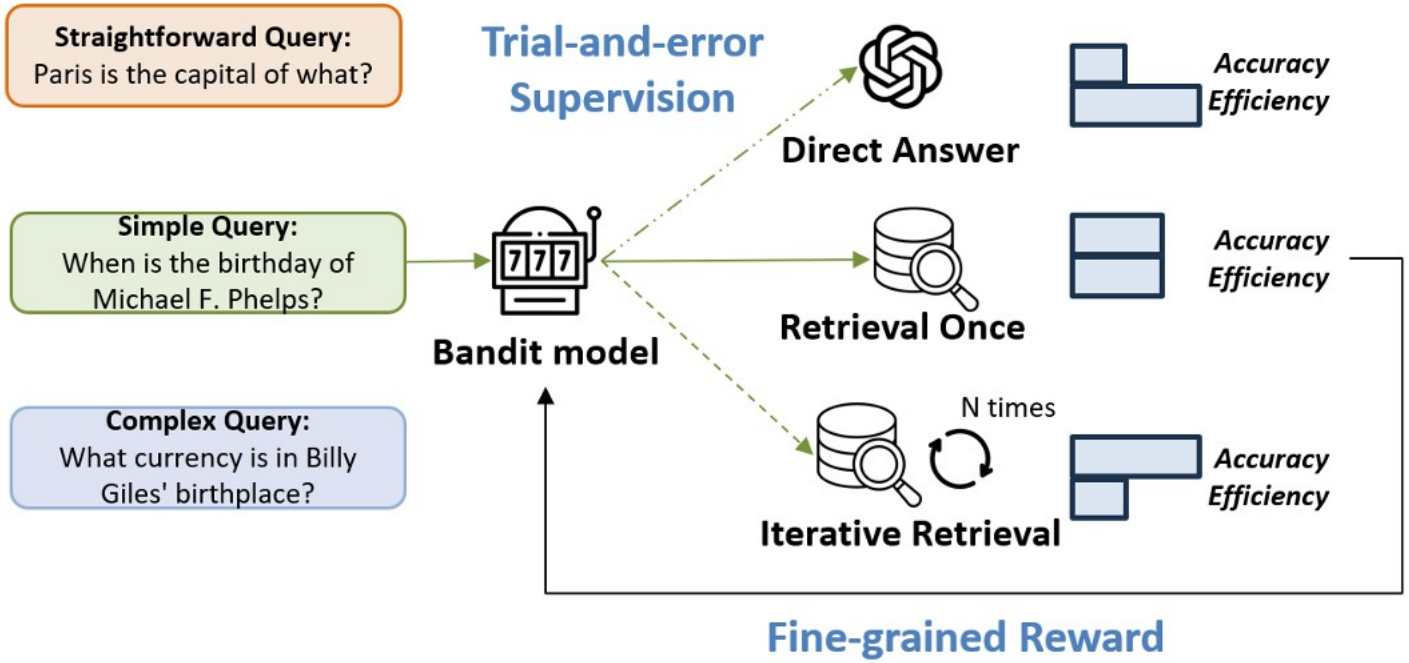
AdaptiveRAG

https://aclanthology.org/2024.naacl-long.389/

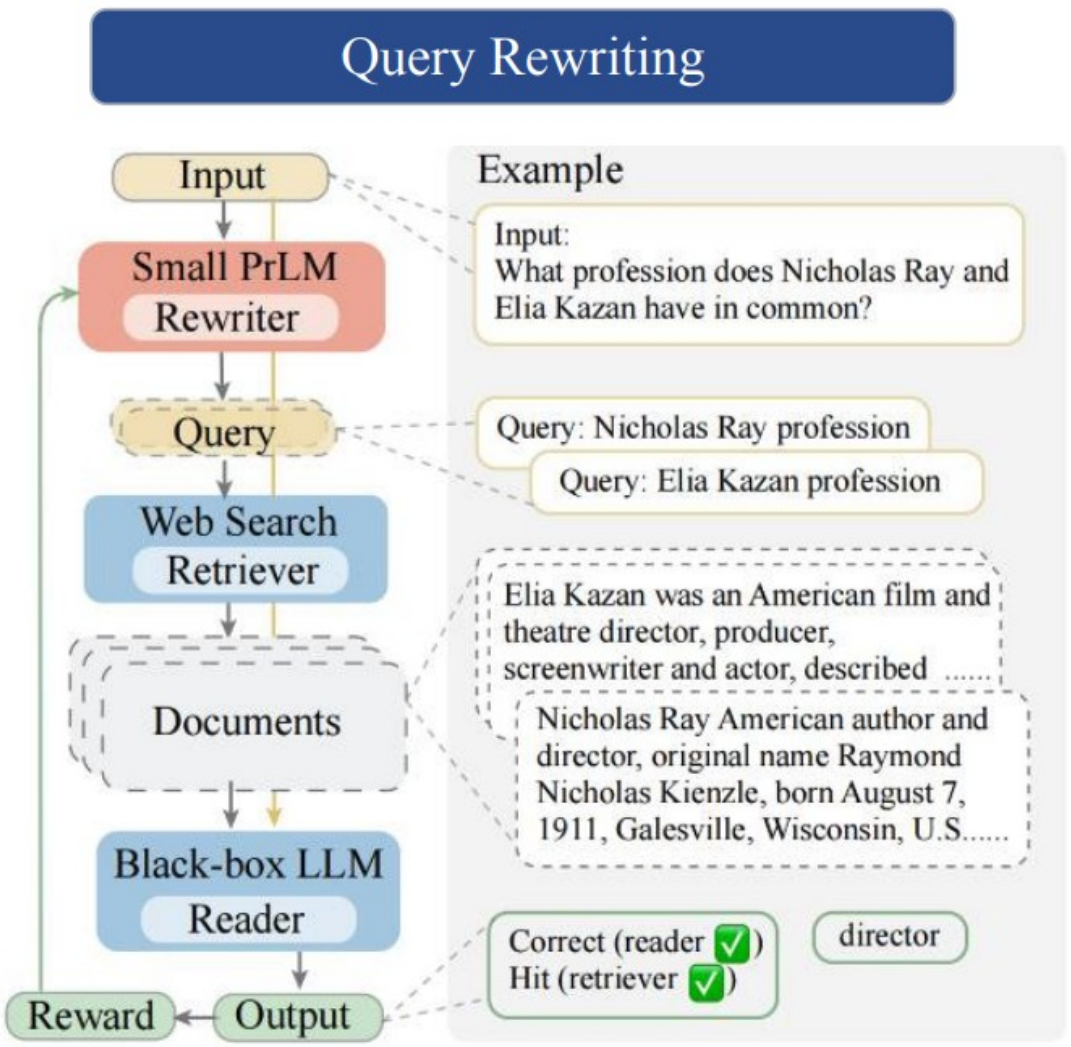


MBA-RAG

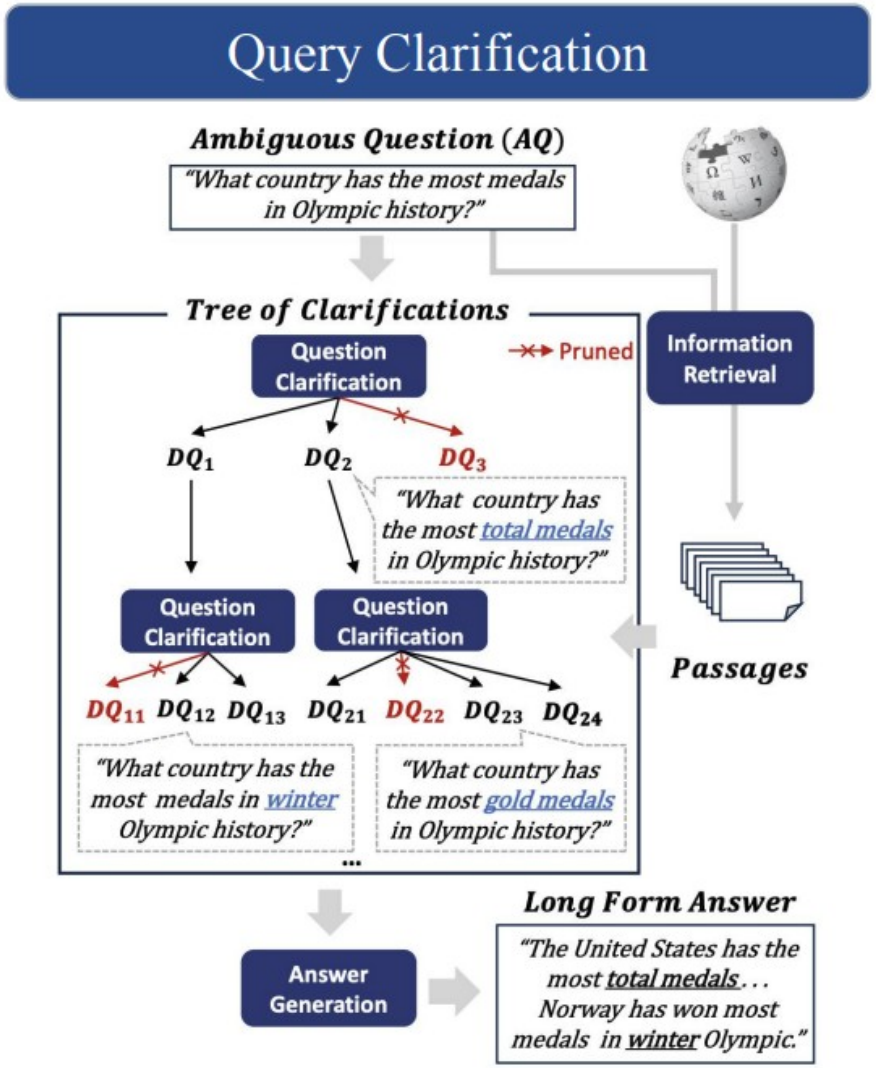
https://aclanthology.org/2025.coling-main.218/



Query Optimization



Rewrite-Retrieve-Read [Ma et al., 2023]



Tree of Clarifications（TOC）[Kim et al.,2023]

Multilingual & Multimodal RAG