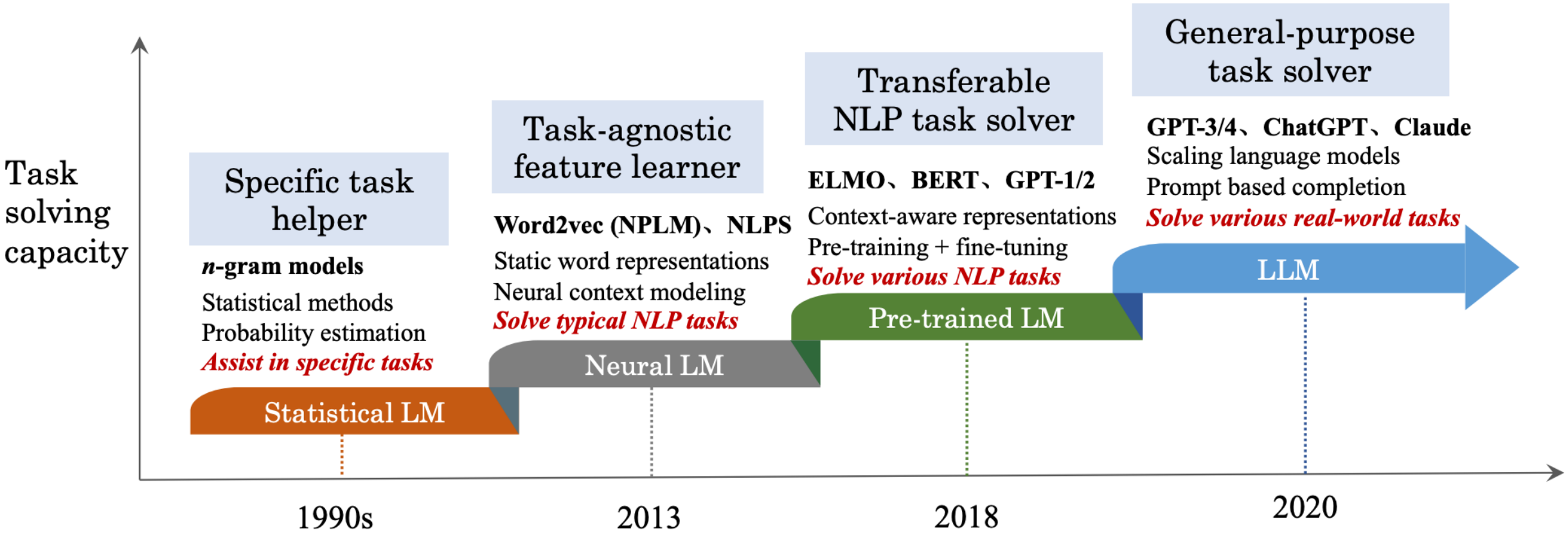
Retrieval Augmented Generation

Mehwish Alam
Associate Professor
Télécom Paris
Institut Polytechnique de Paris
Winter Semester 2024-2025

Brief Recap

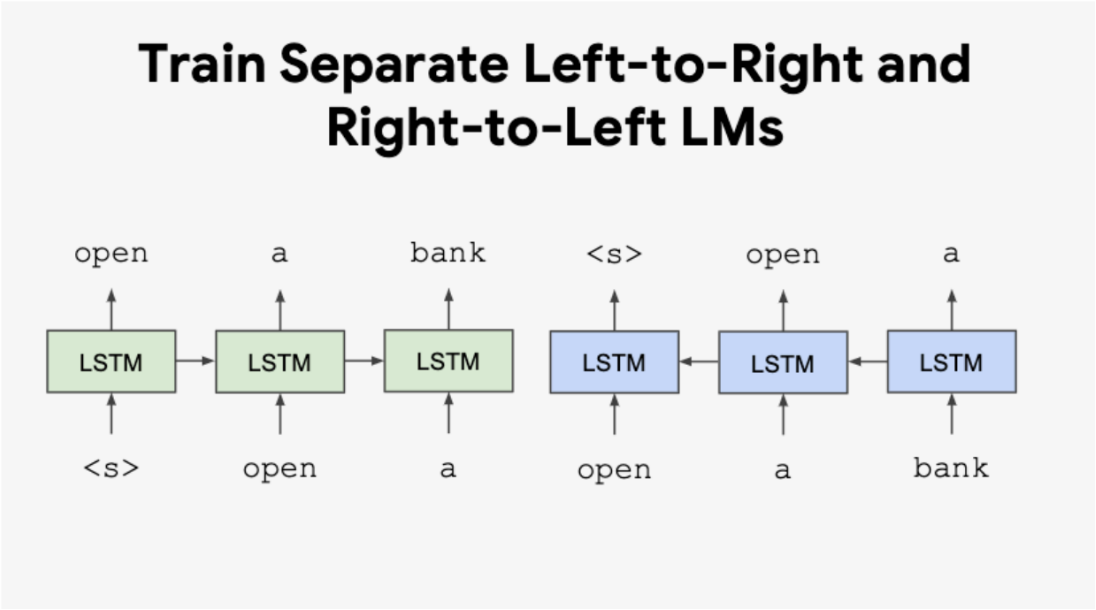
Evolution of Language Models

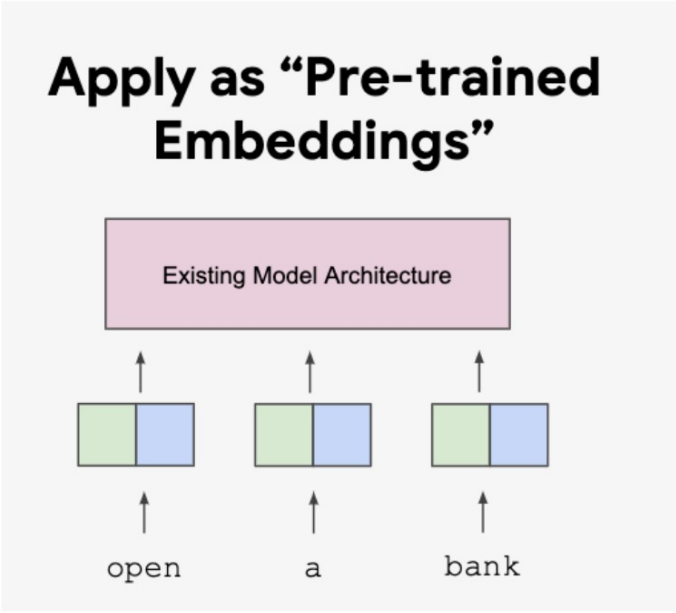


Source: https://arxiv.org/pdf/2303.18223.pdf

Prior Work: ELMo

- ELMo (Peters et al., 2018; NAACL 2018 best paper)
  - Train **two separate unidirectional LMs** (left-to-right and right-to-left) based on LSTMs
  - Feature-based approach: **pre-trained representations** used as input to task-specific models
  - Trained on single sentences from 1B word benchmark (Chelba et al., 2014)

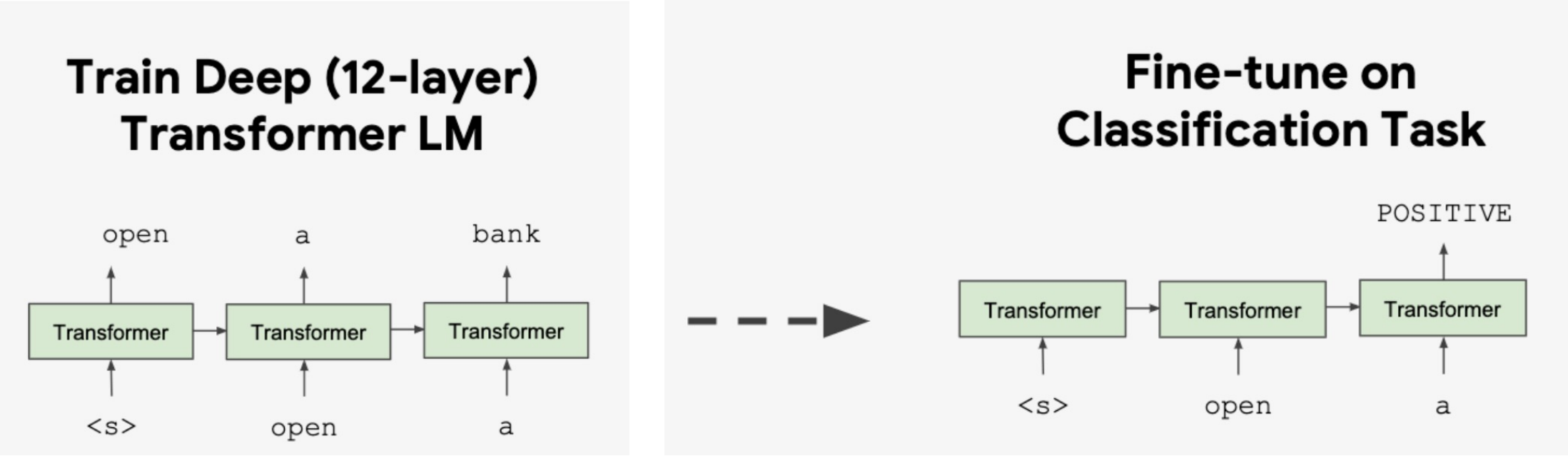




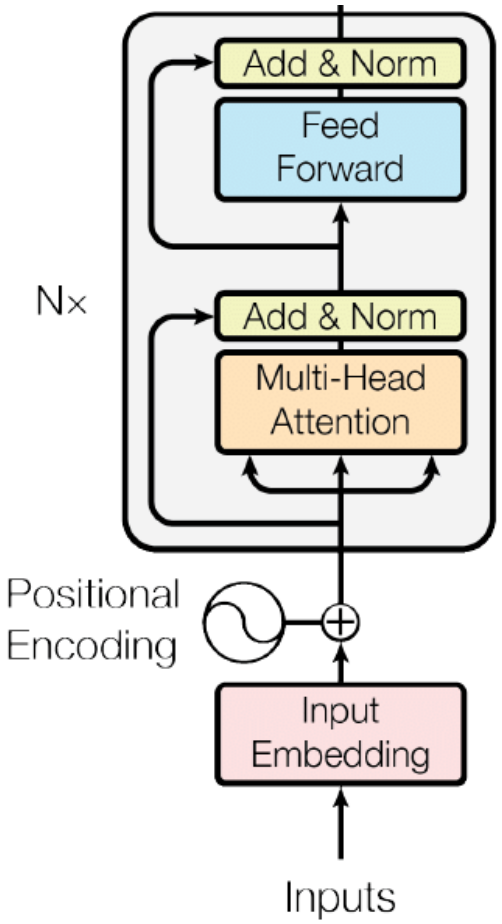
Source: https://www.cs.princeton.edu/courses/archive/fall22/cos597G/

Prior Work: OpenAI GPT

- OpenAI GPT (Radford et al., 2018; released in 2018/6)
  - Train **one unidirectional LM (left-to-right)** based on a deep Transformer decoder
  - Fine-tuning approach: all pre-trained parameters are re-used & updated on downstream tasks
  - Trained on **512-token segments on BooksCorpus** — much longer context!



BERT pre-training: putting together

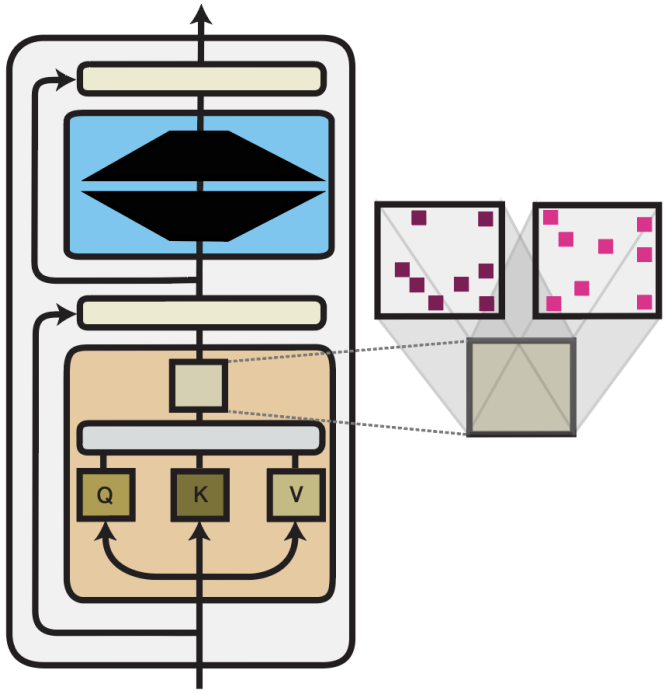


- BERT-base: 12 layers, 768 hidden size, 12 attention heads, 110M parameters
- BERT-large: 24 layers, 1024 hidden size, 16 attention heads, 340M parameters
- Training corpus: Wikipedia (2.5B) + BooksCorpus (0.8B)
- Max sequence size: 512 word pieces
  (roughly 256 and 256 for two non-contiguous sequences)
- Trained for 1M steps, batch size 128k

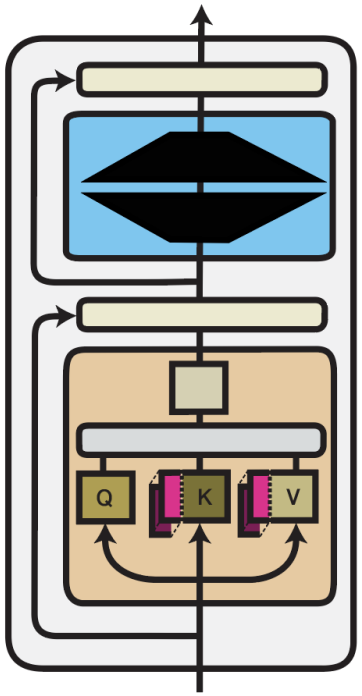
Same as OpenAI GPT

OpenAI GPT was trained on BooksCorpus only!

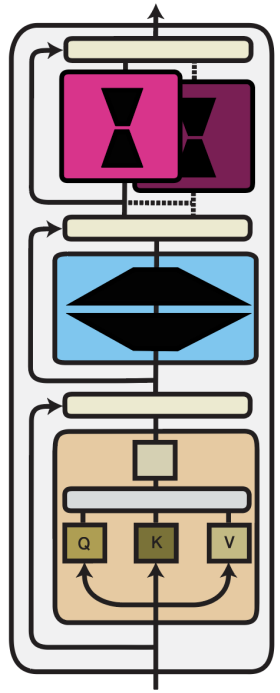
Three Computation Functions



Parameter Composition



Input Composition

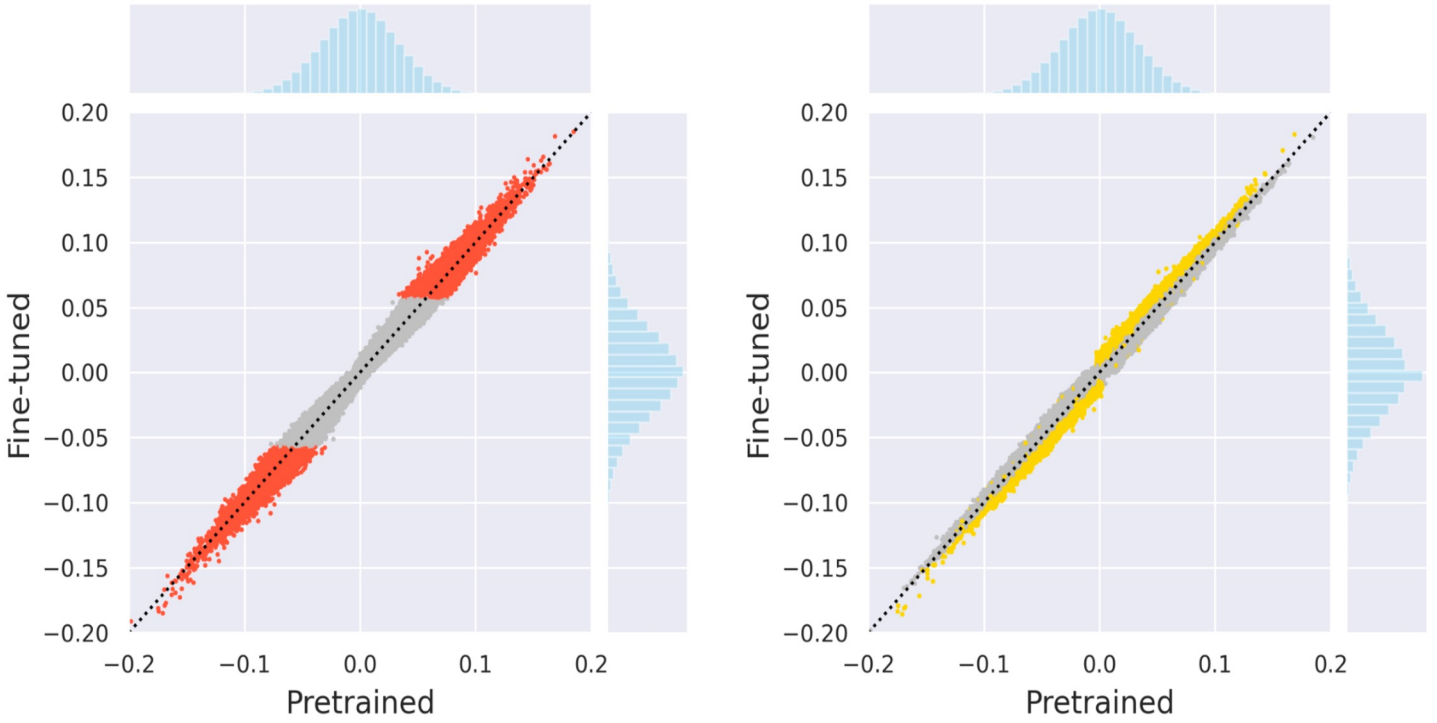


Function Composition

Pruning Pre-trained Models

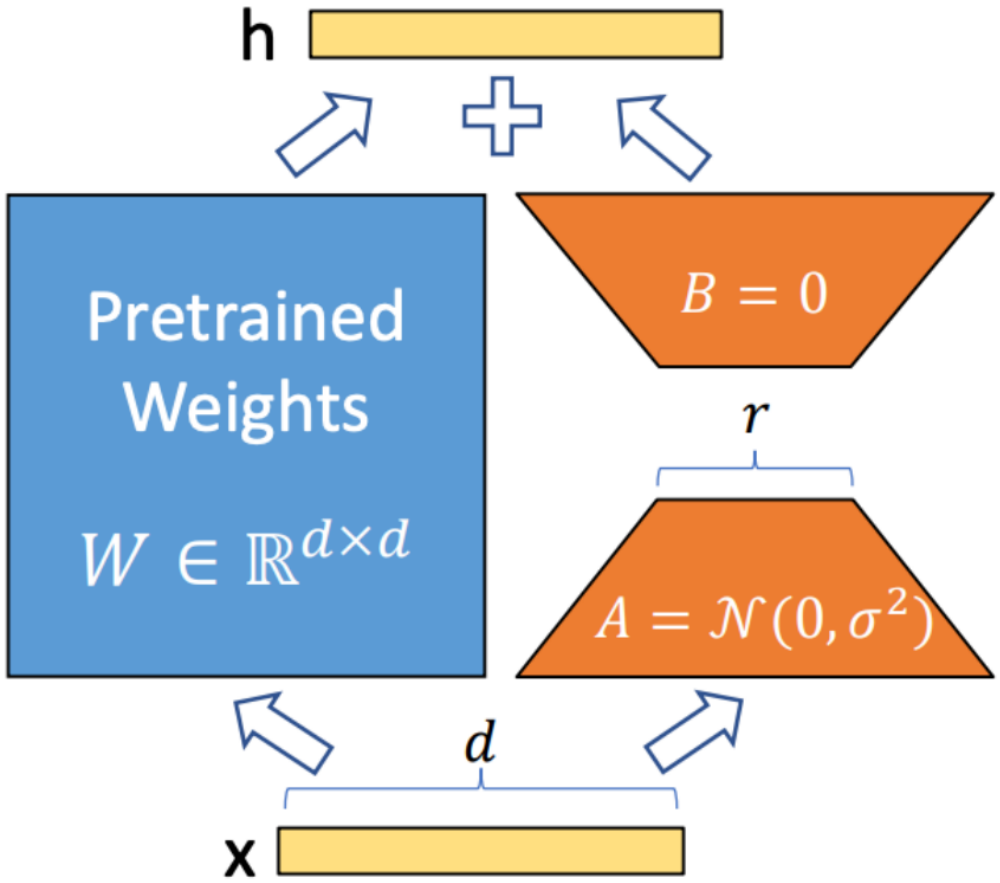
- Pruning does not consider how weights change during fine-tuning
- **Magnitude pruning**: keep weights farthest from 0
- **Movement pruning** [Sanh et al., 2020]: keep weights that *move the most away* from 0

Fine-tuned weights stay close to their their pre-trained values.
Magnitude pruning (left) selects weights that are far from 0.



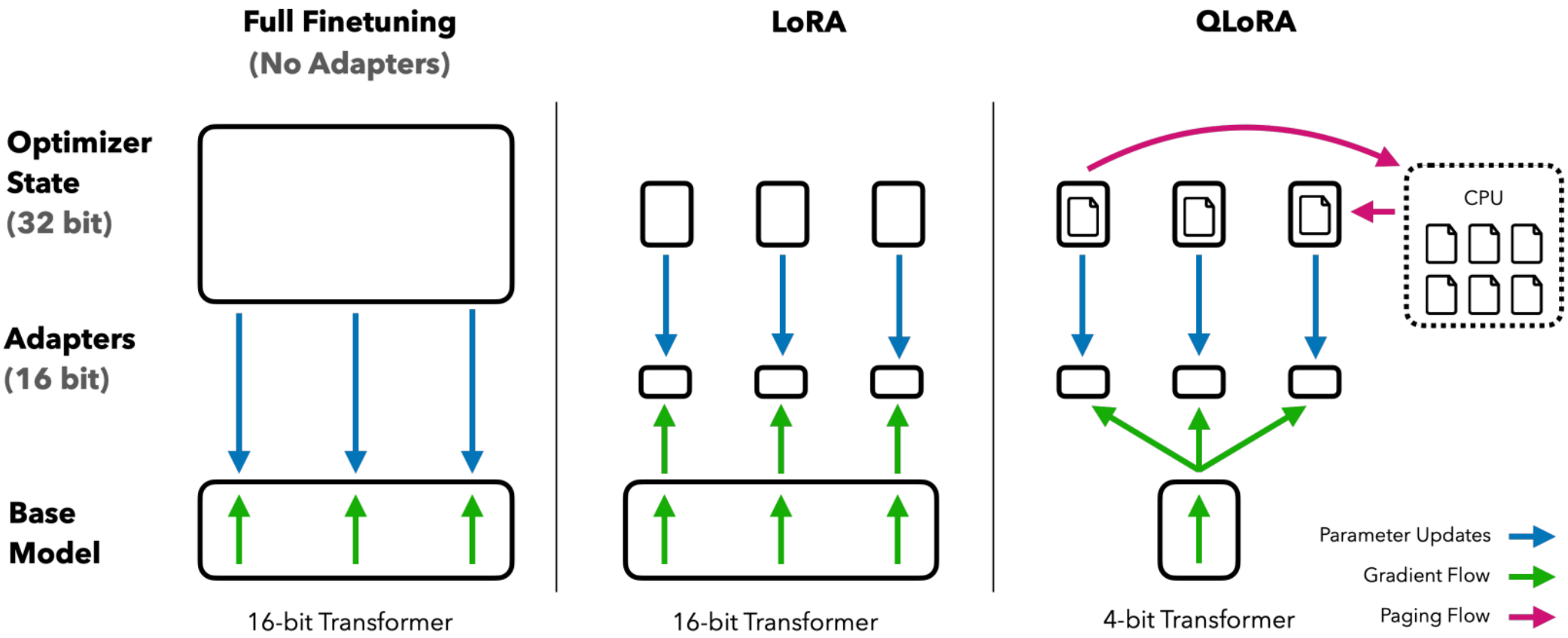
Movement pruning (right) selects weights that move away from 0.

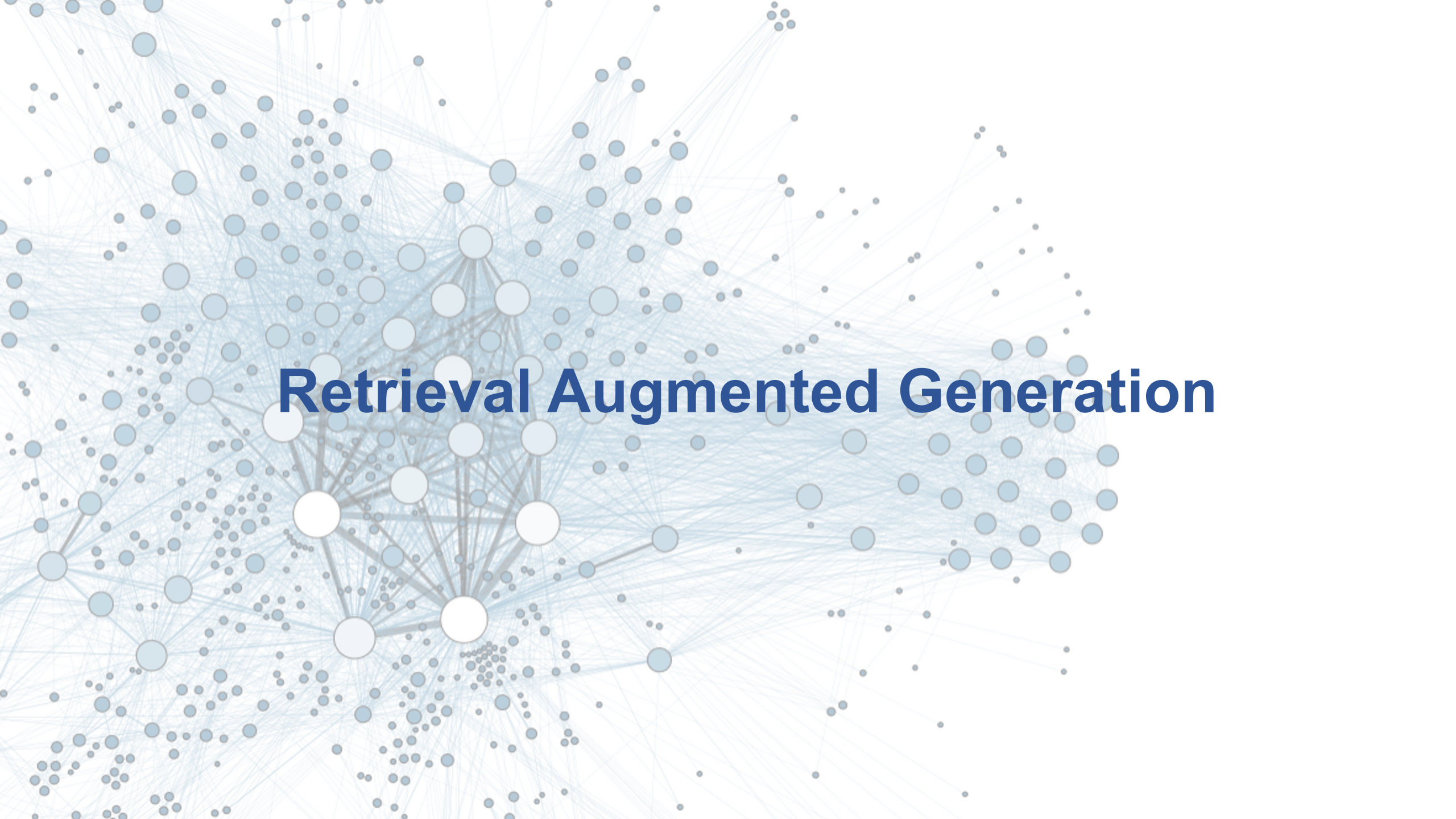
Low-rank-parameterized update matrices



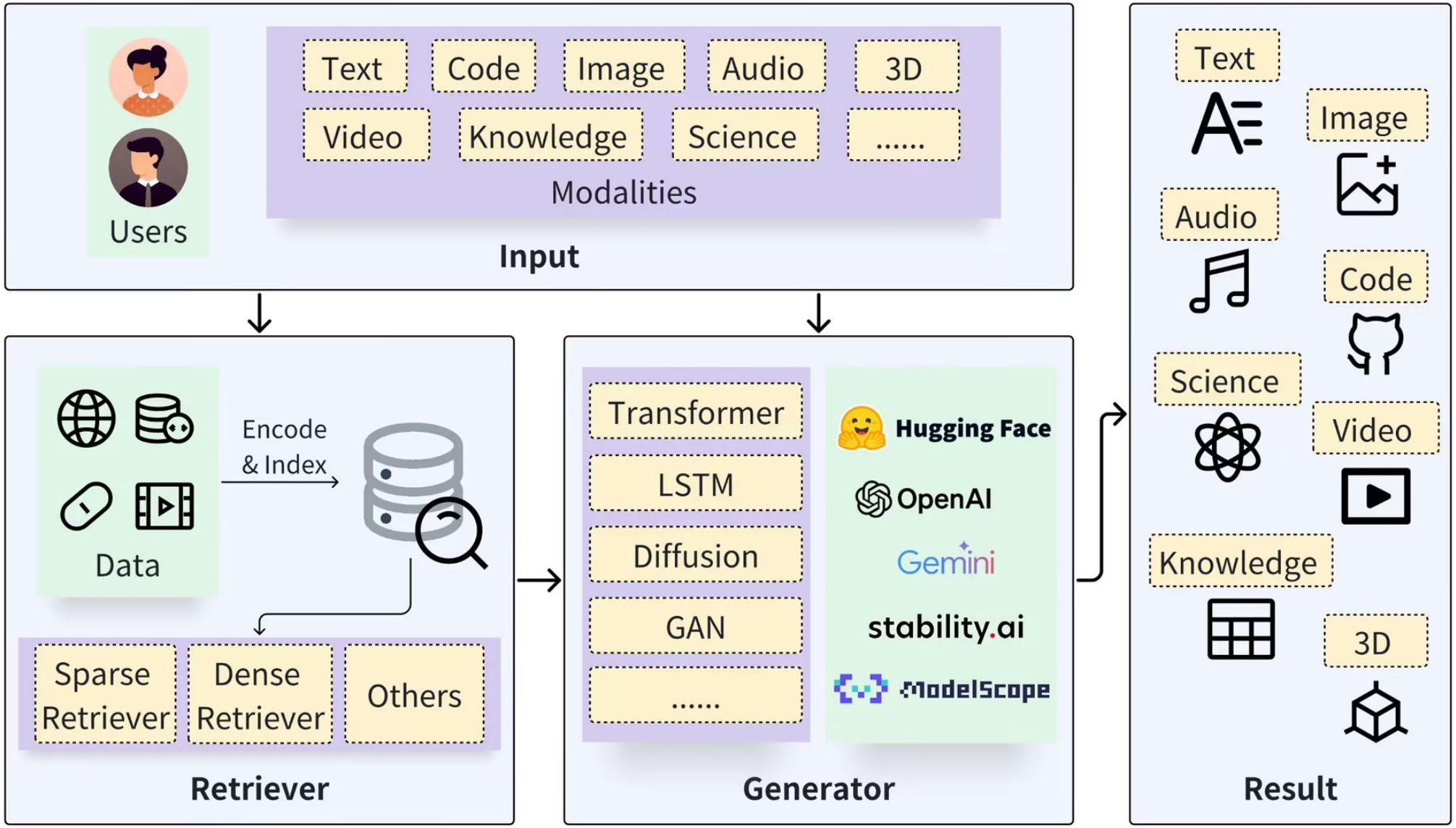
Source: https://arxiv.org/pdf/2305.14314

From LoRA to QLoRA

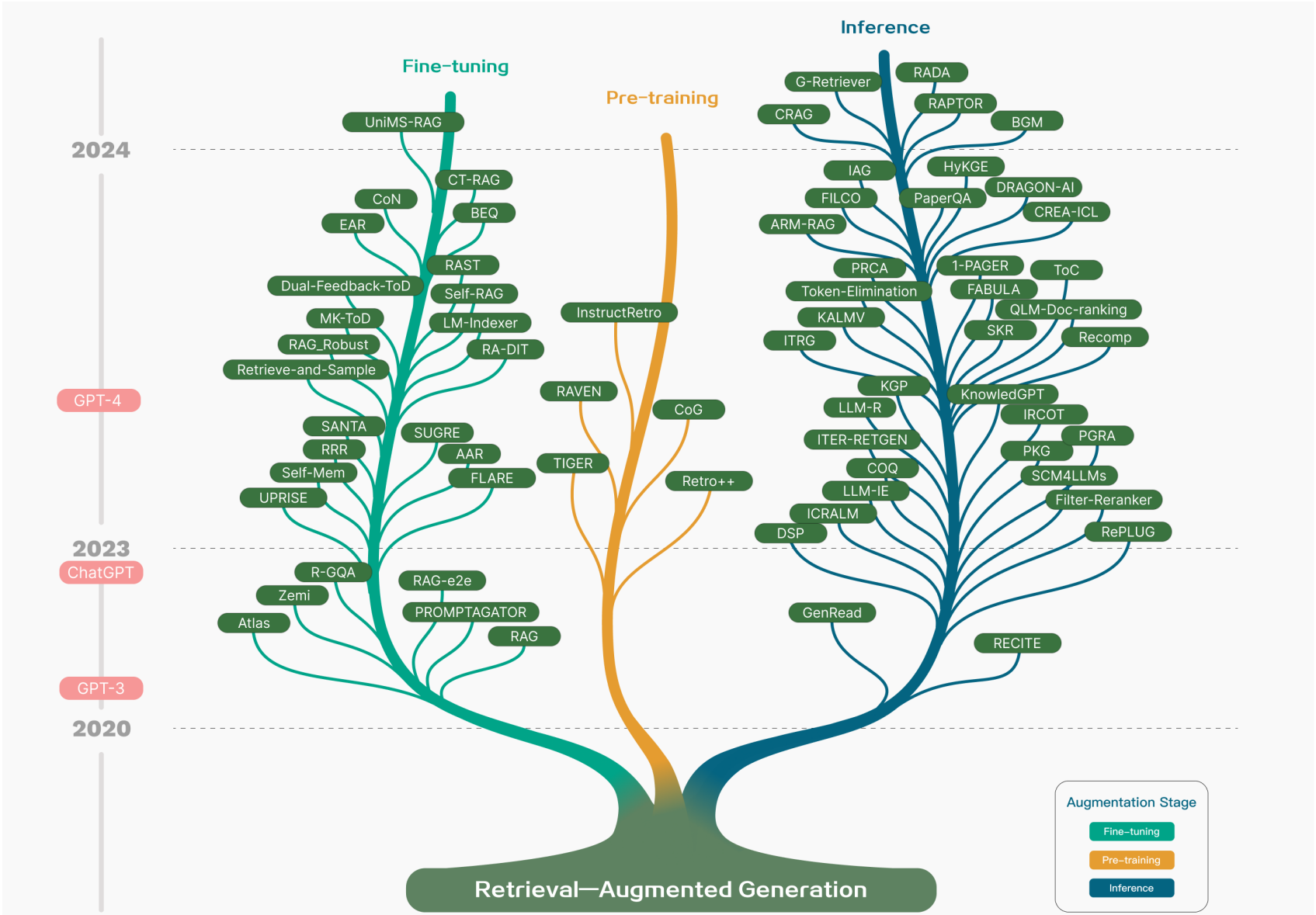



Retrieval Augmented Generation

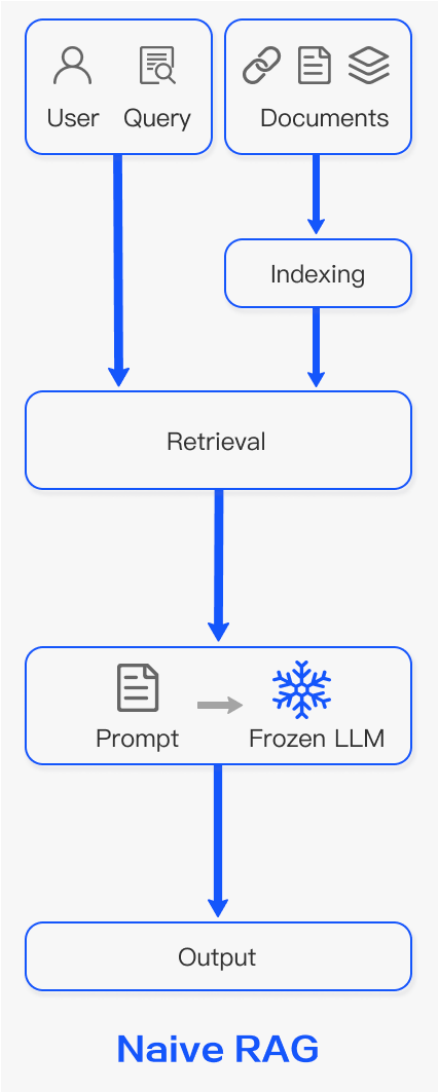
Retrieval Augmented Generation



Retrieval Augmented Generation

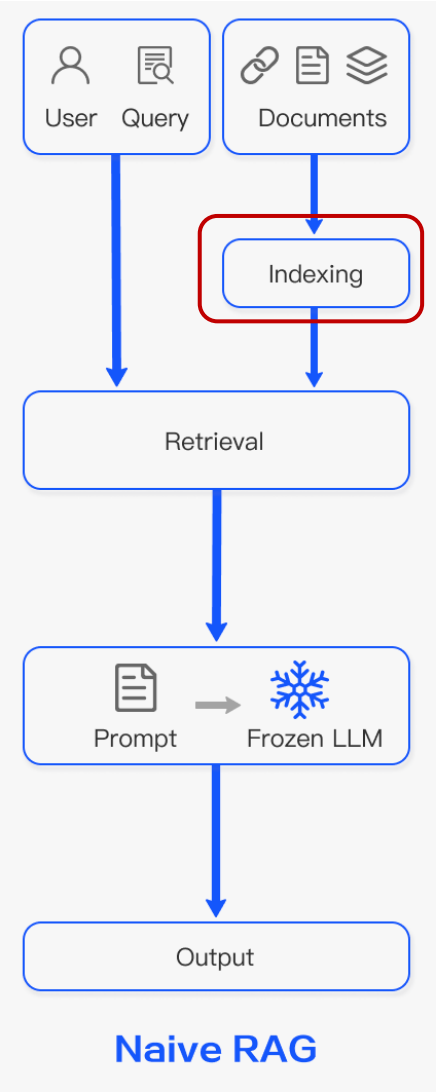


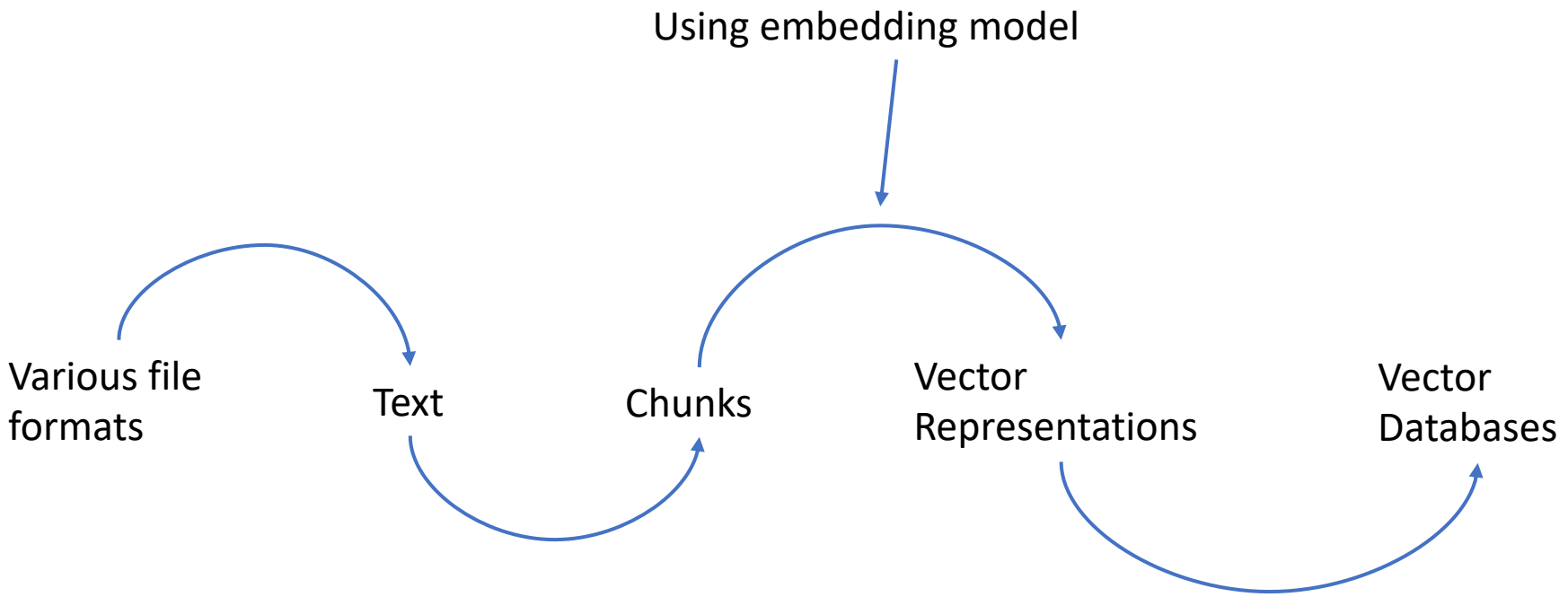
Naive RAG



Retrieve Read Framework

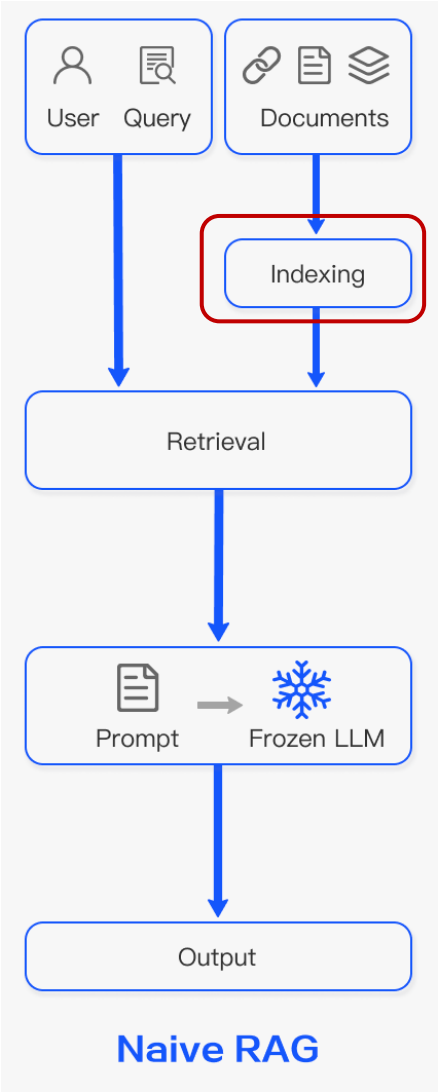
Naive RAG - Indexing





Retrieve Read Framework

Naive RAG - Indexing



Using embedding model

Various file formats → Text → Chunks → Vector Representations → Vector Databases

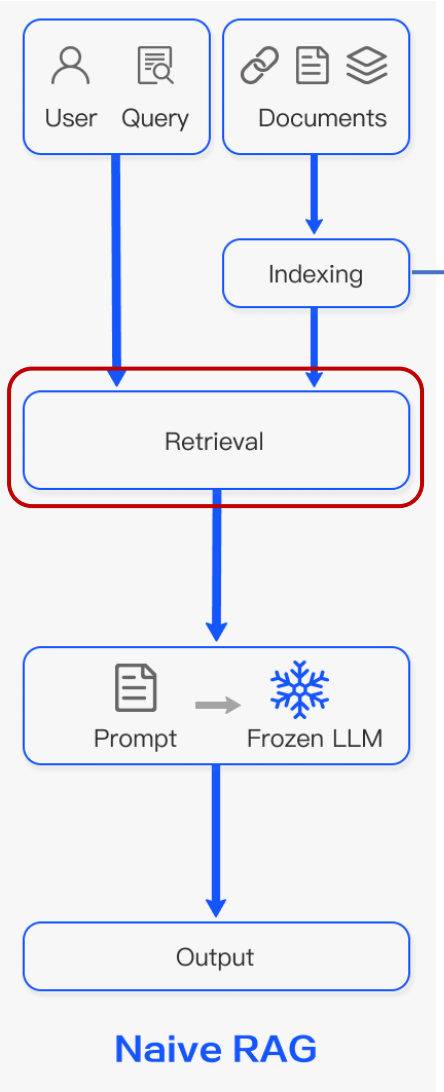
Pinecone

Chroma

Qdrant

Retrieve Read Framework

Naive RAG - Retrieval



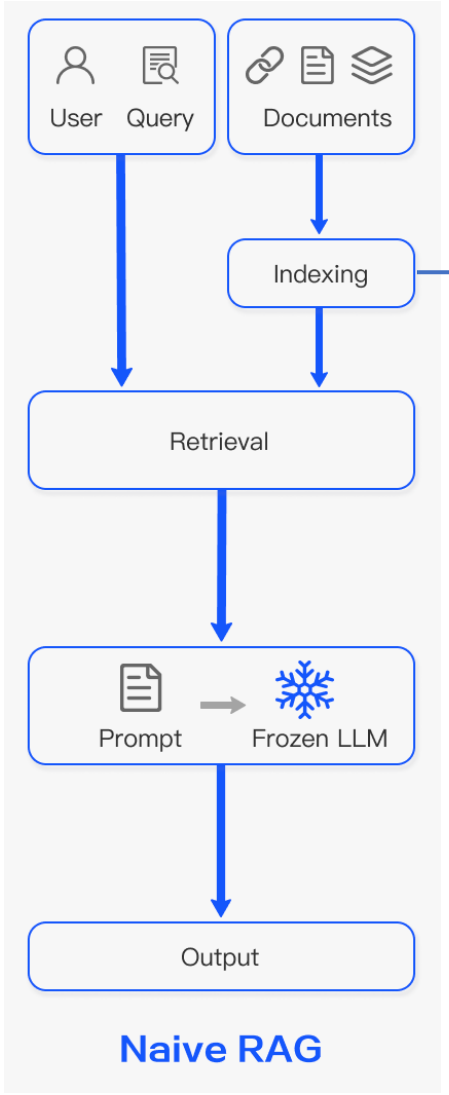
User Prompt → Indexing → Prompt Vector Representation → Indexed Corpus

Top-k similar chunks

Expanded Prompt Context

Retrieve Read Framework

Naive RAG - Generation



Retrieve Read Framework

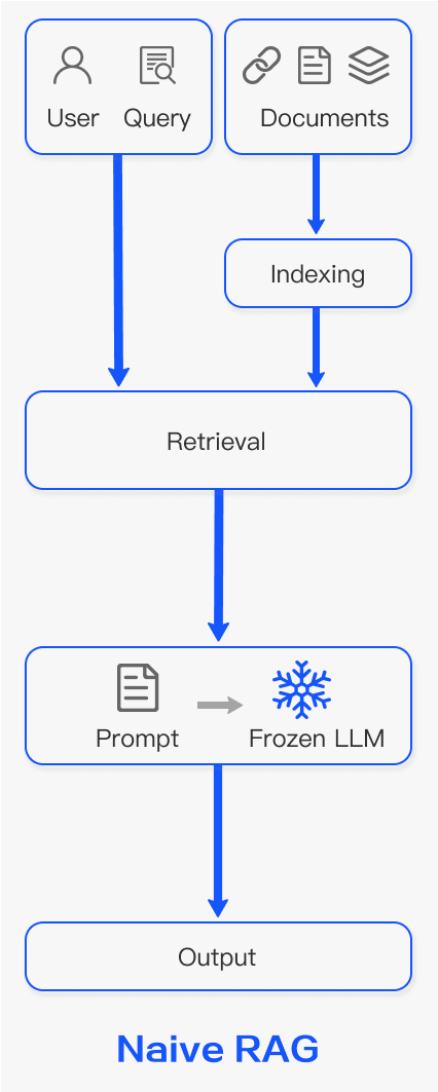
User Prompt → Indexing → Prompt Vector Representation → Indexed Corpus

Top-k similar chunks

Large Language Model ← Expanded Prompt Context

Coherent Prompt

Naive RAG - Drawbacks



Retrieval.

- Struggles with precision and recall:
  - Selection of misaligned or irrelevant chunks
  - Missing of crucial information.

Generation.

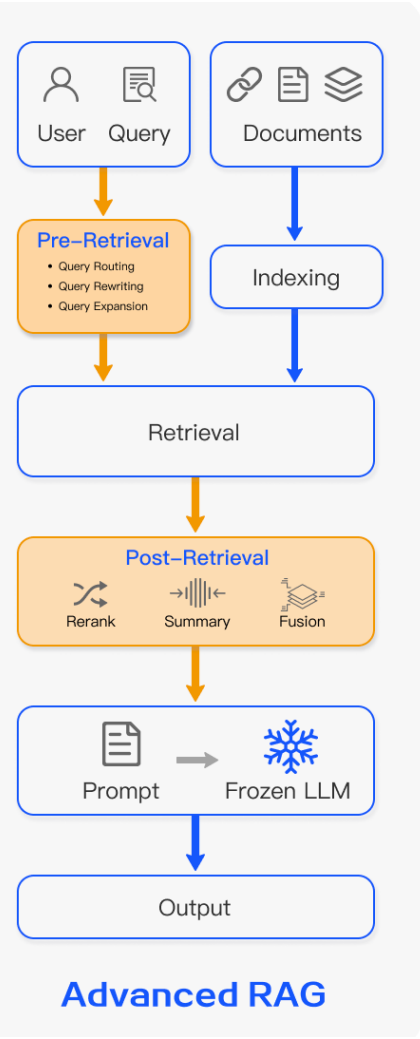
- Hallucinations
- Irrelevance
- Toxicity
- Bias

Augmentation.

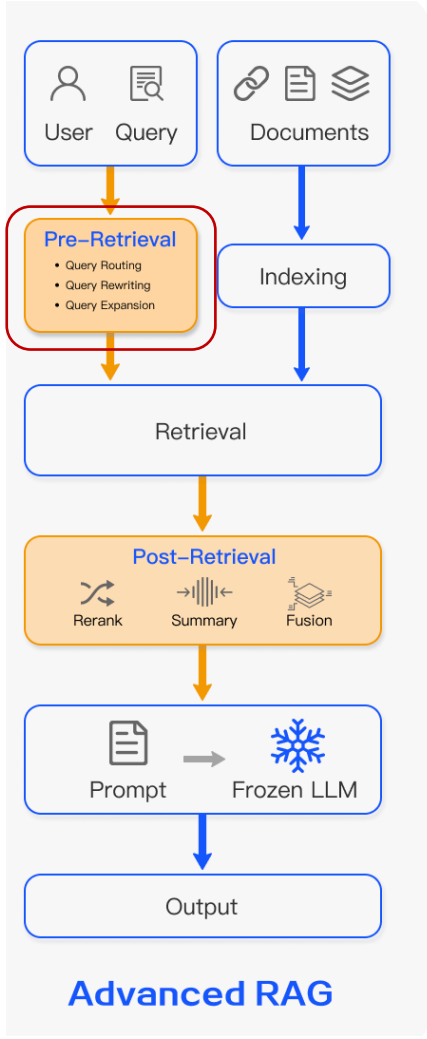
- Integrating retrieved information
- Redundancy
- Over reliance on augmented information

Retrieve Read Framework

Advanced RAG



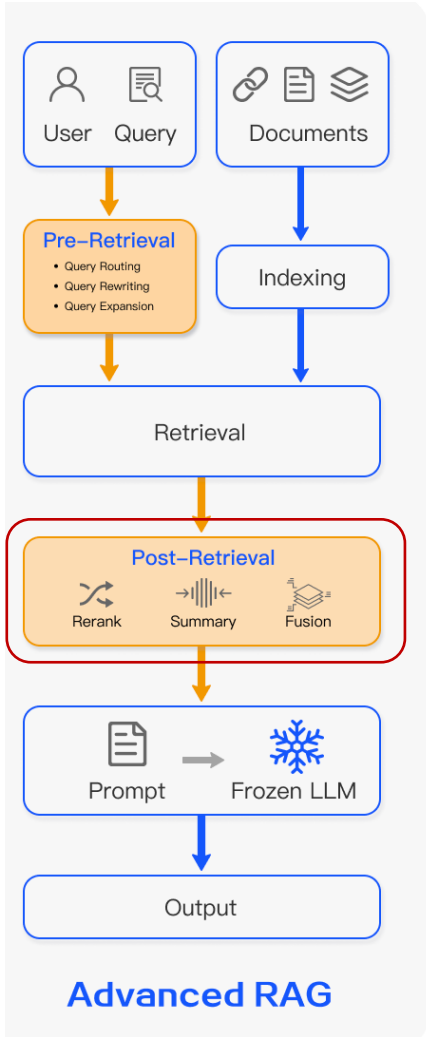
Advanced RAG



Pre-retrieval Process.

- Optimizing the indexing structure and the original query.
- Enhance the quality of the content being indexed
  - Enhancing data granularity
  - optimizing index structures
  - adding metadata
  - alignment optimization
- Query optimization
  Original question is clearer and more suitable for the retrieval task.
  - Query transformation
  - Query rewriting
  - Query expansion, etc.

Advanced RAG



Post-Retrieval Process.

Integrate retrieved context effectively with the query.

- Re-ranking
- Implementations: LlamaIndex, LangChain, and HayStack

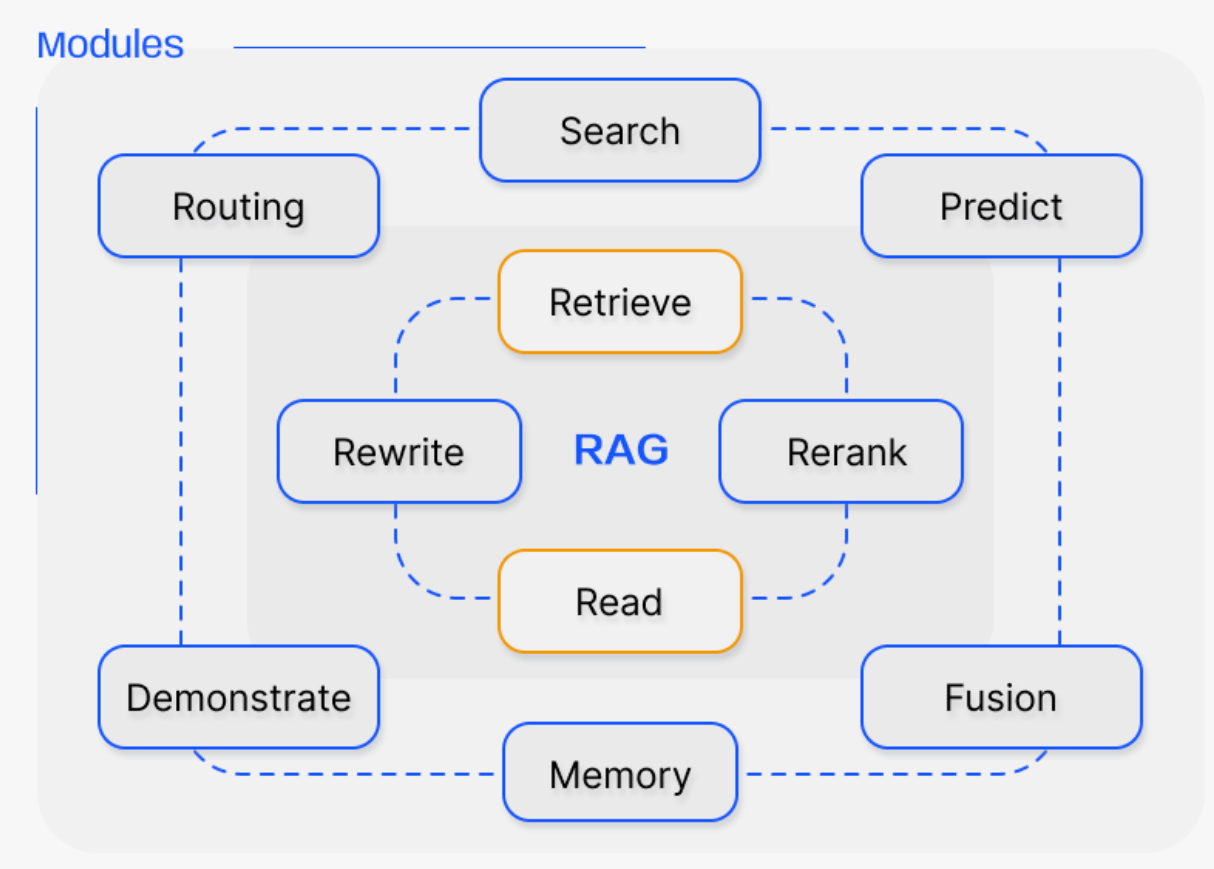
Drawback.

- Information overload
- Irrelevant content

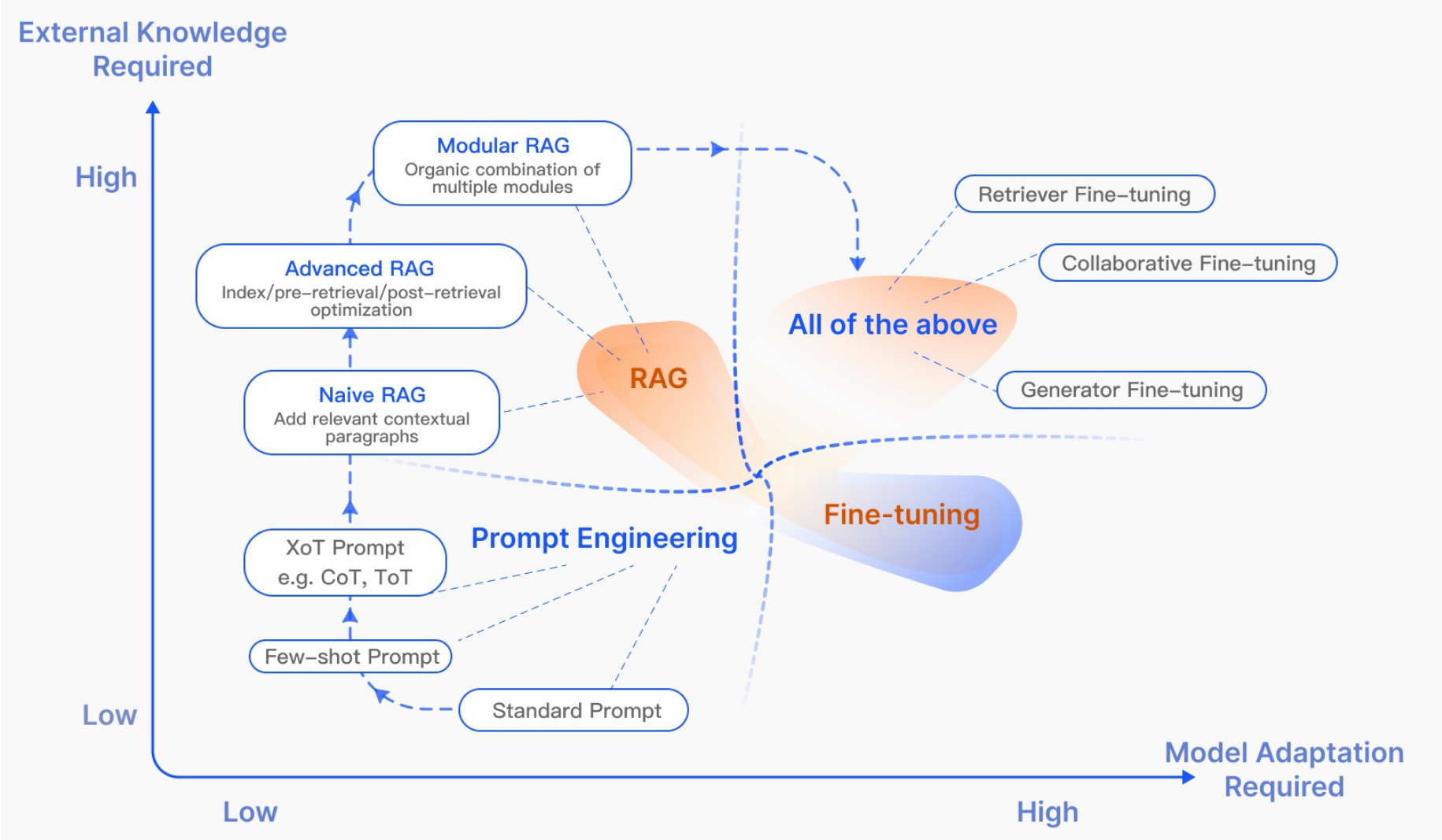
Mitigation.

- Selecting the essential information, emphasizing critical sections
- Shortening the context to be processed

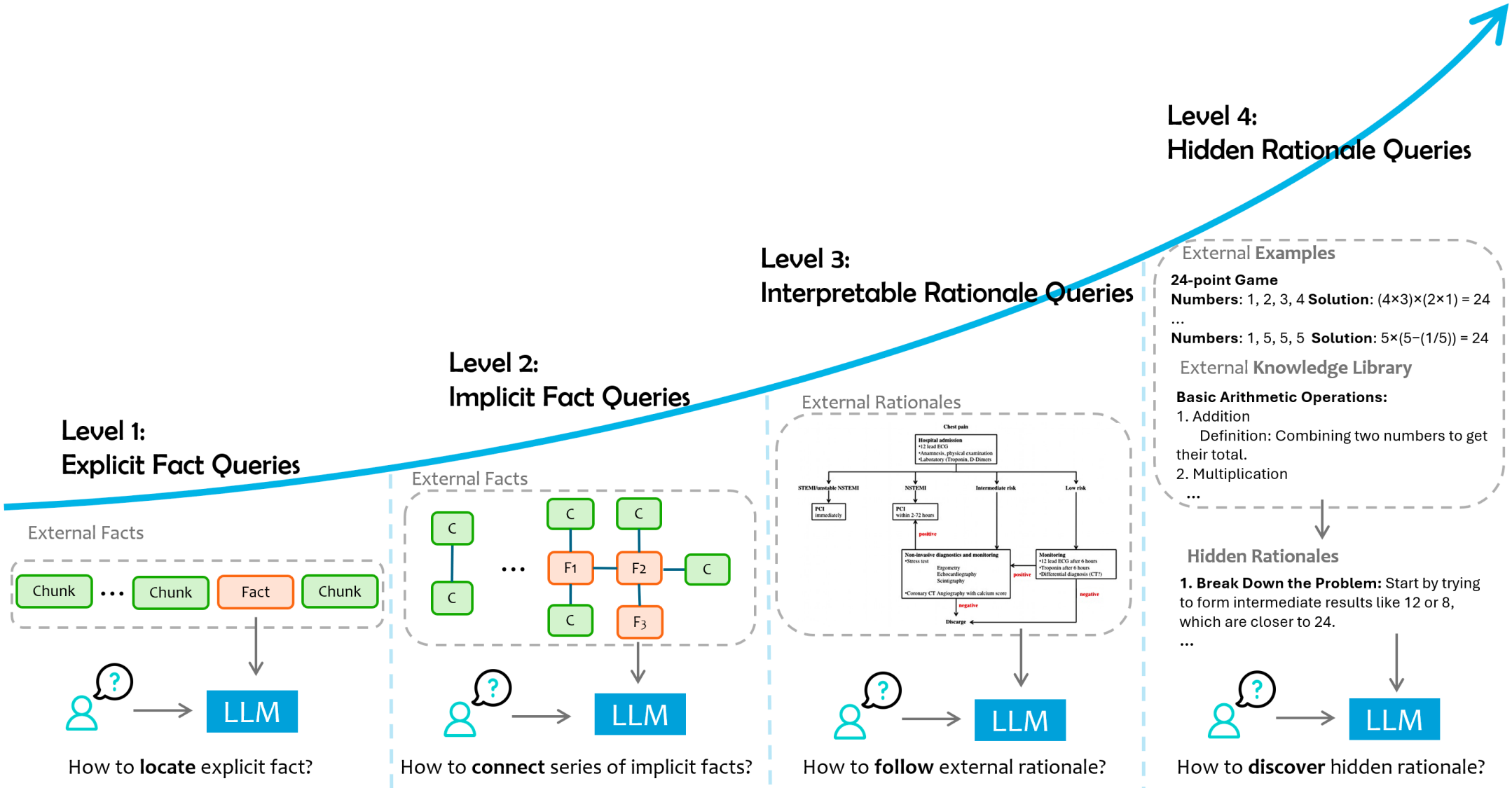
Modular RAG


Modules
Search
Routing
Predict
Retrieve
Rewrite
RAG
Rerank
Read
Demonstrate
Fusion
Memory

RAG - Comparison



Levels of Queries



Source: https://arxiv.org/abs/2409.14924

Level 1 – Explicit Facts

- Explicit facts directly present in the domain specific data
- Do not require any additional reasoning

Example.

“Where were the Summer Olympics 2024 held?”

“When did Keith Flint passed away?”

Level 1 – Explicit Facts

Challaneges and Solutions

Data Processing Difficulties

- Data is often highly unstructured
- contains multi-modal components such as tables, images, videos, and more
- Segmentation or "chunking" poses difficulties in retaining the original context and meaning

Data Retrieval Difficulties

- Retrieval of relevant data segments from a large, unstructured dataset can be computationally intensive and prone to errors

Evaluation Difficulties

- Requires robust metrics for assessing the quality of data retrieval and response generation

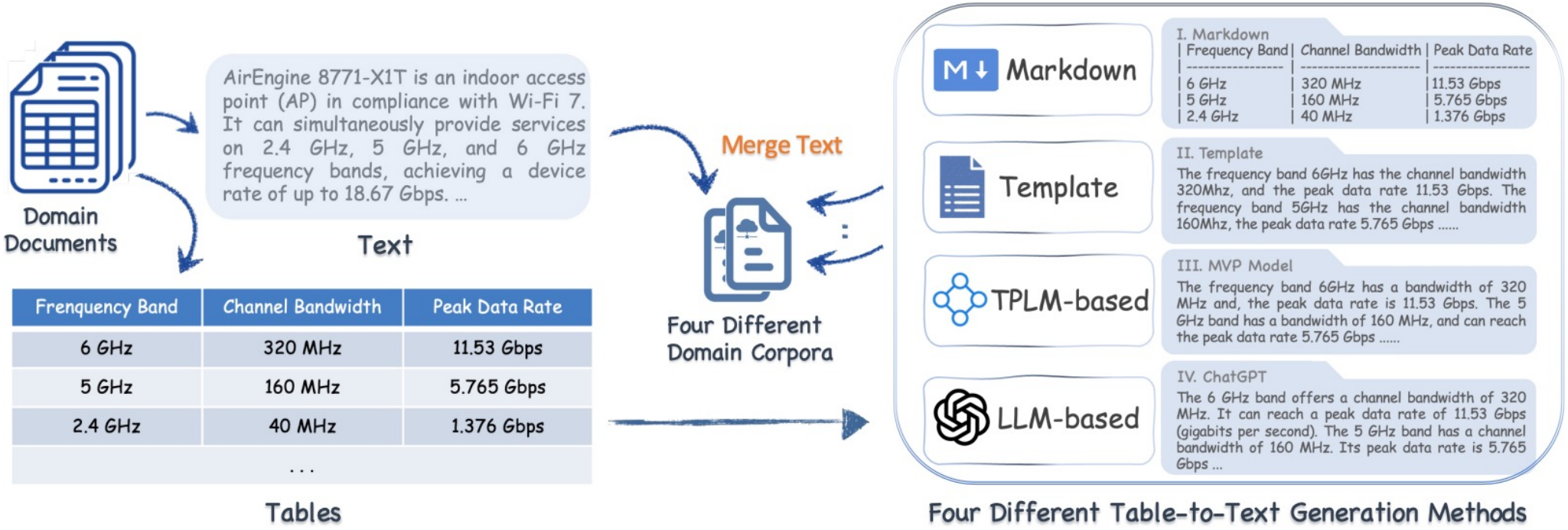
Source: https://aclanthology.org/2024.naacl-industry.41.pdf

Level 1 – Explicit Facts - RAG

Data Processing Enhancement

Multimodal Documents Parsing

- Converting multi-modal content into textual form
- Multimodal embeddings as soft prompts for input



Level 1 – Explicit Facts - RAG

Data Processing Enhancement

Multimodal Documents Parsing

- Converting multi-modal content into textual form
- Multimodal embeddings as soft prompts for input

Source: https://multimodal-kdd-2023.github.io/papers/2-2%20REVEAL.pdf

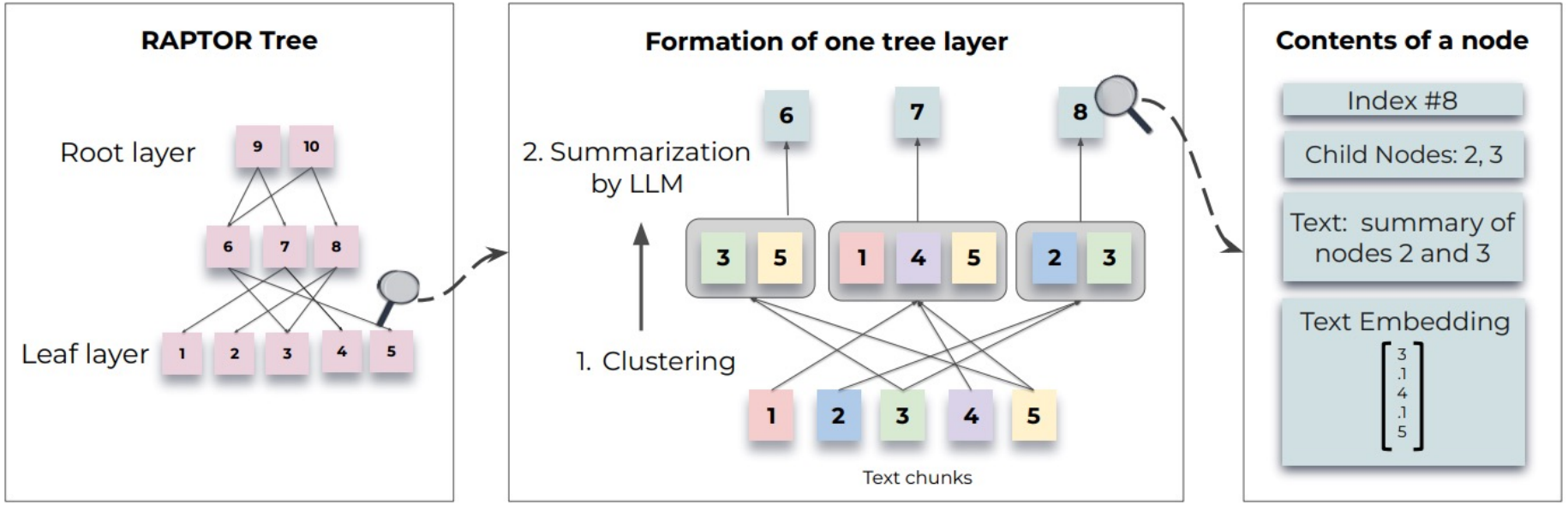
Source: https://arxiv.org/pdf/2401.18059

Level 1 – Explicit Facts - RAG

Data Processing Enhancement

Chunking Optimization

- Fixed size chunking, recursive chunking, sliding window chunking, paragraph-based chunking, semantic chunking, etc.
- Refine the text into smaller segments that maintain a high degree of information completeness



Level 1 – Explicit Facts - RAG

Data Retrieval Enhancement - Indexing

- Sparse Retrieval
- Dense Retrieval
- Other Methods

Sparse Retrieval

- TF-IDF and BM25
- Word matching methods
- Similarity-based matching (KNN)

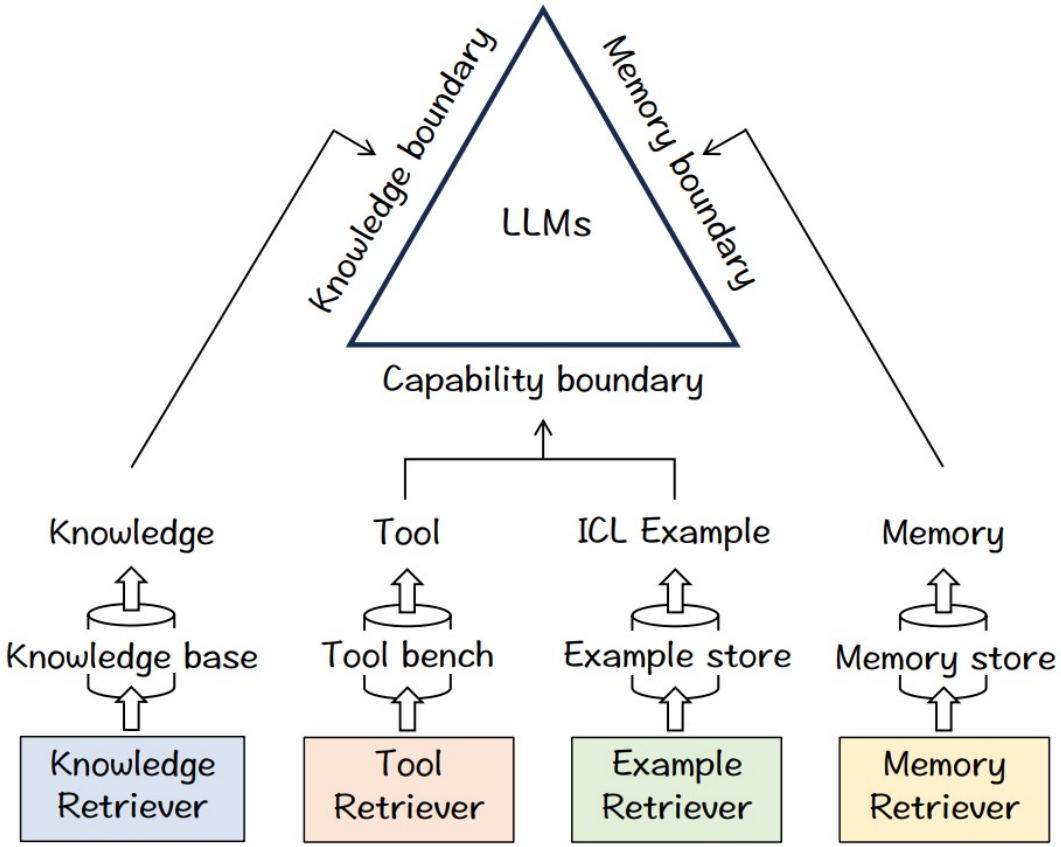
Level 1 – Explicit Facts - RAG

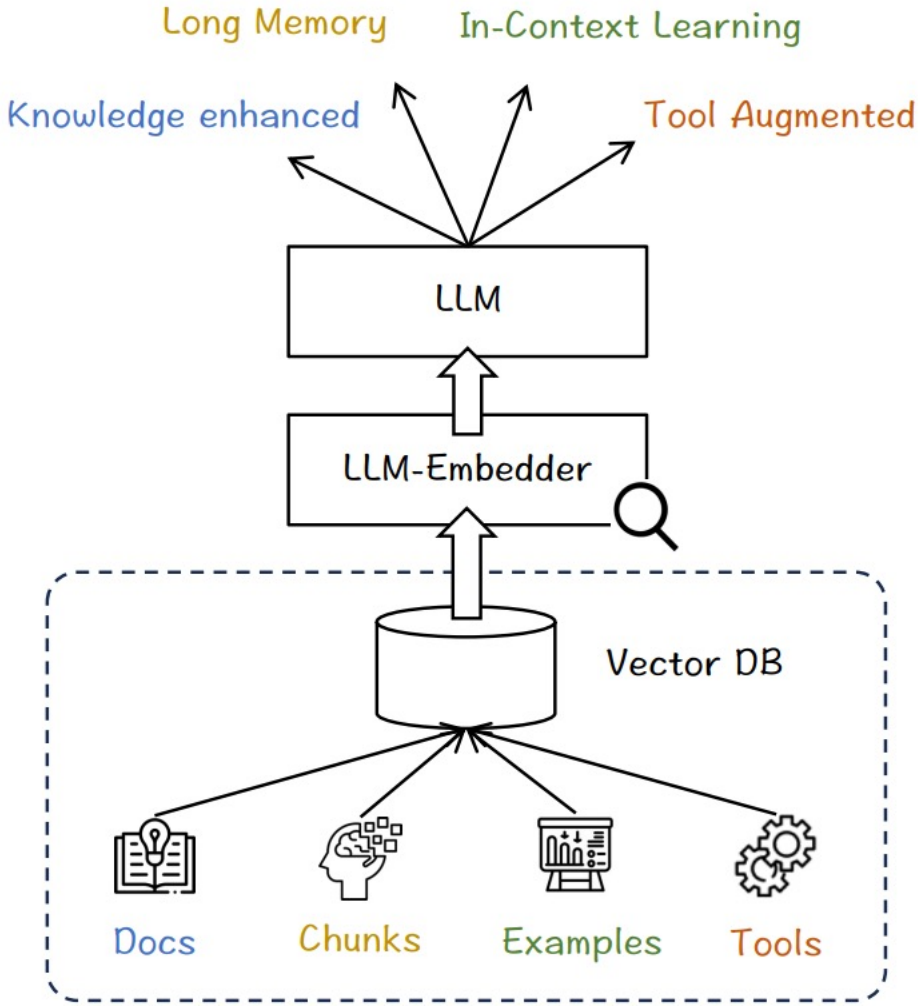
Data Retrieval Enhancement - Indexing

Dense Retrieval

- BERT-based encoders
- unsupervised contrastive learning for fine-tuning
- feedback from LLMs to guide the training objectives of retrievers
- LLM-based dense retrieval
  - LLM2Vec
  - Llama2Vec

Level 1 – Explicit Facts - RAG





Source: https://arxiv.org/pdf/2310.07554

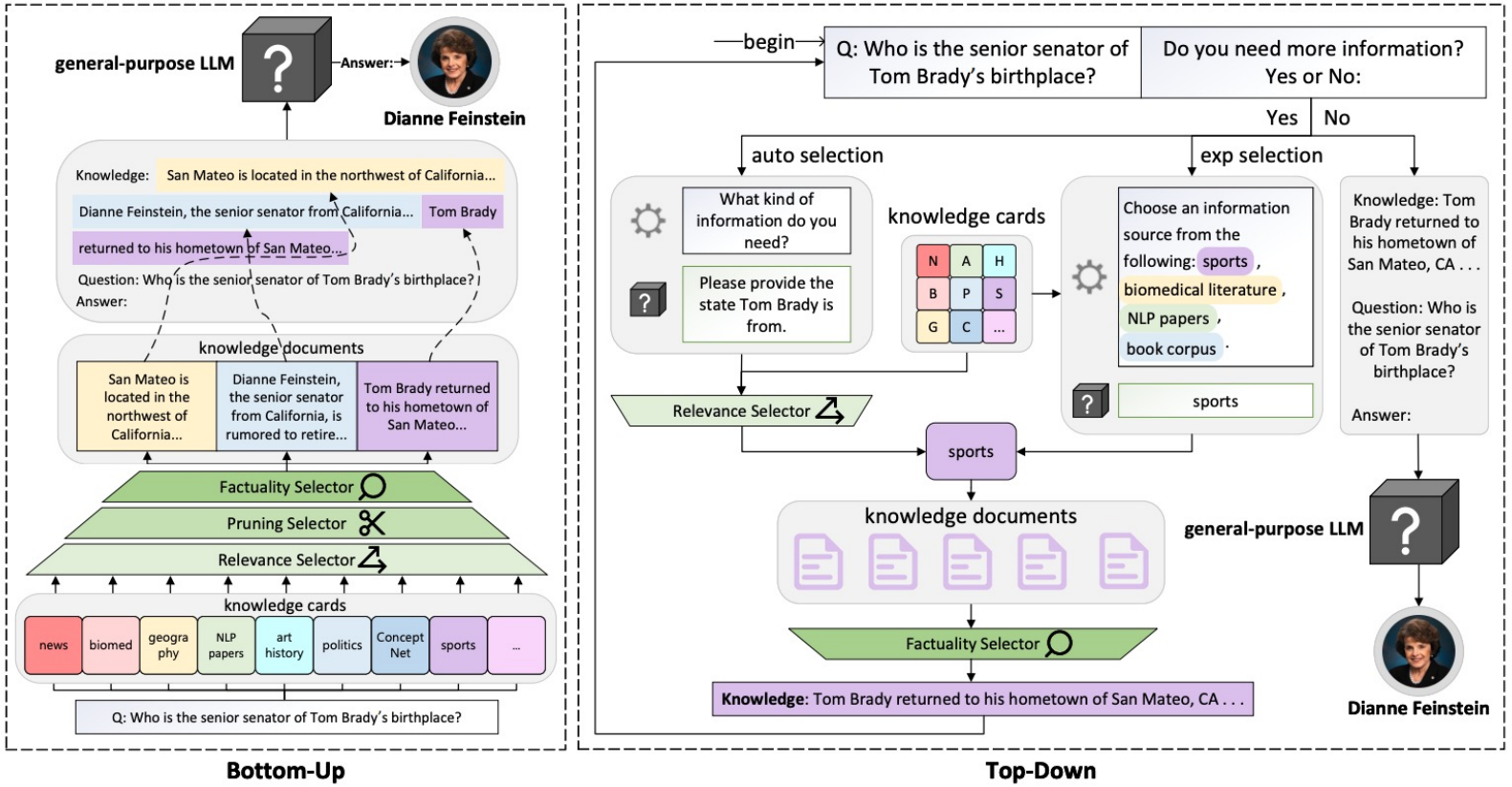
Source: https://arxiv.org/pdf/2305.09955

Level 1 – Explicit Facts - RAG

Data Retrieval Enhancement - Indexing

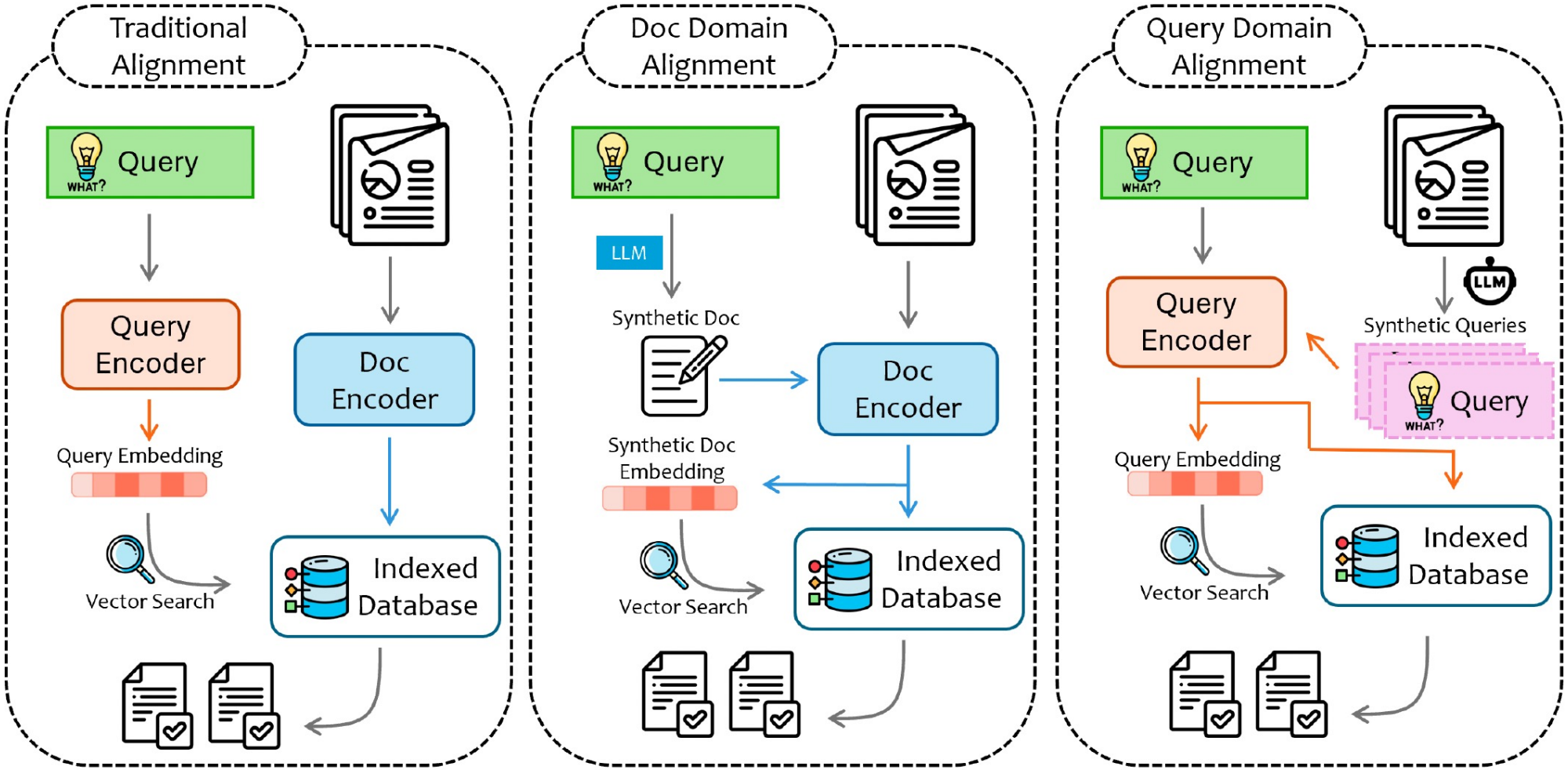
Others

- Initially determining the knowledge domain needed to answer a query as a fixed area of expertise, and then using dense retrieval to recall supplementary information within this domain



Level 1 – Explicit Facts - RAG

Data Retrieval Enhancement – Query Document Alignment



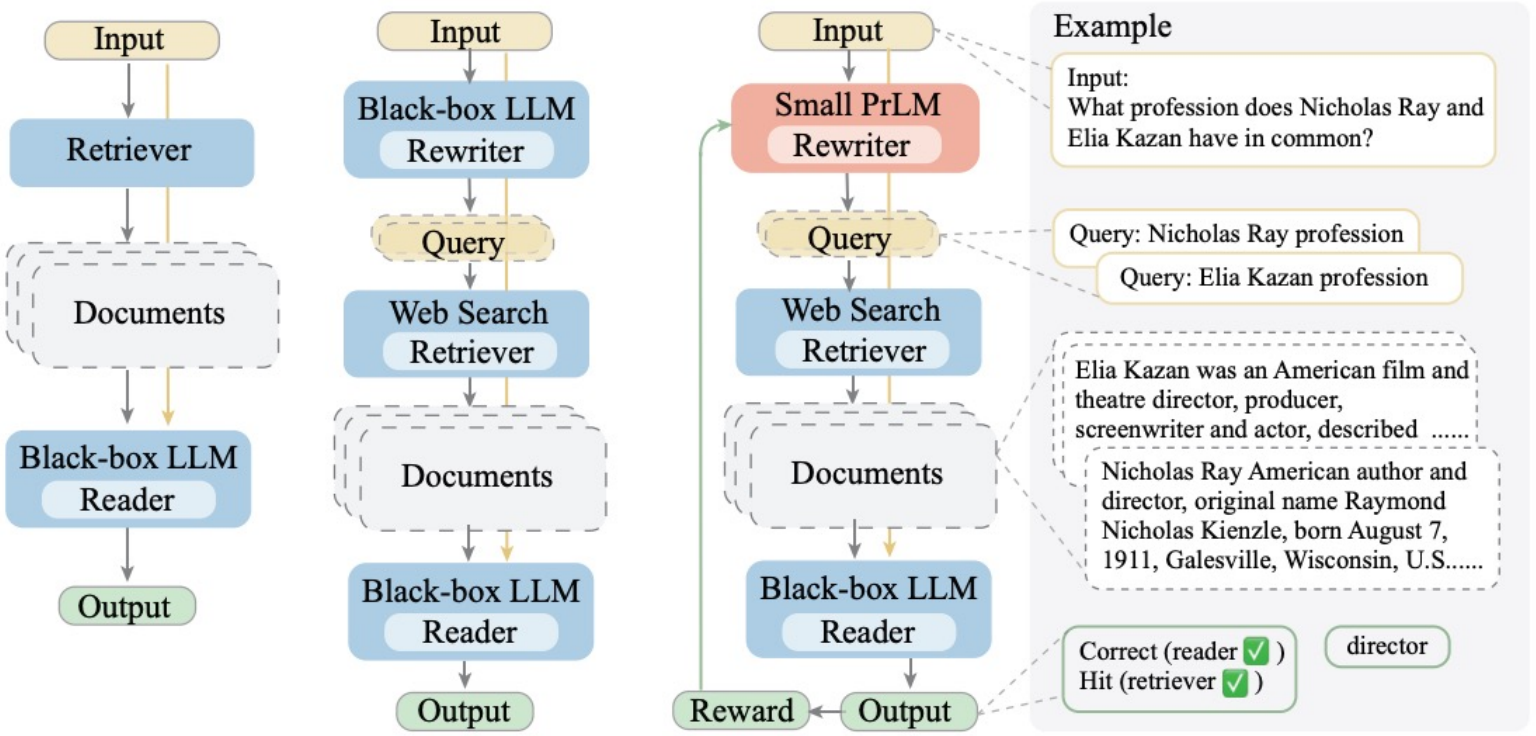
Source: https://aclanthology.org/2023.emnlp-main.322.pdf

Level 1 – Explicit Facts - RAG

Data Retrieval Enhancement – Query Document Alignment

Traditional Alignment

- Mapping both document segments and the query into the same encoding space.



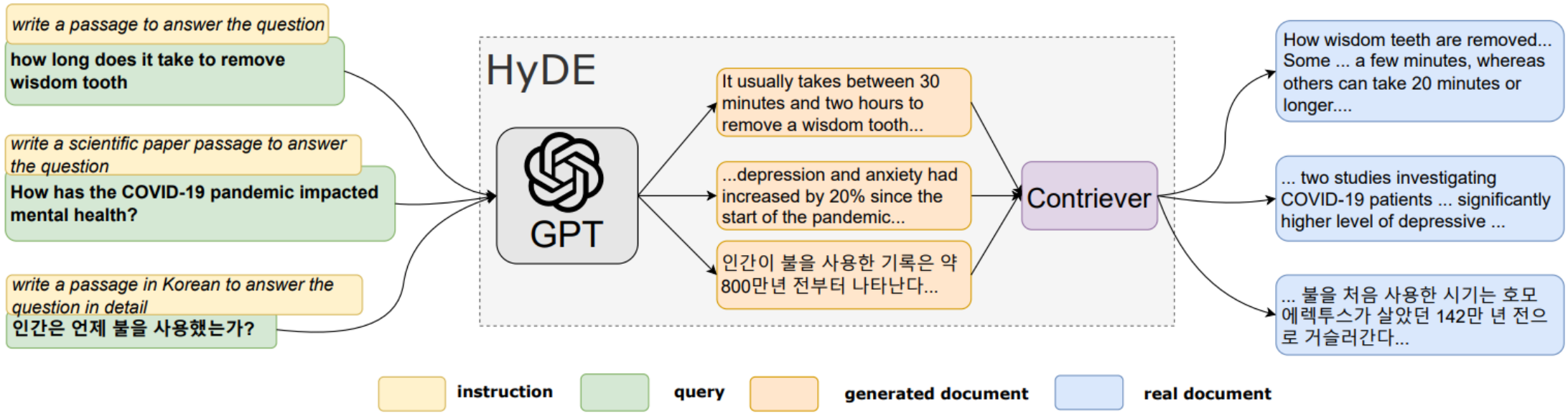
(a) Retrieve-then-read (b)Rewrite-retrieve-read (c) Trainable rewrite-retrieve-read

Level 1 – Explicit Facts - RAG

Data Retrieval Enhancement – Query Document Alignment

Document Domain Alignment

- generating synthetic answers first, then using these answers to recall relevant data



Source: https://aclanthology.org/2023.acl-long.99.pdf

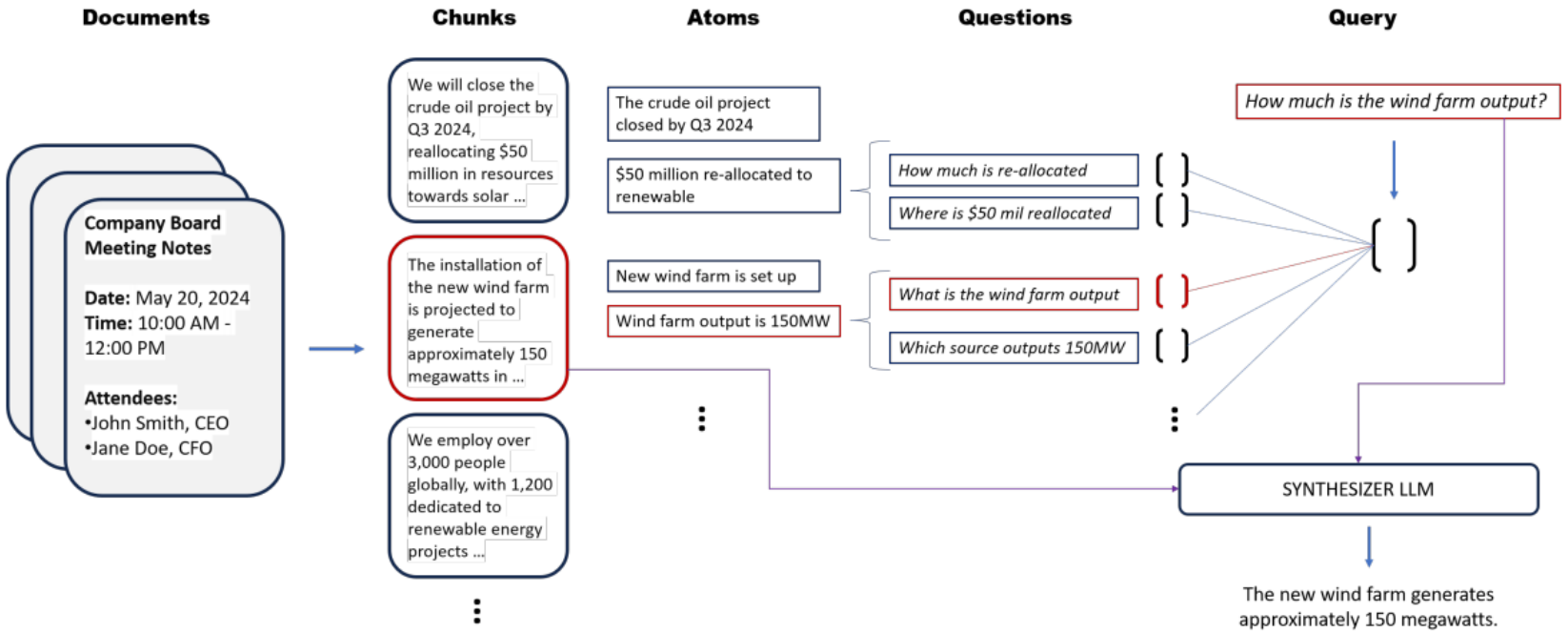
Source: https://aclanthology.org/2024.fever-1.25.pdf

Level 1 – Explicit Facts - RAG

Data Retrieval Enhancement – Query Document Alignment

Query Domain Alignment

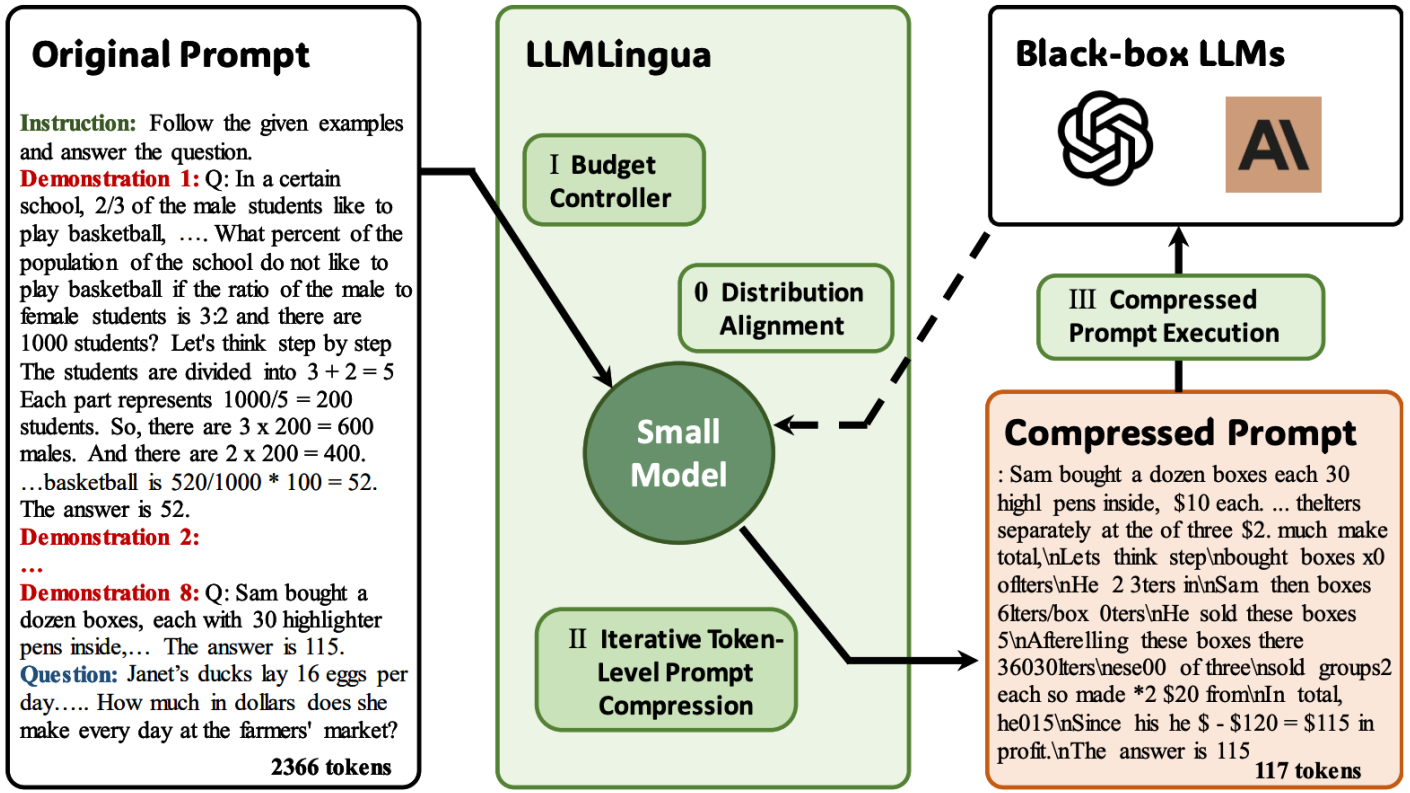
- generating a set of synthetic questions for each atomic unit of text,
- mapping text segments into the query space,
- retrieving the synthetic questions closest to the original query along with their corresponding text segments.



Level 1 – Explicit Facts - RAG

Data Retrieval Enhancement – Re-ranking & Correction

- Filter and reorder the segments
- Relevance scores for re-ranking
- perplexity or perplexity gain as ranking criteria



Source: https://arxiv.org/pdf/2310.05736

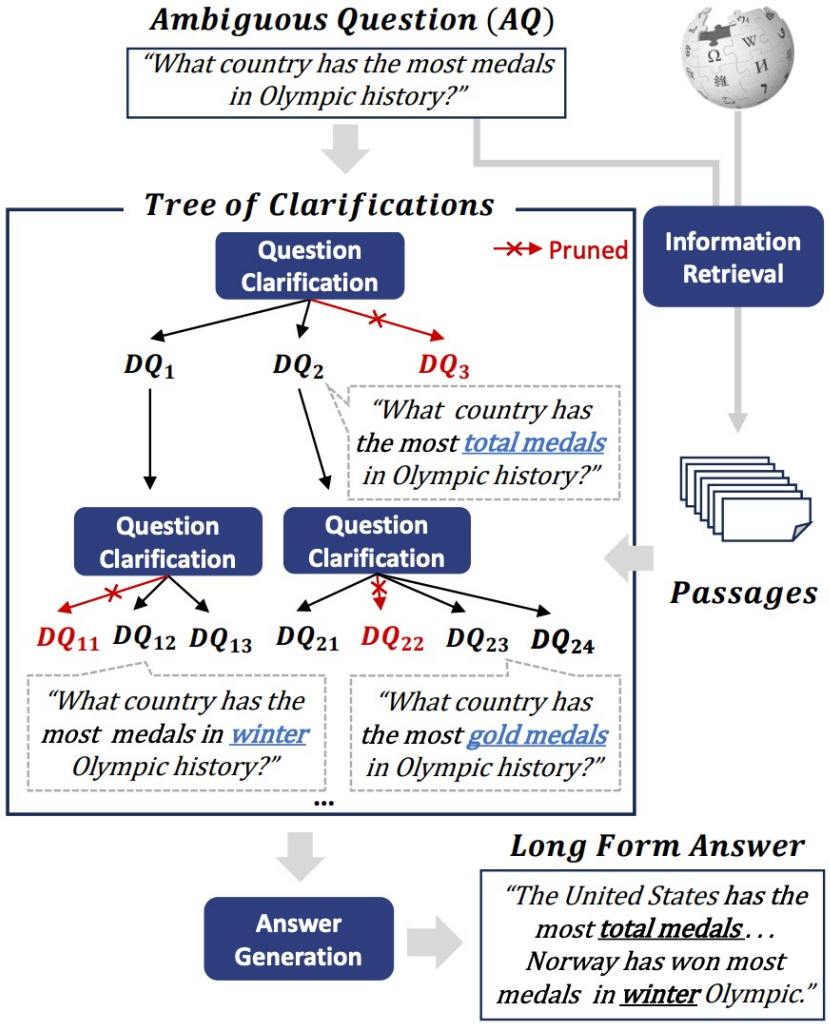
Level 1 – Explicit Facts - RAG

Data Retrieval Enhancement – Recursive Retrieval of Iterative Retrieval

- Overcome the drawbacks of single retrieval
- Tree-like recursive retrieval method
  - Incorporating pruning strategies to incrementally break down ambiguous questions
  - ultimately arriving at the closest correct answer
- SEATER
  - K-means algorithm to construct a hierarchical tree structure of items

Level 1 – Explicit Facts - RAG

Data Retrieval Enhancement – Recursive Retrieval of Iterative Retrieval



Source: https://aclanthology.org/2023.emnlp-main.63.pdf

Level 1 – Explicit Facts - RAG

Response Generation Enhancement

- Determining if the retrieved information is sufficient or if additional external data is needed
- Handling conflicts between retrieved knowledge and the model's internal prior knowledge
- Supervised Fine-Tuning
- Training data with irrelevant retrieval noise, relevant retrieval noise, and counterfactual retrieval noise
- Joint training of both retriever and generator

Level 2 – Implicit Facts

Overview.

- Requires common sense reasoning or basic logical deductions
- Information from multiple segments or require simple inferencing

"What is the majority party now in the country where Canberra is located?”

- Decomposing the original query into multiple retrieval operations
- The aggregation of results

Example.

- Statistical queries, descriptive analysis queries, and basic aggregation queries
- Counting, comparison, trend analysis, and selective summarization are common in "how many" and "what’s the most” type queries (multi-hop queries)

Level 2 – Implicit Facts

Challenges and Solutions.

Adaptive Retrieval Volumes

- Questions require varying numbers of retrieved contexts
- Specific number of retrieved contexts can depend on both the question and the dataset
- Fixed number of retrievals may result in either information noise or insufficient information

Coordination between Reasoning and Retrieval

- Reasoning can guide retrieval
- Insights from retrieved information can iteratively refine reasoning strategies

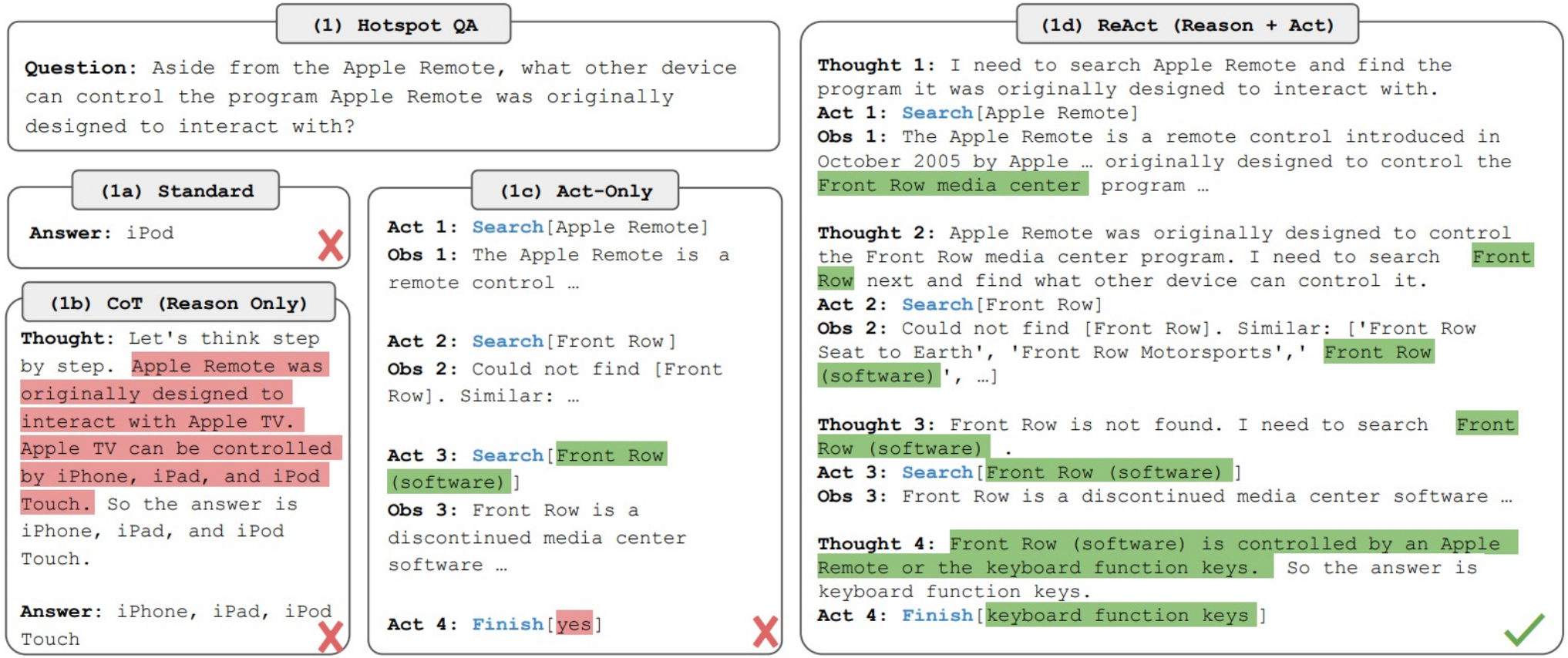
Source: https://arxiv.org/pdf/2210.03629

Level 2 – Implicit Facts

Iterative RAG.

Planning Based

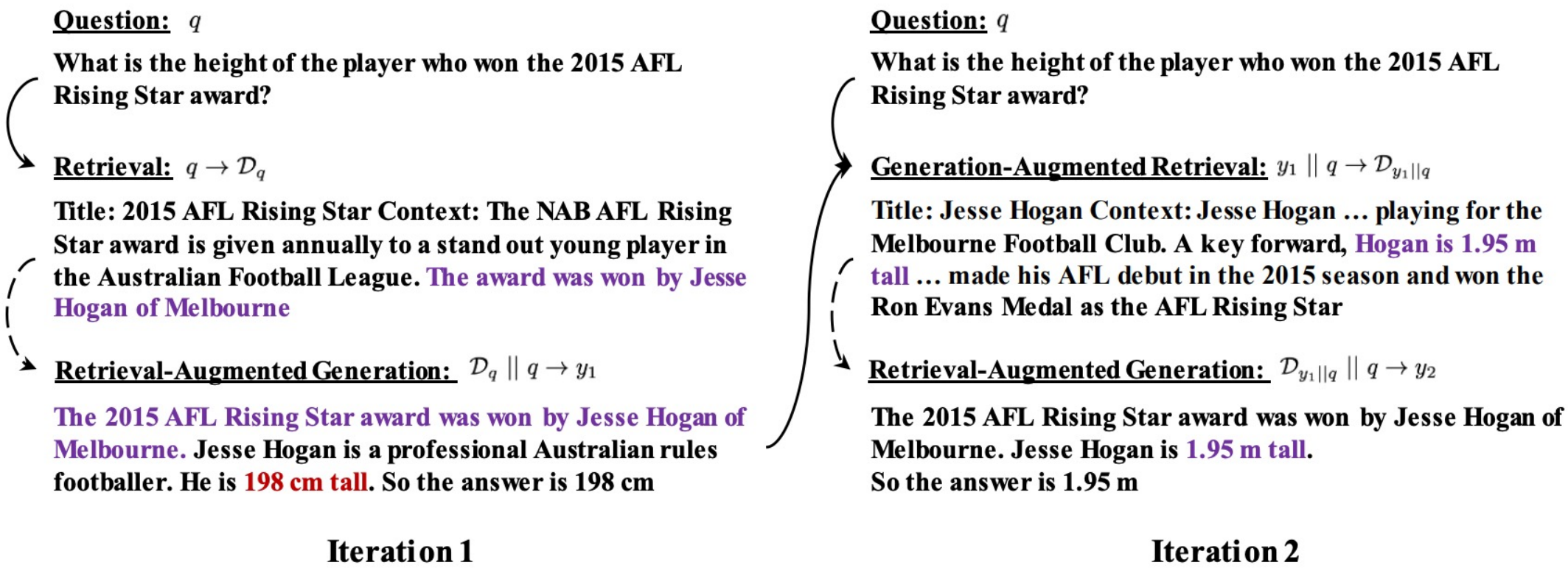
- Generating stepwise retrieval plan during the prior retrieval stage or dynamically within the retrieval process



Level 2 – Implicit Facts

Iterative RAG.

Information Gap Filling Based



Source: https://aclanthology.org/2023.findings-emnlp.620.pdf

Level 3 – Interpretable Rationales

Challenges and Solutions.

Prompt Optimization Cost

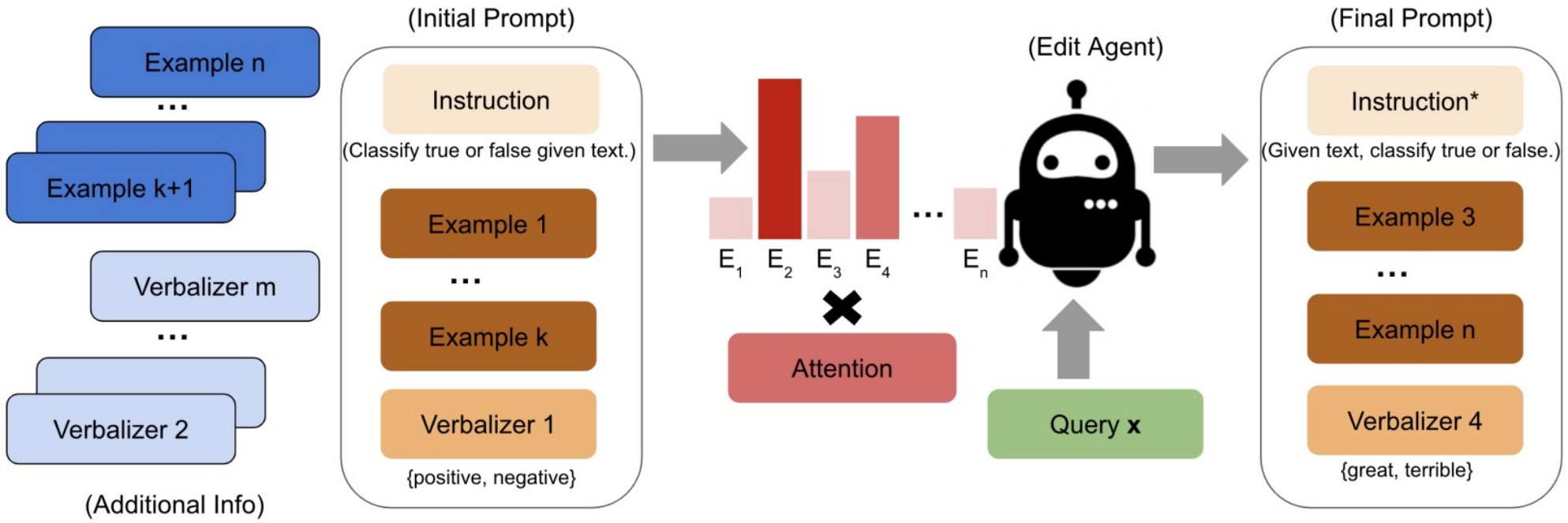
- Queries demand tailored background knowledge and decision-making criteria
- Manually designed prompts can be highly effective but are labor-intensive and time consuming
- Training models to generate tailored prompts incurs significant computational overhead

Limited Interpretability

- Impact of prompts on LLMs is opaque
- Access to the internal parameters of LLMs is restricted

Level 3 – Interpretable Rationales

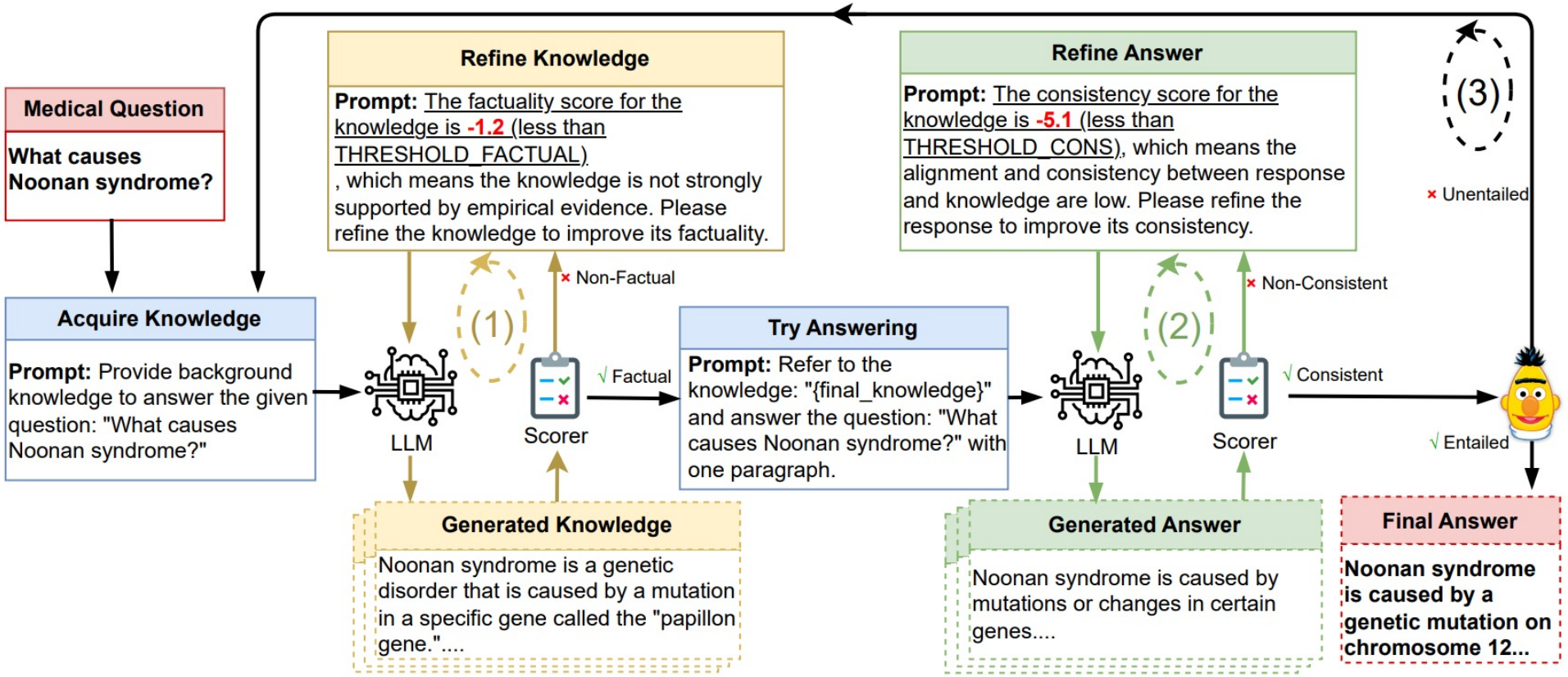
Prompt Tuning.



Source: https://arxiv.org/pdf/2211.11890

Level 3 – Interpretable Rationales

Chain of Thought.



Source: https://aclanthology.org/2023.findings-emnlp.123.pdf

Level 4 – Hidden Rationales

Overview.

In-domain Data

- utilize data from the same domain
- historical question-and-answer records or artificially generated data.
- contains the reasoning skills or methodologies
- e.g., Python programming puzzles

Preliminary Knowledge

- extensive, dispersed knowledge bases
- comprehensive axiomatic system
- e.g., all local legal codes that form the basis for legal judgments
- intermediate conclusions that simplify reasoning processes
- e.g., mathematical proofs

Level 4 – Hidden Rationales

Challenges and Solutions.

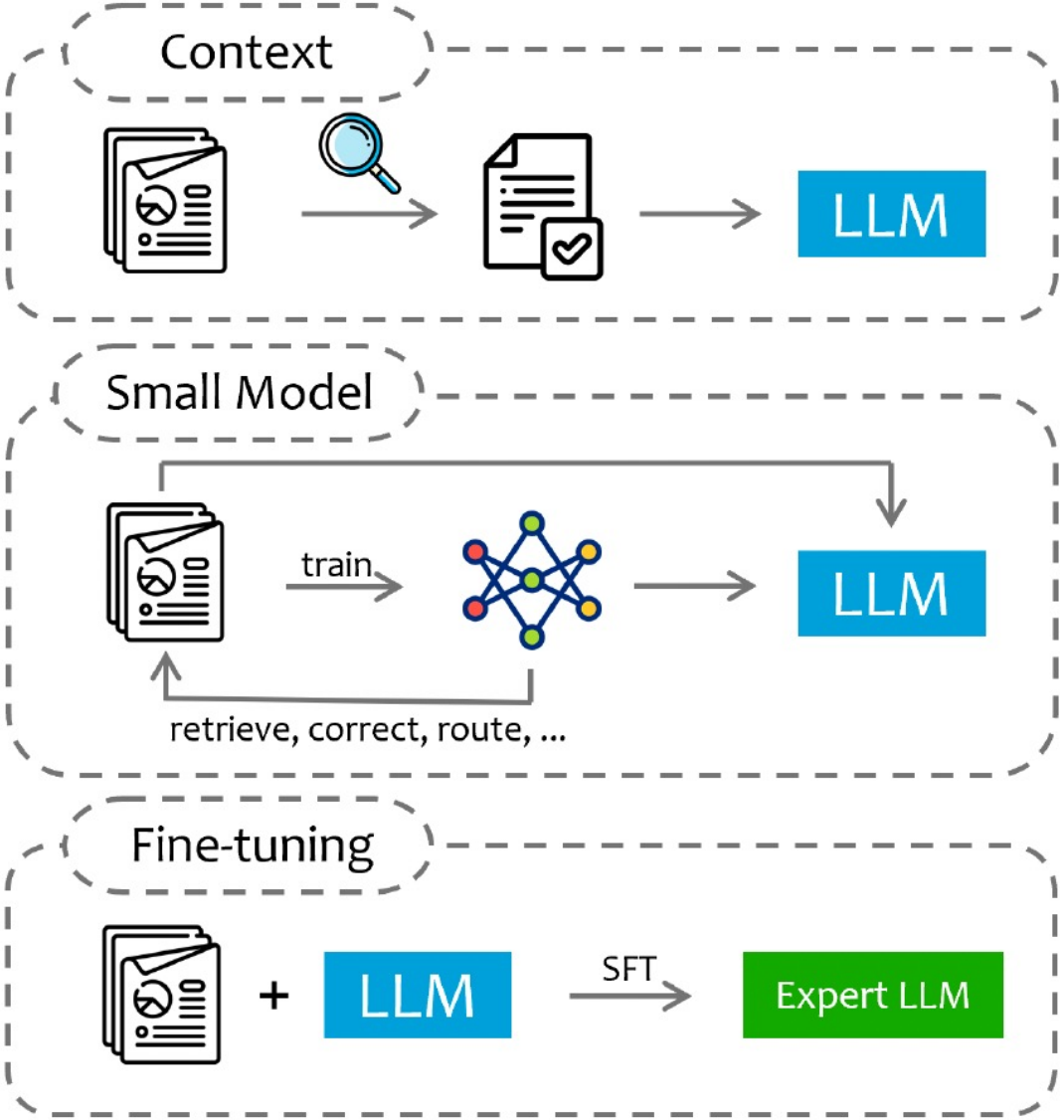
Logical Retrieval

- Struggle to capture the true target of the query
- Need for parse and identify underlying logical structures

Data Insufficiency

- Relevant information is often dispersed knowledges
- Demands robust capabilities in data interpretation and synthesis

Level 4 – Hidden Rationales



Evaluating RAG Systems

Metrics for Evaluating RAG Systems

- Context Precision: the proportion of relevant chunks in the retrieved contexts
- Context Recall: how many of the relevant documents were successfully retrieved
- Noise Sensitivity: how often a system makes errors by providing incorrect responses when utilizing either relevant or irrelevant retrieved documents
- Response Relevancy: assessing how pertinent the generated answer is to the given prompt
- Faithfulness: factual consistency of the generated answer against the given context

Source: https://docs.ragas.io/en/latest/getstarted/