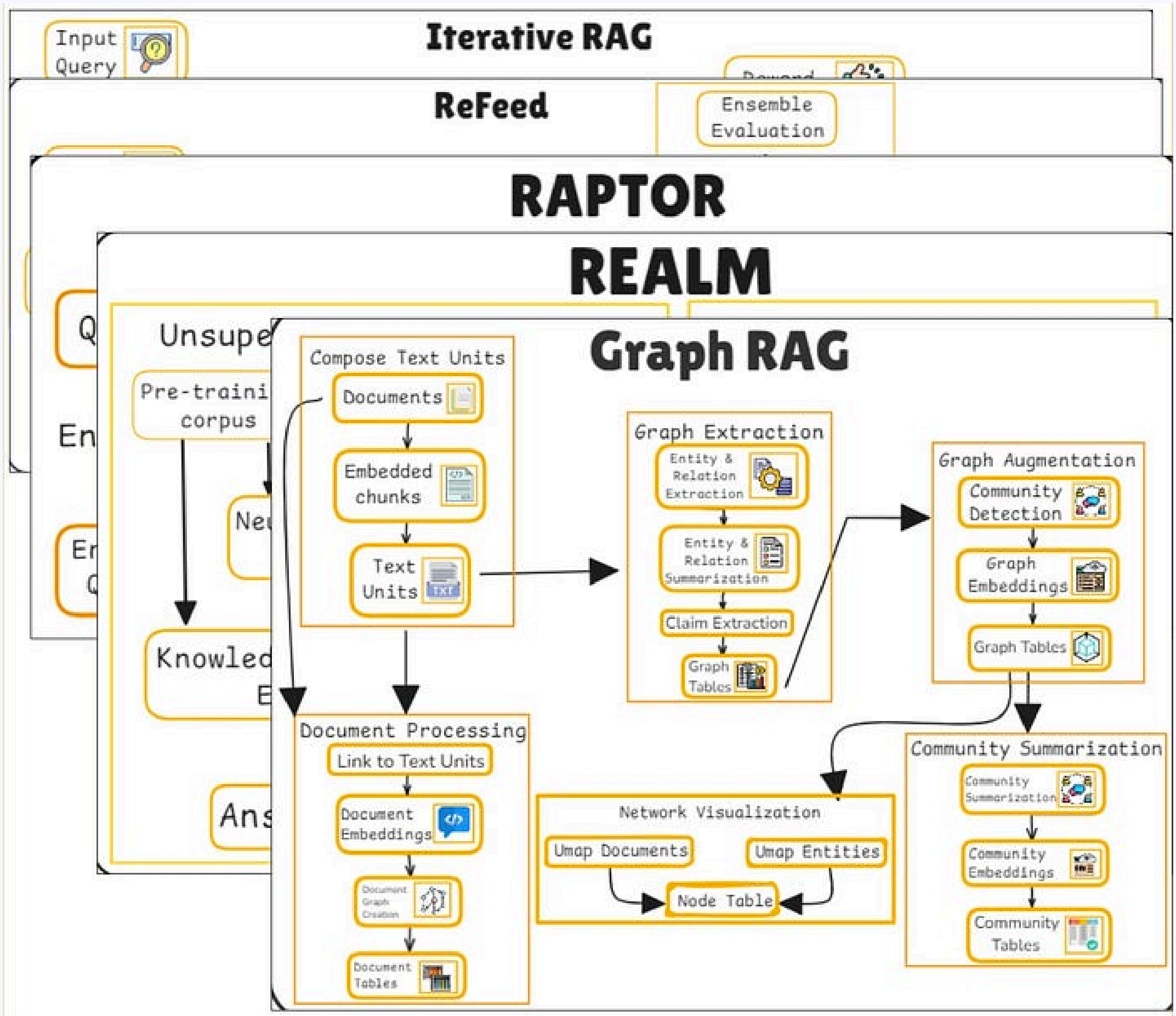
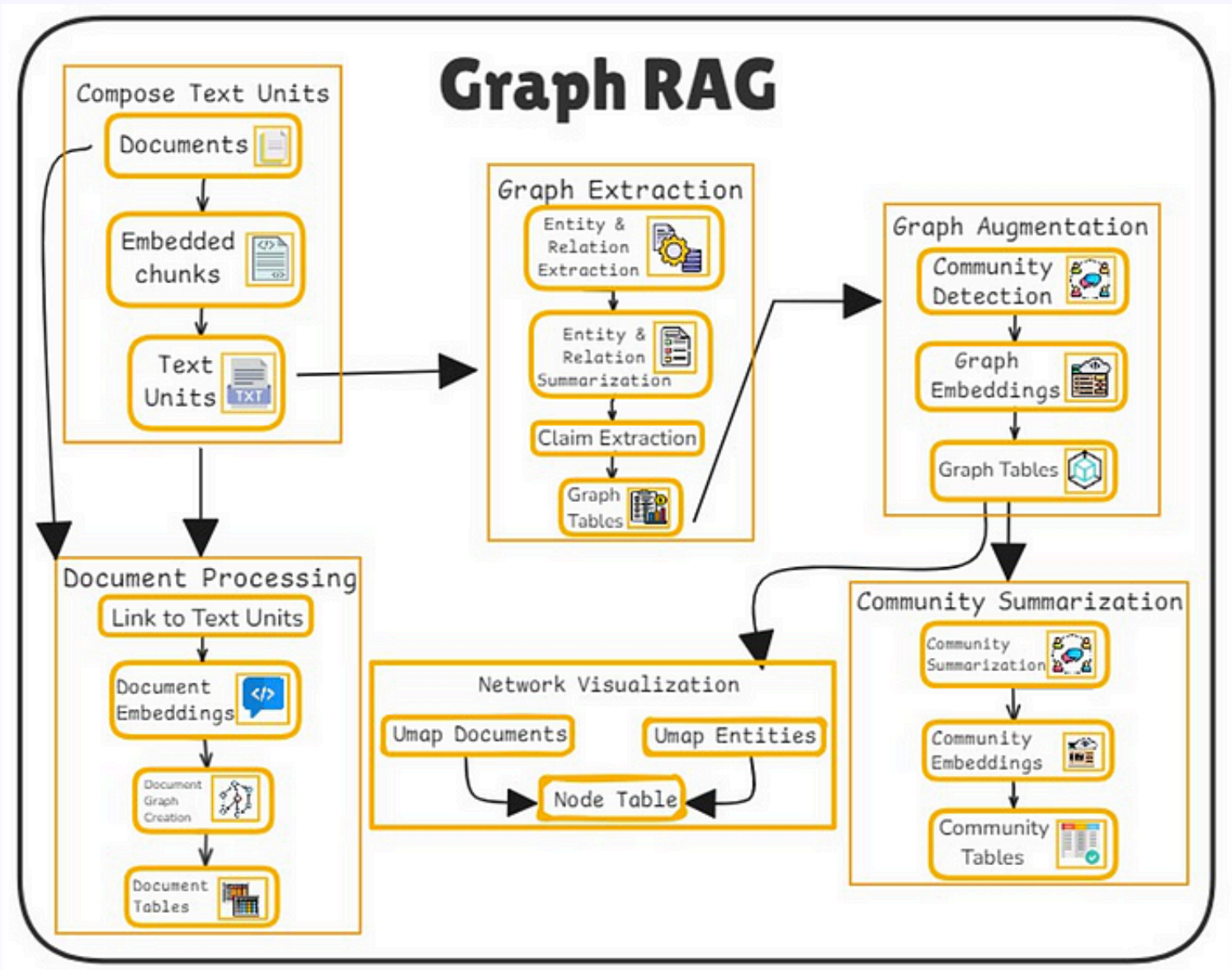
Different Types of RAG Architectures

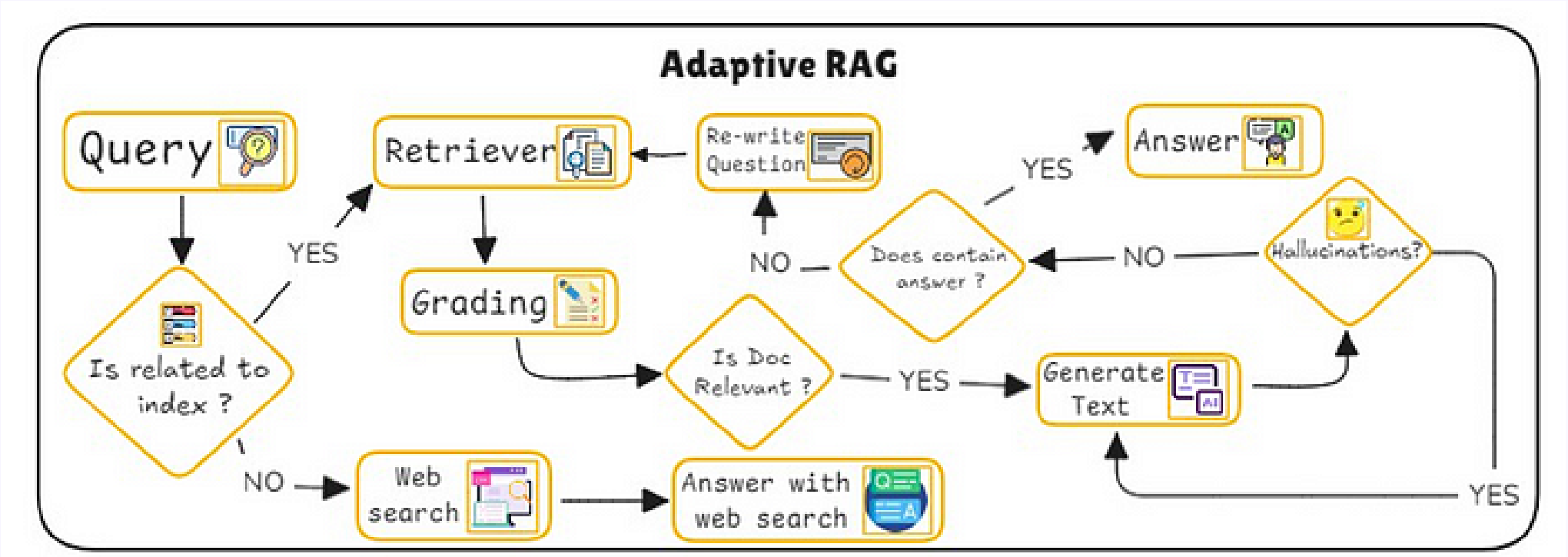


Graph RAG



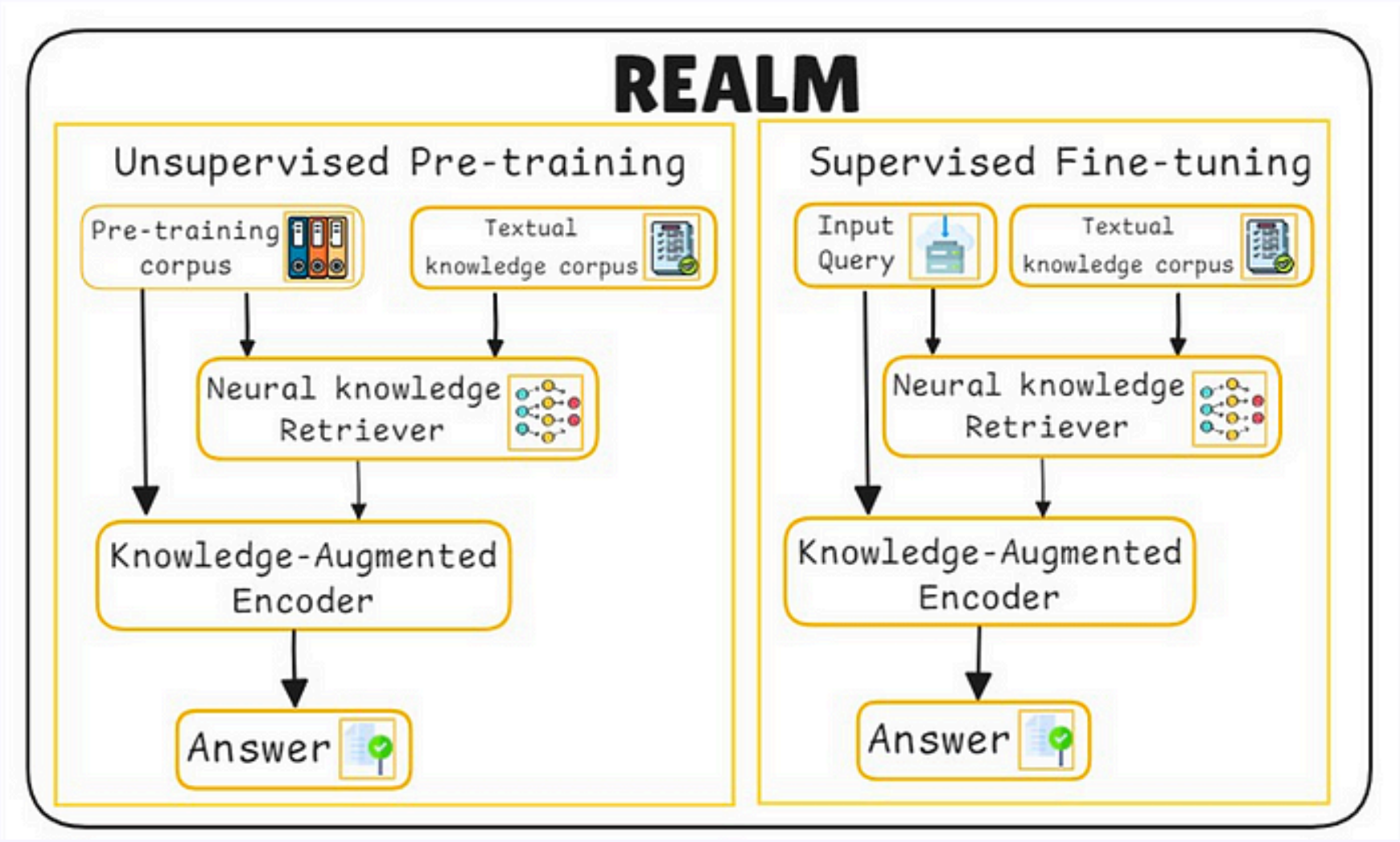
- A variation of Retrieval-Augmented Generation (RAG) where the retrieval process utilizes graph structures.
- Graphs represent entities and relationships, enhancing context-aware retrieval.
- Nodes in the graph can represent concepts, while edges denote relationships or contextual links.
- The graph ensures that retrieved documents or pieces of information are linked logically.
- Leveraging graph traversal enables more coherent reasoning over interrelated entities.

Adaptive RAG



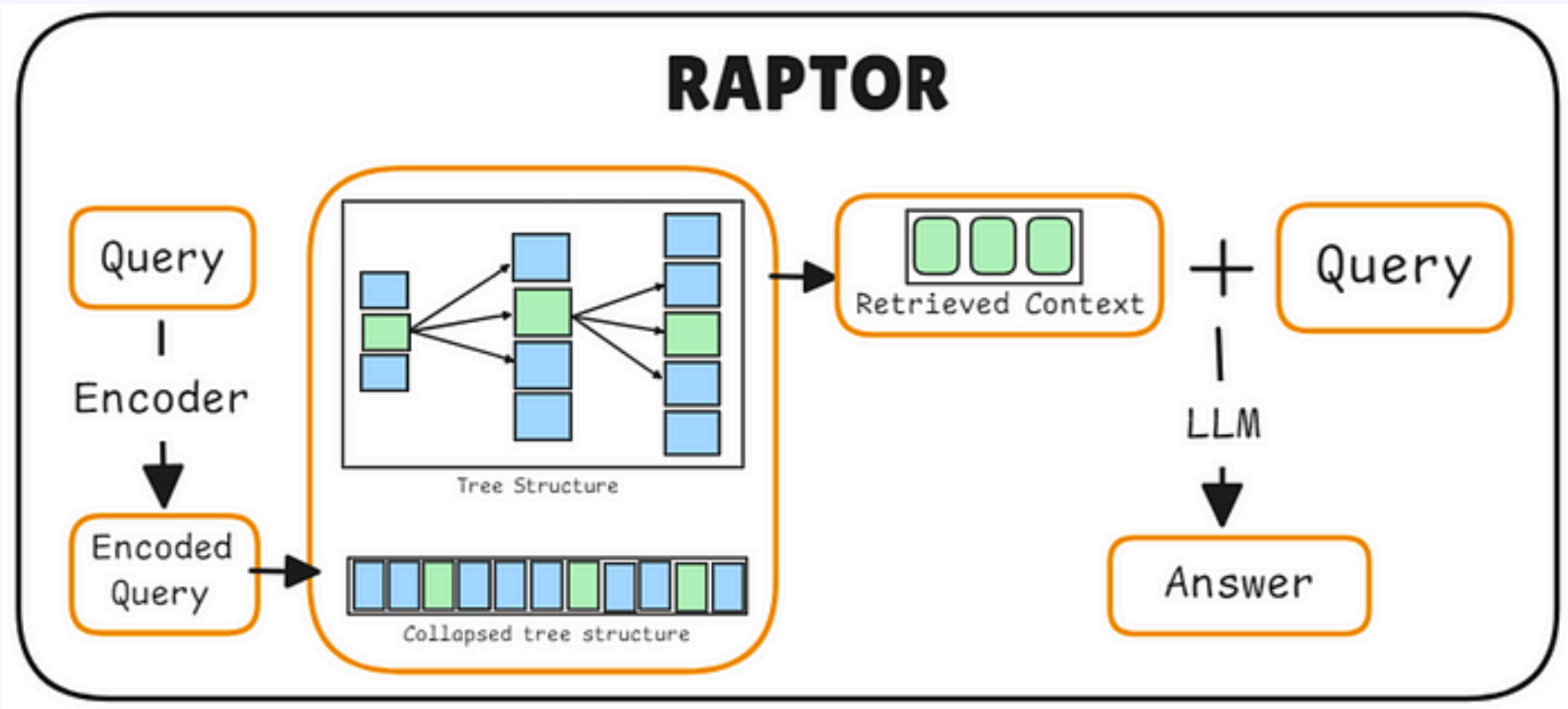
- A RAG method that dynamically adjusts its retrieval and generation strategies based on the input or context.
- The retrieval mechanism evolves based on the query or generated content.
- The generation adapts to the retrieval results and specific context dynamically.
- Greater flexibility and better handling of diverse queries.
- Improved performance on complex or multi-turn tasks.

REALM: Retrieval augmented language model pre-training



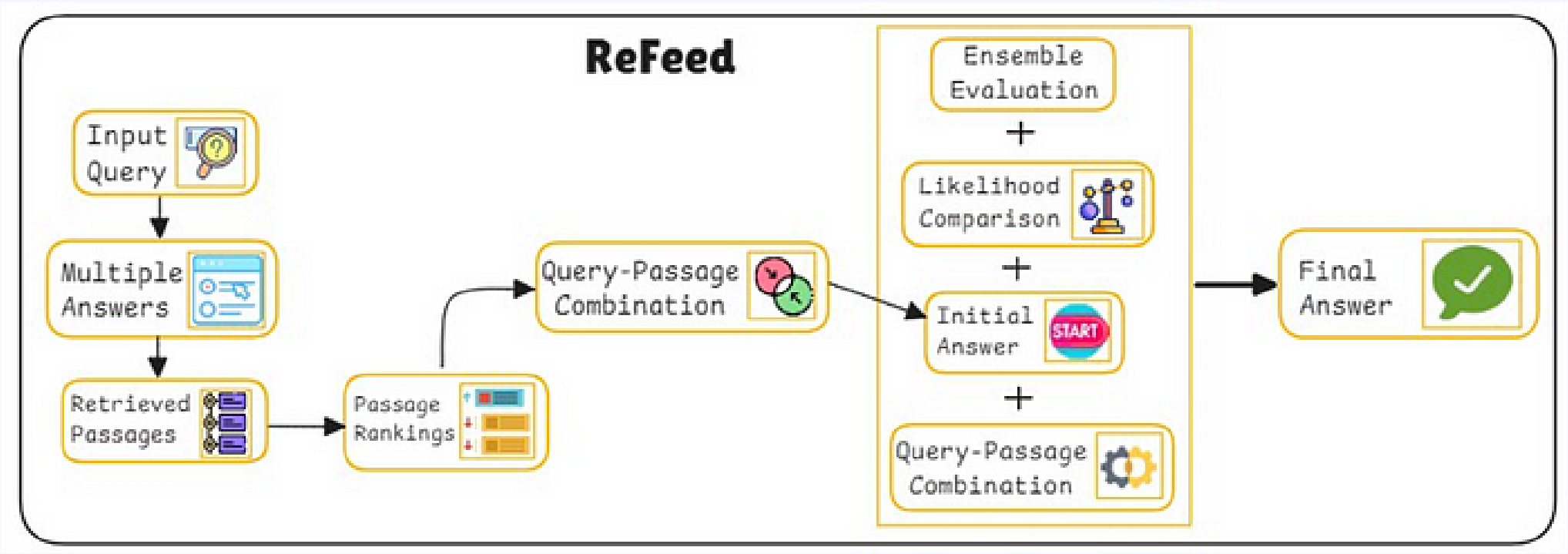
- A pretraining framework where a language model learns to retrieve and incorporate relevant information during training and inference
- Retriever: Learns to fetch relevant passages from a large corpus based on a query.
- Reader/Generator: Processes the retrieved data to produce an answer or text.
- During training, the model jointly optimizes retrieval and generation, learning to align the retrieved content with the task.

RAPTOR: Recursive Abstractive Processing for Tree-Organized Retrieval



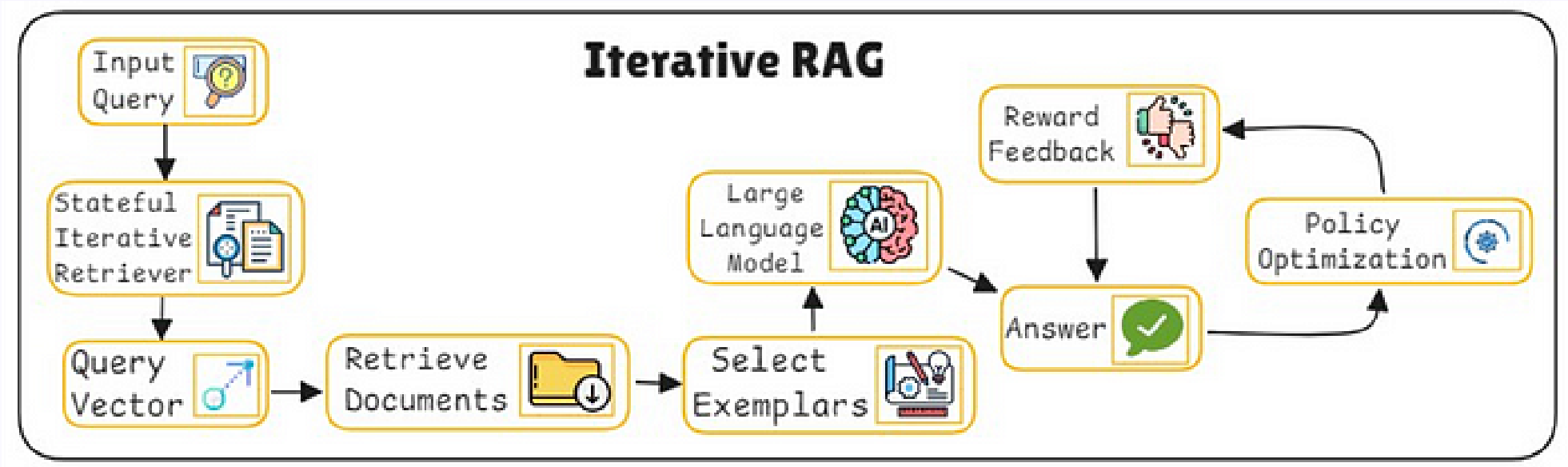
- A retrieval method designed to work with hierarchical or tree-organized data structures.
- Navigates hierarchical data (e.g., tree-like structures) to fetch information.
- Processes retrieved elements to generate a coherent abstractive summary or response.
- Suitable for multi-level document structures such as legal texts, knowledge graphs, or technical documentation.
- Handles hierarchical complexity effectively.
- Produces more concise and relevant summaries by abstracting over the recursive structure.

REFEED: Retrieval Feedback



- A method where the output or intermediate results from generation provide feedback to enhance retrieval.
- The generated text (or a summary of it) is fed back into the retrieval system to refine subsequent searches.
- Works in a feedback loop where generation informs retrieval, and retrieval informs generation.
- Can be viewed as a dynamic improvement cycle.
- Enables retrieval systems to adjust based on contextual clues from the generation phase.
- Better alignment between retrieved content and the task requirements.

Iterative RAG



- A retrieval-augmented generation approach that iterates between retrieval and generation steps.
- Each step refines the query or context, leading to progressively better retrieval results.
- The generation adapts dynamically to the refined retrieval results.
- Refinement helps address ambiguity or incomplete queries.
- Effective for complex and layered information needs.

- Workflow
  - Retrieve initial information based on the input.
  - Use the generated output to reformulate or refine the query.
  - Repeat retrieval and generation until convergence or a stopping criterion is met.

Moreover,
we are offering a

Free Certification

on RAGs, check the link in the description

@Harshit Ahluwalia