

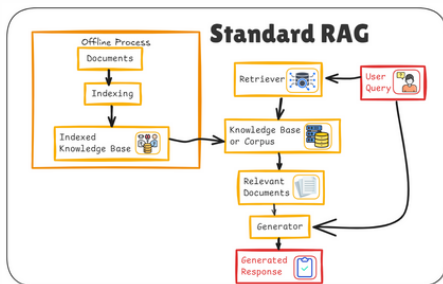
6 Different RAG Techniques

PART 2

@Bhavishya Pandit

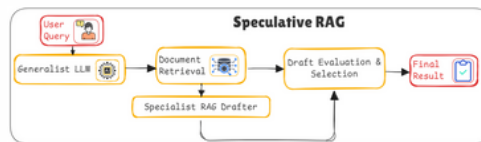
Standard RAG

- Combines **retrieval** with **large language models** for accurate, context-aware responses.
- Breaks **documents into chunks** for efficient information retrieval.
- Aims for **1-2 second response times** for real-time use.
- **Enhances answer quality** by leveraging external data sources.



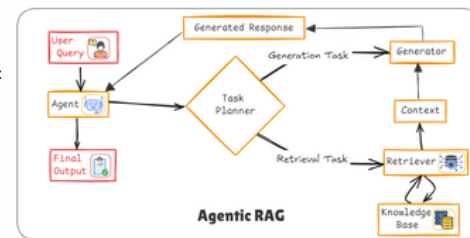
Speculative RAG

- Uses a small **specialist model** for drafting and a larger **generalist model** for verification, ensuring efficiency and accuracy.
- **Parallel Drafting**: Speeds up responses by generating multiple drafts simultaneously.
- **Superior Accuracy**: Outperforms standard RAG systems.
- **Efficient Processing**: Offloads complex tasks to specialized models, reducing computational load.



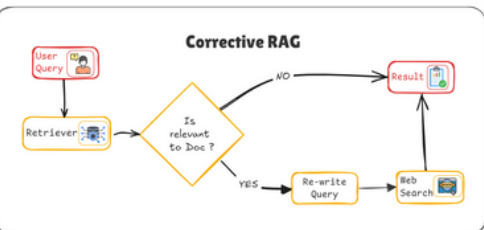
Agentic RAG

- Uses **adaptive agents** for real-time strategy adjustments in information retrieval.
- Accurately **interprets user intent** for relevant, trustworthy responses.
- **Modular design** enables easy integration of new data sources and features.
- Enhances **parallel processing** and **performance** on complex tasks by running agents concurrently.



Corrective RAG

- Focuses on **identifying and fixing errors** in generated responses.
- Uses multiple passes to **improve outputs** based on feedback.
- Aims for **higher precision** and **user satisfaction** compared to standard RAG.
- Leverages user feedback to **enhance the correction** process.

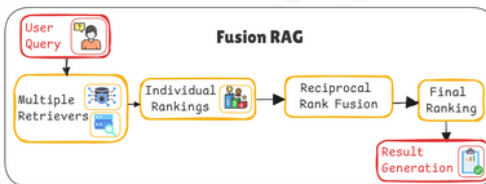


@Bhavishya Pandit



Fusion RAG

- Integrates **multiple retrieval** methods and data sources for enhanced response quality.
- Provides **comprehensive answers** by leveraging diverse data inputs.
- **Increases system resilience** by reducing dependence on a single source.
- Adapts retrieval **strategies dynamically** based on query context.

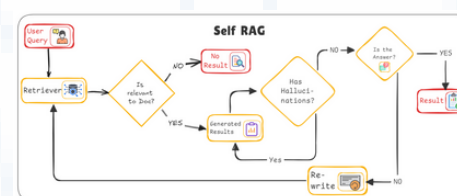


@Bhavishya Pandit



Self RAG

- Uses the model's own outputs as retrieval candidates for **better contextual relevance**.
- Refines responses iteratively, improving **consistency** and **coherence**.
- Grounds responses in prior outputs for **increased accuracy**.
- **Adapts retrieval** strategies based on the conversation's evolving context.



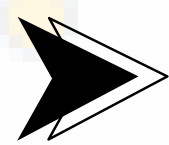
@Bhavishya Pandit



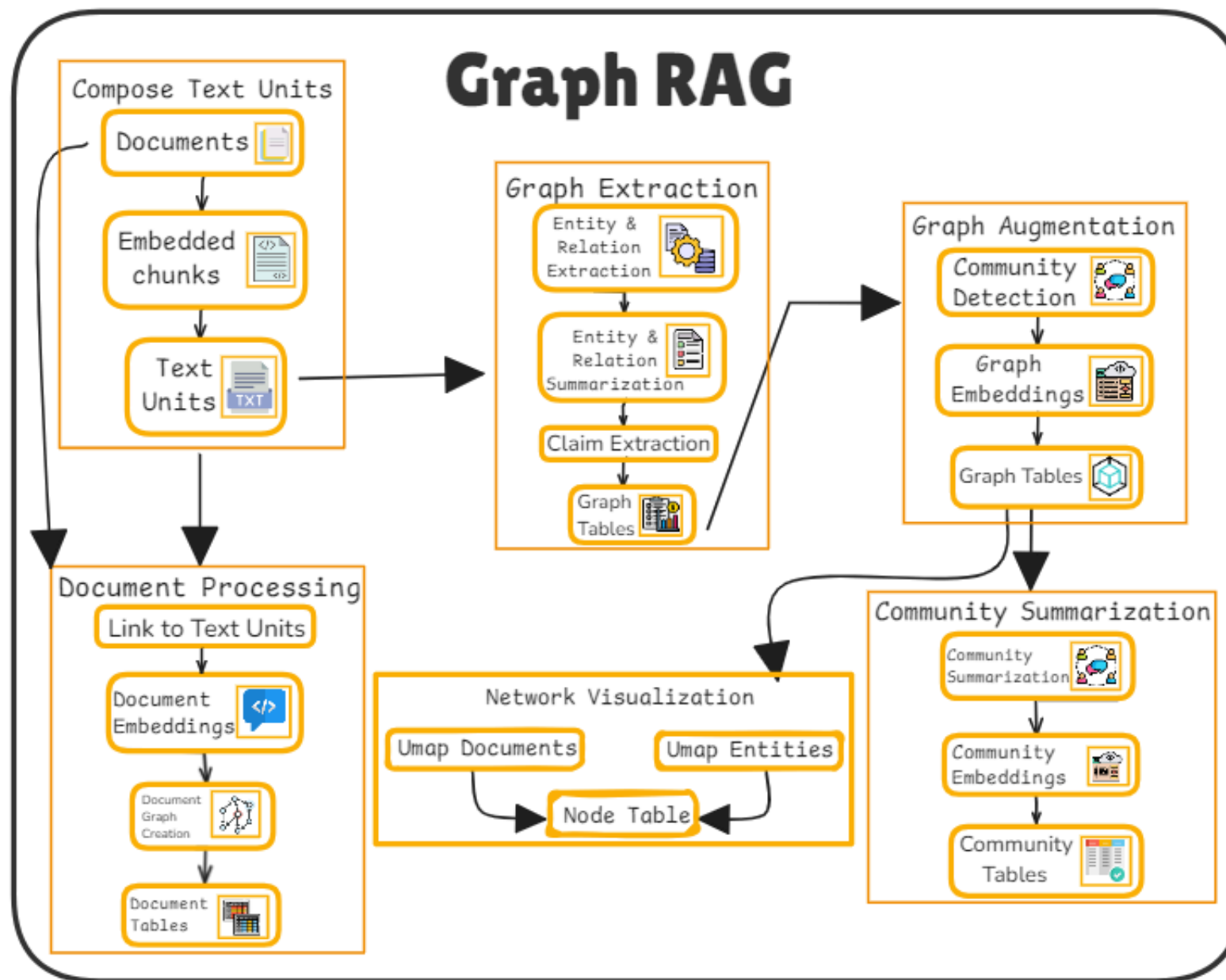
Part 1

In Part 1, we delved into 6 RAG techniques. Now, in Part 2, we expand on that foundation by introducing 6 more innovative RAG techniques.

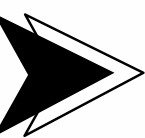
@Bhavishya Pandit



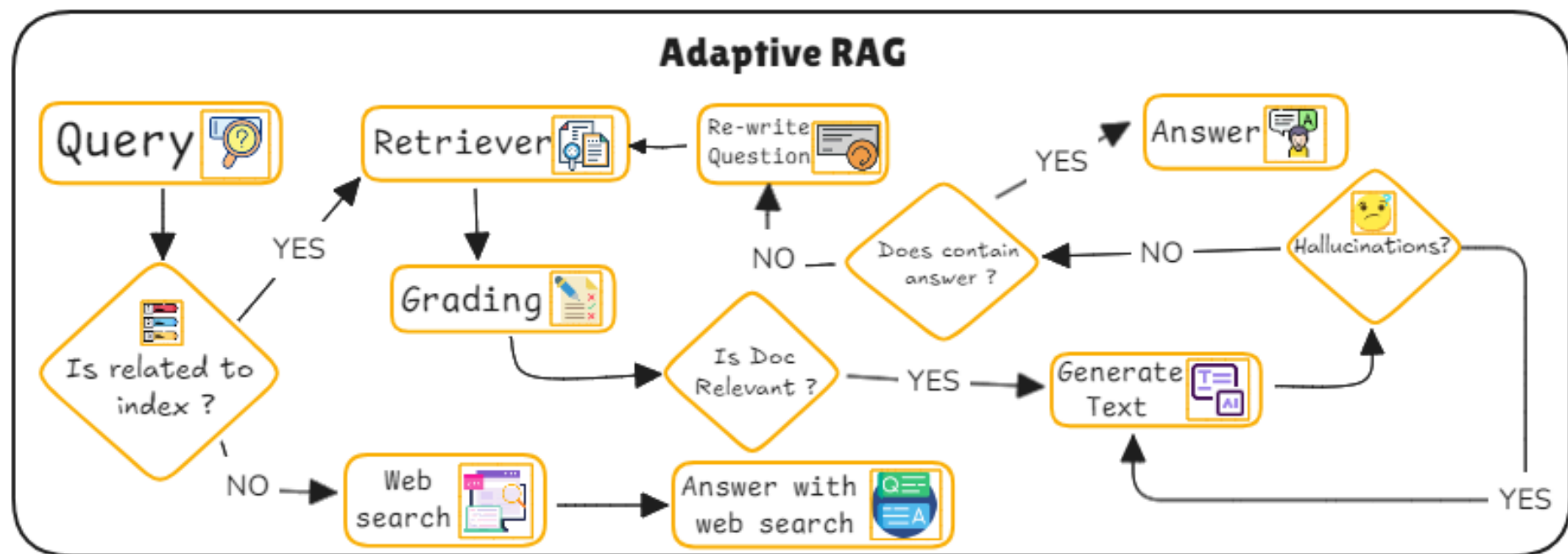
Graph RAG



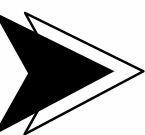
- Graph RAG constructs a **knowledge graph** on-the-fly, linking relevant entities during retrieval.
- It leverages **node relationships** to decide when and how much external knowledge to retrieve.
- **Confidence scores** from the graph guide expansion, avoiding irrelevant additions.
- This approach **improves efficiency** and **response accuracy** by keeping the knowledge graph compact and relevant.



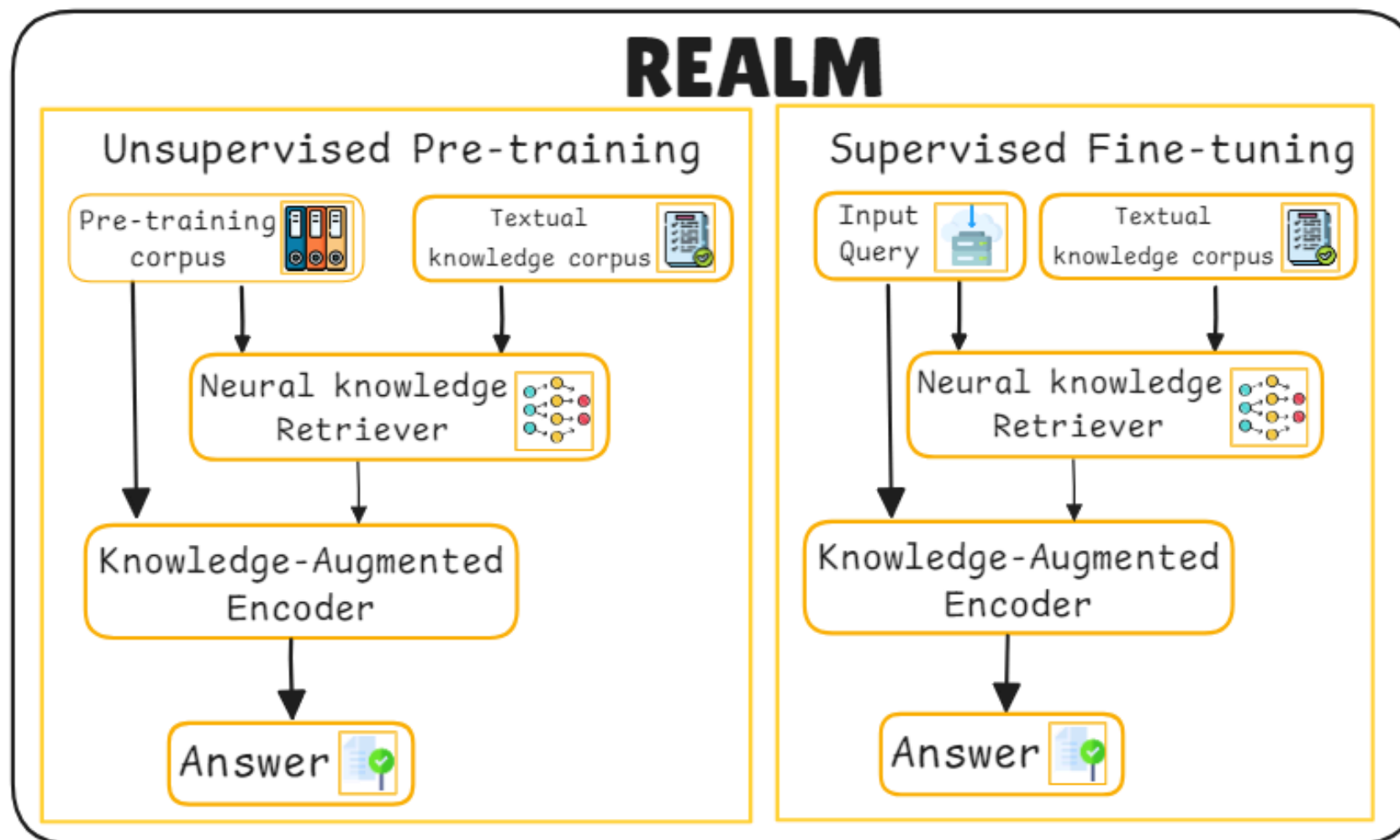
Adaptive RAG



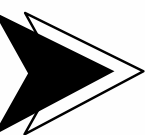
- It **dynamically** decides when to retrieve external knowledge, balancing internal and external knowledge.
- It uses **confidence scores** from the language model's internal states to assess retrieval necessity.
- An honesty probe helps the model **avoid hallucinations** by aligning its output with its actual knowledge.
- It **reduces unnecessary retrievals**, improving both efficiency and response accuracy.



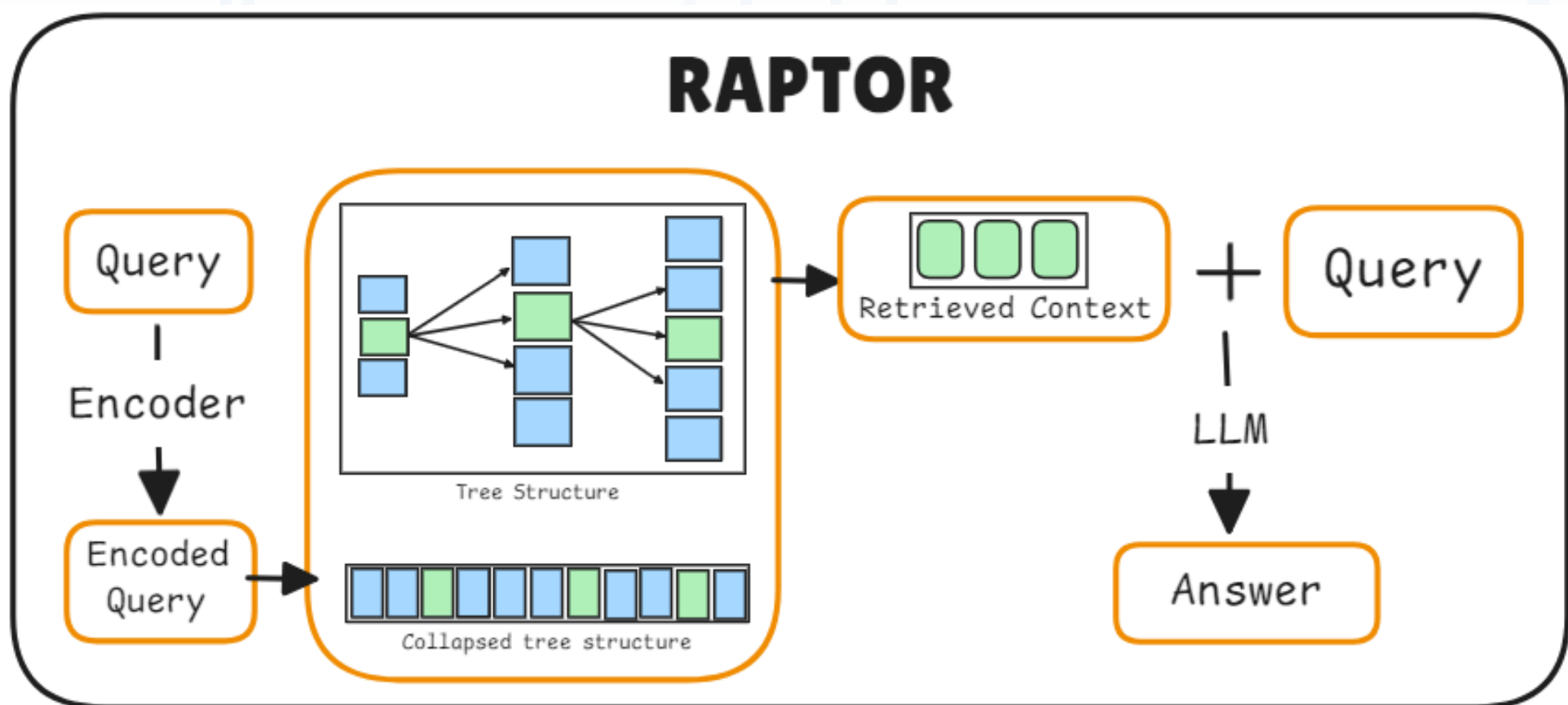
REALM: Retrieval augmented language model pre-training



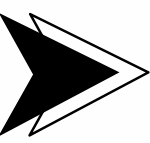
- REALM retrieves relevant documents from large corpora like Wikipedia to **enhance model predictions**.
- The retriever is trained with masked language modeling, optimizing retrieval to **improve prediction accuracy**.
- It uses **Maximum Inner Product Search** to efficiently find relevant documents from millions of candidates during training.
- REALM outperforms previous models in **Open-domain Question Answering** by integrating external knowledge.



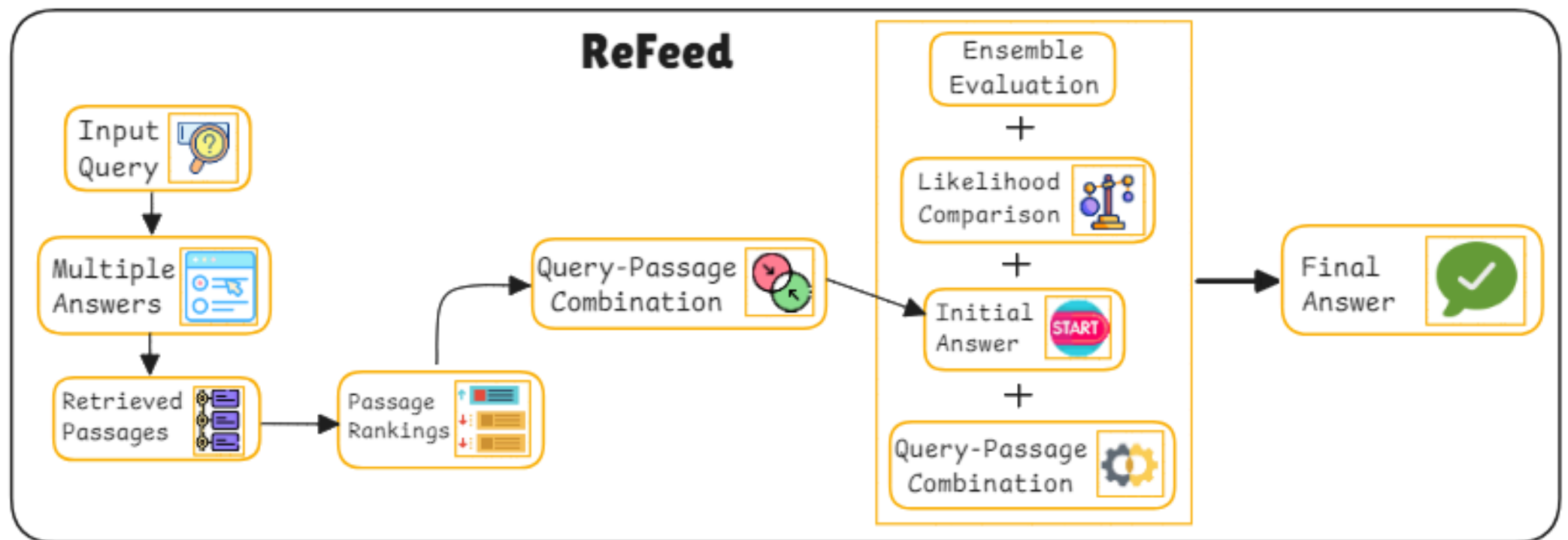
RAPTOR: Recursive Abstractive Processing for Tree-Organized Retrieval



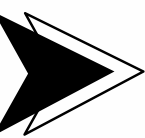
- RAPTOR builds a **hierarchical tree** by **clustering** and **summarizing text recursively**.
- It enables retrieval at **different abstraction levels**, combining **broad themes** with specific details.
- RAPTOR **outperforms traditional methods** in complex question-answering tasks.
- Offers tree traversal and collapsed tree methods for **efficient information retrieval**.



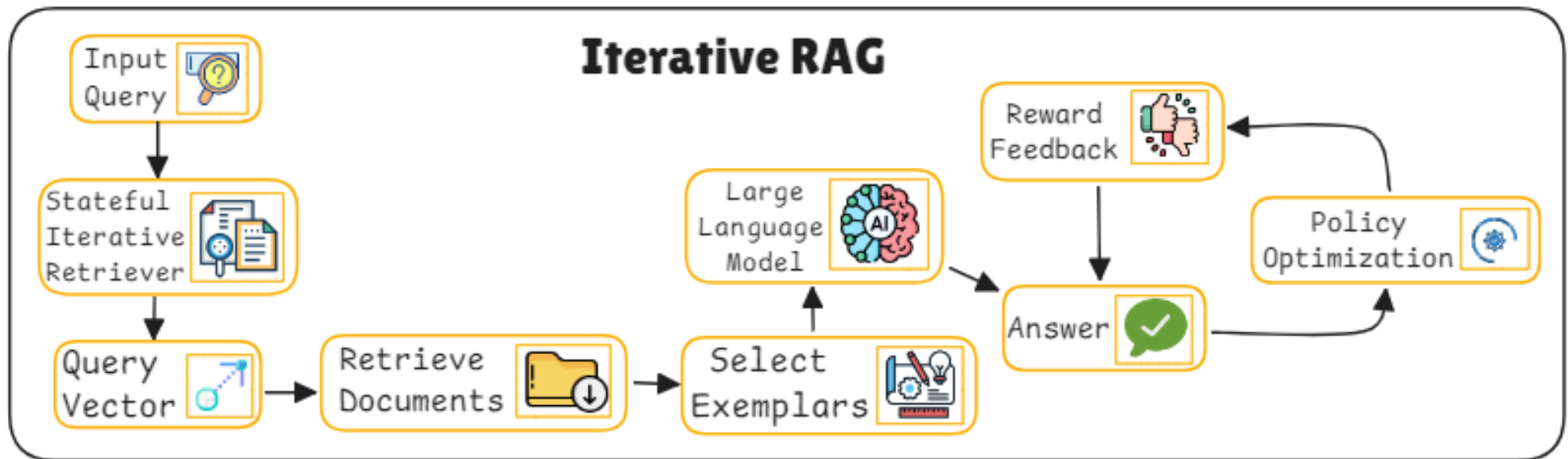
REFEED: Retrieval Feedback



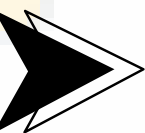
- REFEED refines model outputs using **retrieval feedback without fine-tuning**.
- Initial answers are improved by retrieving relevant documents and adjusting the response based on the new information.
- Generates **multiple answers** to improve retrieval accuracy.
- Combines pre- and post-retrieval outputs using a **ranking system to enhance answer reliability**.



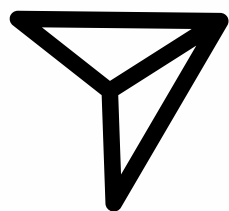
Iterative RAG



- Unlike traditional retrieval, iterative RAG performs **multiple retrieval steps**, refining its search based on feedback from previously selected documents.
- Retrieval decisions follow a **Markov decision process**.
- **Reinforcement learning** improves retrieval performance.
- The iterative retriever **maintains an internal state**, allowing it to adjust future retrieval steps based on the accumulated knowledge from previous iterations.



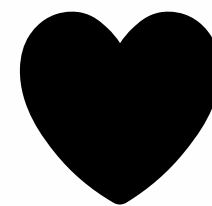
Bhavishya Pandit



Share your
thoughts



Save for
later



Like this
post