Agentic Retrieval-Augmented Generation: A Comparative Survey

Harshit Mandad
Department of Computer Science and Engineering
Manipal University Jaipur
harshit.2427030400@muj.manipal.edu

Dr. Lav Upadhyay
Department of Computer Science and Engineering
Manipal University Jaipur
lav.upadhyay@jaipur.manipal.edu

***Abstract*—Retrieval-Augmented Generation (RAG) has emerged as an effective approach for improving the factual reliability of large language models (LLMs) by integrating external knowledge sources into the generation process. However, most conventional RAG systems rely on static, single-step retrieval, which limits their ability to handle complex queries and evolving information contexts. Agentic Retrieval-Augmented Generation (Agentic RAG) addresses these limitations by introducing capabilities such as iterative retrieval, task decomposition, contextual memory, and autonomous decision-making. These systems are able to dynamically refine queries, adapt retrieval strategies, and structure multi-step reasoning processes more effectively. In this work, we present a comparative study of prominent Agentic RAG paradigms, including goal-oriented, multi-agent, graph-based, and adaptive approaches. The analysis highlights their respective strengths in improving reasoning accuracy and reducing hallucinations, while also identifying practical challenges such as increased computational cost, system complexity, and limited scalability. To address these issues, we propose a reinforcement learning-based Agentic RAG framework that enables adaptive retrieval strategy selection through hierarchical policy learning. The proposed approach aims to improve decision-making, enhance adaptability, and support more reliable knowledge-grounded generation in complex environments. This paper provides a structured comparative analysis and highlights key research challenges in Agentic RAG systems.**

Index Terms : Agentic RAG, Retrieval-Augmented Generation, Large Language Models (LLMs), Reinforcement Learning, Multi-Agent Systems, Adaptive Retrieval, Hallucination Mitigation

I. Introduction

Recent advances in Large Language Models (LLMs) have enabled significant progress in natural language understanding and generation. However, despite their capabilities, these models often generate factually incorrect or unsupported information, particularly when operating beyond their training data. This limitation poses serious challenges in domains where accuracy and reliability are critical. Retrieval-Augmented Generation (RAG) has been proposed as a practical solution to these challenges by combining information retrieval with text generation. By grounding responses in external knowledge sources, RAG improves factual consistency and reduces unsupported outputs. However, traditional RAG systems are inherently limited by their static design. Most approaches rely on a single retrieval step followed by generation, without any mechanism for iterative refinement or reasoning. This makes them less effective for complex queries that require multi-step inference or contextual adaptation. As application demands grow—particularly in domains such as healthcare, finance, and education—there is an increasing need for systems that can not only retrieve relevant information but also plan, adapt, and reason dynamically.

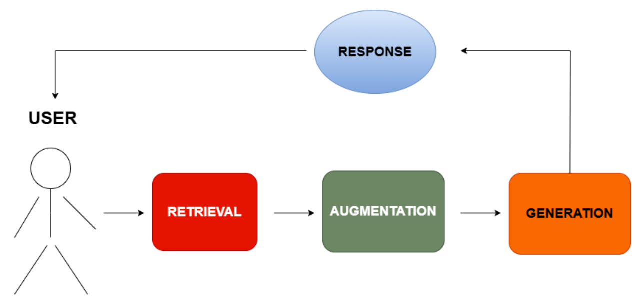

Fig. 1: Basic Components of RAG

Agentic Retrieval-Augmented Generation (Agentic RAG) has recently emerged as a promising extension to address these challenges. By incorporating planning, memory, and tool usage, these systems enable structured task execution and iterative reasoning. Despite these improvements, many existing Agentic RAG frameworks still rely on heuristic or rule-based coordination strategies, which limits their adaptability in dynamic environments. Motivated by these limitations, this work explores the integration of reinforcement learning into Agentic RAG to enable adaptive and data-driven decision-making. In addition, a comparative analysis of existing methodologies is conducted to better understand their strengths, limitations, and trade-offs.

II. Overview of Agentic Retrieval-Augmented Generation

Agentic Retrieval-Augmented Generation (Agentic RAG) extends traditional RAG architectures by introducing elements of planning, adaptability, and autonomous decision-making. Unlike conventional pipelines that follow a fixed retrieve-then-generate structure, Agentic RAG systems incorporate iterative feedback and dynamic control, allowing them to handle more complex and goal-oriented tasks. Although implementations may vary, most Agentic RAG systems consist of three primary components: the retriever, the generator, and the augmentation module, as illustrated in Fig. 1. Together, these components enable the system to generate responses that are both context-aware and grounded in external knowledge.

A. Retriever Module

The retriever plays a central role in identifying relevant information from external knowledge sources based on a given query or intermediate reasoning step. In traditional RAG systems, this process is typically performed once, which limits flexibility. In contrast, Agentic RAG allows for iterative and context-aware retrieval. Queries can be refined over multiple steps, enabling the system to gather more relevant and targeted information. This is particularly important for multi-hop reasoning tasks, where a single retrieval step is often insufficient. The quality of retrieval directly impacts the overall system performance. Irrelevant or incomplete retrieval can propagate errors into later stages, increasing the likelihood of hallucination.

B. Generator Module

The generator is typically implemented using a large language model (LLM) that synthesizes responses by combining the input query with retrieved contextual information. In Agentic RAG, the generator extends beyond producing final outputs and actively participates in intermediate reasoning processes. It can generate sub-queries, perform task decomposition, and produce step-by-step reasoning under the guidance of planning mechanisms. This enhances coherence, interpretability, and reasoning accuracy. By leveraging both retrieved knowledge and parametric understanding, the generator contributes to more accurate and contextually relevant outputs.

C. Augmentation Module (Knowledge Base)

The augmentation module consists of external knowledge sources that support the retrieval process. These sources may include unstructured document corpora, structured databases, knowledge graphs, domain-specific repositories, or hybrid storage systems. The knowledge base provides factual grounding and domain alignment, enabling the system to access up-to-date and specialized information beyond the static knowledge encoded within the LLM. Efficient indexing, semantic search capabilities, and scalable storage are essential for ensuring effective retrieval. The diversity and quality of the knowledge base directly influence the overall performance and reliability of the Agentic RAG system. The overall workflow of the Agentic RAG pipeline, including iterative retrieval and reasoning, is illustrated in Fig. 2.

III. Contributions

This work makes the following key contributions:

- A structured comparative analysis of existing Agentic RAG methodologies, identifying key strengths, limitations, and research gaps.

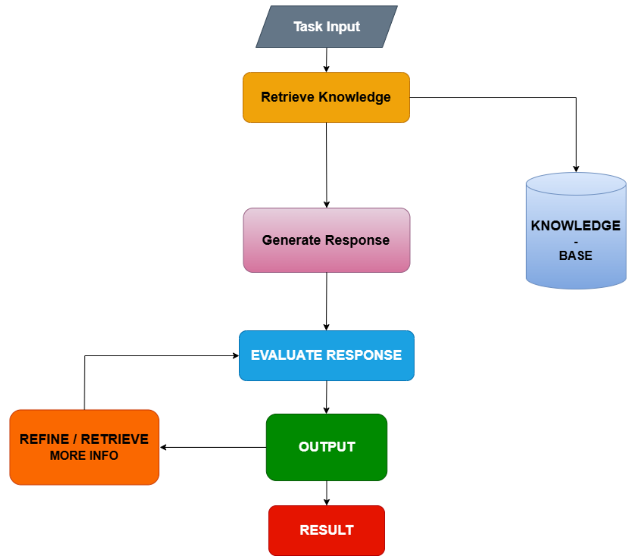

Fig. 2: Work Flow of Agentic-RAG

- A detailed evaluation of architectural trade-offs across multiple paradigms, including graph-based, multi-agent, and adaptive systems.
- The proposal of a Reinforcement Learning-enhanced Agentic RAG framework for adaptive retrieval and reasoning optimization.
- Identification of open challenges and future research directions for scalable and autonomous retrieval systems.

A. Comparative Analysis of Agentic RAG Methodologies

This section presents a comparative evaluation of major Agentic RAG methodologies, with a focus on their design choices, strengths, and limitations. A summary of these approaches is provided in Table **??**. Rather than a single unified framework, Agentic RAG has evolved into multiple paradigms, each addressing specific limitations of traditional systems. Table **??** further highlights differences in retrieval strategies, reasoning capabilities, and system trade-offs.

B. Traditional Agentic RAG

Traditional retrieval-based RAG systems emphasize simplicity, low latency, and cost efficiency. Their single-pass retrieval mechanism enables fast response generation and ease of implementation. However, these systems lack iterative reasoning, dynamic task decomposition, and adaptive retrieval strategies. Consequently, they exhibit limited hallucination control and are unsuitable for complex, multi-hop reasoning tasks or ambiguous query environments.

C. Graph-Based Agentic RAG

Graph-based RAG introduces structured reasoning through graph representations and traversal strategies. By modeling relationships between entities, these systems enable explainable reasoning paths and improved factual grounding. Despite these advantages, graph-based approaches suffer from high construction and maintenance costs, scalability challenges, and strong dependency on graph quality. Additionally, adapting

Sr. No. (CITATIONS)	Methodology	Datasets	Advantages	Disadvantages
[Zhang, Y., et al. (2025).arXiv.]	Multi-Agent Agentic RAG	Hotpot QA Multi-hop benchmark Multi- domain QA datasets	- High Accuracy - Parallel Execution - Strong Performance	- High system complexity - Harder debugging - Expensive
[Kumar, S., et al. (2025). arXiv.]	Goal-Oriented Agentic RAG	FinTech enterprise knowledge base(1,624 docs) Enterprise documentation, Human-curated procedural QA	- Strong multi-step procedural reasoning - Better handling ambigous and complex queries - Reduced Hallucination	- High latency - High computational and infrastructure cost - High System complexity
[Kumar, A., et al. (2025). IEEE Access.]	Traditional Agentic RAG	Natural Questions (NQ) Wikipedia-based QA datasets Enterprise document repositories Open-domain QA benchmarks	- Low latency - Fast response rate - Simple architecture and debugging - Cost efficient	- No multi-step reasoning - Limited hallucination control
[Zhang, Y., et al. (2025). . arXiv.]	Graph-Based Agentic RAG	Knowledge-graph QA datasets Biomedical knowledge graphs Healthcare knowledge graphs Relational multi-hop QA datasets	- Graph traversal for structured reasoning - Graph paths provide reasoning traces - Reduced hallucination	- High graph construction maintenance cost - Scalability issues - Limited flexibility
[Zhang, Y., et al. (2025). arXiv.]	Hierarchical RAG	Enterprise knowledge bases FinTech corpora Smart city IoT datasets OpenAI Gym (simulation tasks)	- Efficient handling of complex tasks - Improved Coordination - Better resource allocation	- Increased latency - Error propagation - Extreme complex system design
[Li, X., et al. (2025). arXiv.]	Capability-Centric Agentic RAG (Conceptual)	Tool-use benchmarks QA corpora (general references)	- Clear conceptual framework - Holistic Intelligence Modelling	- Conceptual framwork - Difficult capability measurement - Implementation ambiguity
[Zhang, Y., et al. (2025). . arXiv.]	Adaptive Agentic RAG	PubMed corpus arXiv corpus IEEE Xplore documents Google Scholar documents 500-user interaction dataset Query logs & feedback data	- Continous Performance Improvement - Better Personalization	- Increased Computational cost - Longer Response time - May adapt to frequent patterns too close
[Zhang, Y., et al. (2025). . arXiv.]	Corrective Agentic RAG	HotpotQA Scientific QA corpora Open-domain factual QA datasets Enterprise QA collections	- Reduced Hallucination - Improved Output Reliability - Better handling of retrieval failure	- Increased latency - High cost - Self-correction not always reliable
[(2025). Domain-specific . SSRN.]	Domain-Specific Agentic RAG	Clinical guidelines Legal case law repositories Financial filings Regulatory databases FinTech enterprise document	- High domian accuracy - Uses domain curated knowledge bases - Reduced hallucination	- Limited generalization - High data curation cost - Integration complexity
[(2025). Domain-specific SSRN.]	Unified-domain Agent Stack	Domain-specific LLM training (BioBERT, LegalBERT, FinBERT) Regulatory databases Clinical and financial repositories Enterprise knowledge bases	- End-to-End integrated intelligence - High reliability for High-stackes domain	- Extreme high system complexity - High implementaition cost

TABLE I: Comparative Analysis of Agentic-RAG Methodologies

to rapidly evolving knowledge domains remains a significant limitation.

D. Multi-Agent Agentic RAG

Multi-agent architectures decompose tasks across multiple specialized agents, enabling parallel execution and improved performance in complex reasoning scenarios such as multi-hop question answering. However, these systems introduce significant challenges, including increased system complexity, high computational cost, and coordination overhead. Debugging becomes difficult due to inter-agent dependencies, and conflicts between agent outputs may lead to inconsistent reasoning. Furthermore, scalability degrades as the number of agents increases, making large-scale deployment challenging.

E. Goal-Oriented Agentic RAG

Goal-oriented frameworks focus on structured task decomposition and procedural reasoning. By aligning retrieval with explicit objectives, these systems improve multi-step reasoning, reduce hallucination, and handle complex queries more effectively. Nevertheless, they incur higher latency and infrastructure costs due to iterative planning and execution. Additionally, the absence of standardized evaluation metrics and risks associated with tool misuse limit their practical deployment in large-scale systems.

F. Hierarchical Agentic RAG

Hierarchical architectures introduce layered control mechanisms to manage complex workflows. These systems enhance coordination and structured reasoning by organizing tasks across multiple levels of abstraction. However, hierarchical designs increase latency due to sequential processing and are prone to error propagation across layers. The added architectural complexity and computational overhead further limit scalability and real-time applicability.

G. Adaptive Agentic RAG

Adaptive Agentic RAG systems incorporate feedback-driven learning mechanisms to dynamically refine retrieval strategies and improve performance over time. These systems demonstrate strong adaptability and personalization capabilities. Despite these benefits, adaptive approaches introduce

Methodology	Limitations/Challenges
Multi-Agent Agentic RAG	- Multi-Agent Coordination and Communication - Response Conflict across agents
Goal-Oriented Agentic RAG	- Scalability issue - No standard evaluation metric system - Risk of API error/ tool misuse
Traditional Agentic RAG	- Poor performance for comple tasks - No adaptability/ learning - Limited knowledge base
Graph-Based Agentic RAG	- Dependency on graph quality - Intergration Complexity - Struggle with newly-emerging info
Hierarchical RAG	- More layers=Scalability issues - Lack of standard evaluation metrics - High computational cost
Capability-Centric Agentic RAG (Conceptual)	- Lack of standardized benchmark - High integration complexity
Adaptive Agentic RAG	- Data dependency for learning - Stability and drift issues - Evaluation Complexity
Corrective Agentic RAG	- Over-Correction risk - Evaluation Benchmark gaps - No universal way to measure answer correctness
Domain-Specific Agentic RAG	- Data availability constraint - Knowledge drift
Unified-domain Agent Stack	- Scalability issues - Lack of standardized evaluation framework

TABLE II: Identified Limitations Across Agentic-RAG Methodologies

higher computational overhead, longer response times, and dependency on high-quality feedback data. Additionally, they face challenges in evaluation due to dynamic system behavior and risk overfitting to repetitive query patterns, potentially reducing generalization.

H. Domain-Specific and Unified Agentic Architectures

Domain-specific Agentic RAG systems leverage curated knowledge bases to achieve high accuracy and reduced hallucination in specialized domains such as healthcare, legal, and finance. Unified-domain architectures extend this concept by integrating domain-adapted LLMs with retrieval and agentic components for end-to-end reliability. However, these systems face limitations in generalization beyond their target domain and require significant effort in data curation and maintenance. Knowledge drift, high system complexity, and lack of standardized benchmarking frameworks further complicate their scalability and evaluation.

I. Identified Research Gaps

Based on the comparative analysis, several key research gaps become evident, as summarized in Table III. While existing approaches improve reasoning and retrieval capabilities, they often do so at the cost of increased complexity and computational overhead. In addition, there is a lack of standardized evaluation frameworks, making it difficult to compare different systems objectively. These challenges highlight the need for more adaptive and scalable solutions, motivating the exploration of learning-based approaches.

Limitations Identified
Scalability Optimization : Agentic architectures significantly increase computational complexity, limiting deployment in large-scale systems.
Learning-Driven Retrieval Policy : While adaptive and RL-based approaches show promise, stable and efficient policy learning remains underexplored.
Lack of Standardized evaluation Framework : There is no unified benchmarking methodology for measuring agentic autonomy, goal completion, and hallucination control across architectures.

TABLE III: Critical Identified Research Gaps

By synthesizing these observations, this study establishes a structured evaluation perspective on Agentic RAG methodologies. These insights motivate the need for adaptive, learning-driven coordination mechanisms. The subsequent section explores the integration of Reinforcement Learning within Agentic RAG frameworks, demonstrating how multi-level policy optimization can address existing limitations and enable scalable, efficient, and autonomous retrieval systems.

J. Reinforcement Learning-Enhanced Agentic RAG Framework

To address the limitations identified in existing Agentic RAG methodologies, this paper proposes a Reinforcement Learning (RL)-enhanced Agentic RAG framework. The proposed approach introduces learning-driven optimization into the retrieval and reasoning pipeline, enabling dynamic adaptation, improved decision-making, and efficient multi-step task execution. Unlike existing approaches that rely on predefined rules or heuristics, the proposed framework introduces a learning-based mechanism for optimizing both planning and retrieval. The overall architecture of the system is shown in Fig. 3. By leveraging hierarchical reinforcement learning, the framework separates high-level planning from low-level execution. This allows the system to maintain a balance between long-term strategy and immediate decision-making. The interaction between these components enables more flexible and adaptive behavior, particularly in complex reasoning scenarios.

K. Working Mechanism

The proposed framework operates through a hierarchical policy structure consisting of two primary components:

1) High-Level Policy (Goal Structuring Layer): The high-level policy is responsible for decomposing complex user queries into structured sub-goals and determining the sequence of reasoning steps required to accomplish the task. It performs goal-oriented planning by generating a sequence of intermediate objectives that guide the retrieval and generation process. This policy is trained using long-term reward signals such as answer correctness, reasoning coherence, and hallucination penalties. By optimizing over extended decision horizons, the high-level policy learns effective task decomposition strategies that improve multi-step reasoning and procedural execution.

2) Low-Level Policy (Adaptive Retrieval Control): The low-level policy governs fine-grained retrieval actions, including query reformulation, document selection, tool invocation, and iterative evidence refinement. It dynamically adapts retrieval strategies based on intermediate feedback and contextual signals. This component enables the system to perform context-aware retrieval, ensuring that relevant and high-quality information is selected at each step of the reasoning process. The interaction between high-level planning and low-level execution allows the system to maintain both strategic direction and operational flexibility. The overall architecture of the proposed RL-enhanced Agentic RAG framework is illustrated in Fig. 3.

3) Policy Update and Reward Mechanism: Both policies are updated iteratively using reinforcement learning techniques. The environment provides reward feedback based on multiple evaluation criteria, including:

- Factual accuracy of generated responses
- Relevance and quality of retrieved documents
- Reasoning coherence and consistency
- Retrieval efficiency and latency
- Resource utilization and computational cost

These reward signals guide the learning process, enabling the system to maximize task success while minimizing hallucination and unnecessary retrieval operations. Over time, the framework learns optimal retrieval and reasoning behaviors tailored to different query types and domains. The reinforcement learning workflow, including interaction with the environment and reward feedback, is further illustrated in Fig. 4.

L. Potential Impact and Advantages

The integration of reinforcement learning into Agentic RAG introduces several key advantages:

1) Improved Retrieval and Reasoning Accuracy: The use of feedback-driven optimization enables the system to learn which retrieval strategies yield the most relevant and accurate information. This reduces hallucination and enhances confidence in multi-step reasoning tasks.

Example: In scientific research applications, the system can prioritize retrieval from high-quality, peer-reviewed sources, improving reliability and factual grounding.

2) Adaptive Multi-Step Task Execution: The hierarchical structure enables effective decomposition of complex queries into manageable sub-tasks. Each step is dynamically refined based on intermediate outcomes, allowing robust handling of multi-hop reasoning scenarios.

Example: In healthcare applications, queries can be decomposed into symptom analysis, literature retrieval, and treatment recommendation stages.

3) Scalability and Efficiency Optimization: Reinforcement learning allows the system to identify high-yield retrieval paths, reducing unnecessary computations and improving efficiency. This leads to faster decision-making and better resource utilization compared to static rule-based systems.

4) Enhanced Autonomy and Continuous Learning: The framework supports continuous self-improvement through feedback loops. It can adapt to evolving data distributions, domain changes, and user behavior without requiring manual reconfiguration, resulting in a more autonomous and resilient system.

M. Challenges and Limitations

Despite its advantages, the proposed RL-enhanced Agentic RAG framework introduces several challenges:

1) High Training Complexity: Training hierarchical reinforcement learning models requires large amounts of interaction data and careful reward design. Poorly defined reward functions may lead to suboptimal or unstable learning behavior.

2) Computational Overhead: The integration of RL with multi-step retrieval and reasoning increases computational cost during both training and inference. This may limit real-time applicability in resource-constrained environments.

3) Exploration vs. Exploitation Trade-off: Balancing exploration of new retrieval strategies with exploitation of known optimal behaviors remains a critical challenge. Excessive exploration may degrade performance, while insufficient exploration may limit adaptability.

4) Evaluation Complexity: Evaluating RL-based Agentic RAG systems is inherently difficult due to dynamic behavior and long-term dependencies. Standard benchmarking metrics may not adequately capture policy effectiveness and learning progress.

5) System Stability and Convergence: Ensuring stable convergence of hierarchical policies is non-trivial. Interactions between high-level and low-level policies may introduce instability, particularly in complex, multi-domain environments.

6) Data Dependency and Reward Bias: The performance of the system depends heavily on the quality of feedback signals. Biased or noisy reward signals may lead to incorrect learning patterns and reduced generalization.

N. Discussion

The proposed RL-enhanced Agentic RAG framework directly addresses key limitations identified in existing methodologies, particularly the lack of adaptive optimization and scalability constraints. By introducing learning-driven coordination, the system achieves a balance between reasoning accuracy, adaptability, and computational efficiency. Although

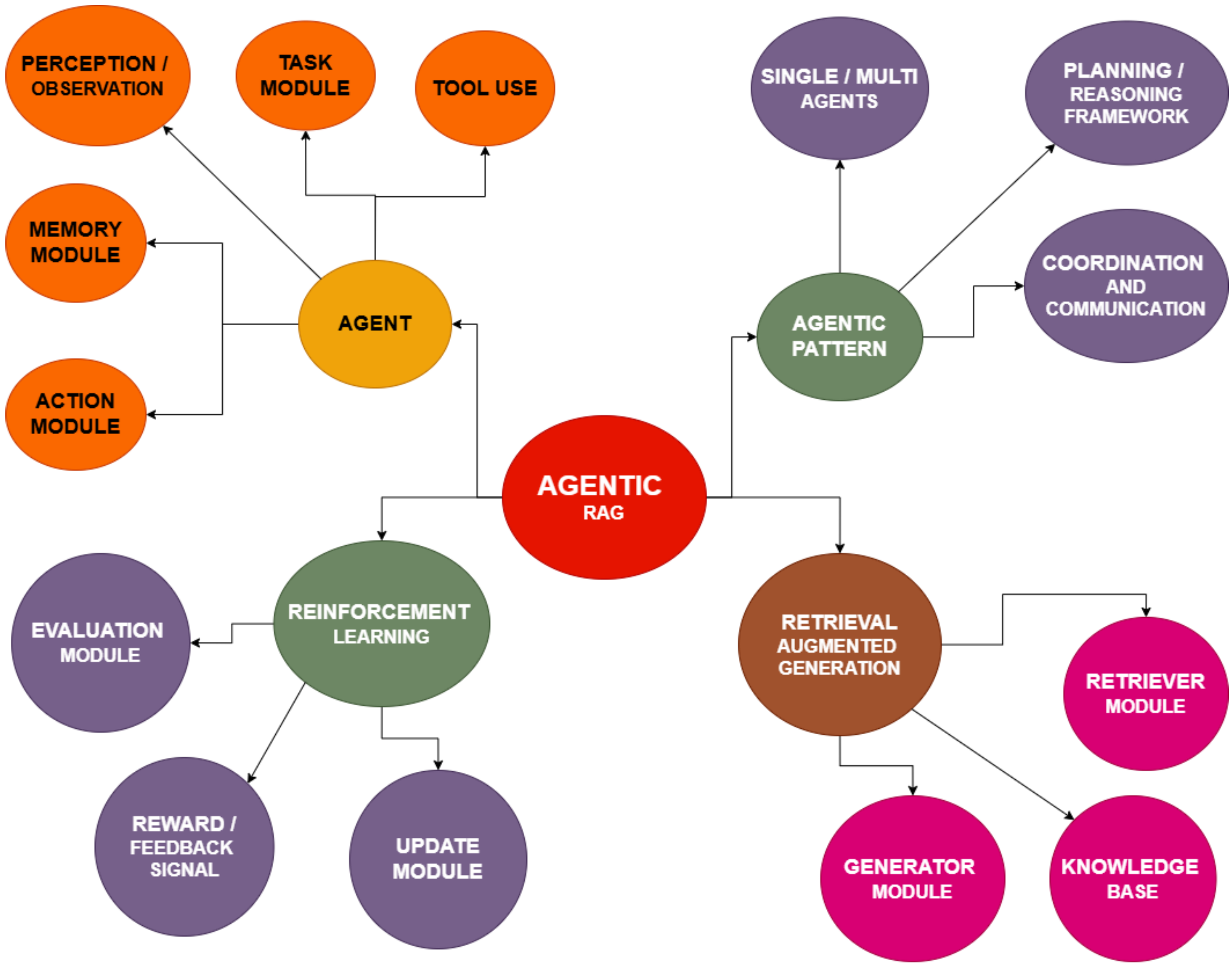

Fig. 3: Agentic-RAG + Reinforcement Learning

challenges remain in terms of training complexity and evaluation, the integration of reinforcement learning represents a promising direction for the development of next-generation intelligent retrieval systems. Future research can focus on improving reward design, reducing computational overhead, and establishing standardized evaluation frameworks for RL-based Agentic RAG architectures.

IV. Conclusion and Future Directions

This work presented a comparative analysis of Agentic Retrieval-Augmented Generation methodologies and examined the role of reinforcement learning in addressing their limitations. While Agentic RAG significantly improves reasoning and factual grounding, many existing approaches remain limited by static or heuristic-driven strategies. To overcome these constraints, a reinforcement learning-based framework was proposed, enabling adaptive retrieval and structured decision-making. Although promising, the proposed approach introduces new challenges, including increased computational requirements and the complexity of training and evaluation. Addressing these issues will be critical for practical implementation. Future work should focus on improving efficiency, developing better reward mechanisms, and establishing standardized evaluation benchmarks. Enhancing explainability and incorporating human feedback are also important directions for making these systems more reliable and widely applicable. In general, the integration of reinforcement learning into Agentic RAG represents a significant step towards more intelligent and adaptive knowledge-driven systems.

Acknowledgment

The author would like to express sincere gratitude to Mr. Lav Upadhyay for his invaluable guidance, insightful feedback, and continuous encouragement throughout the course of this research. His expertise and mentorship played a crucial role in shaping the direction, methodology, and overall quality of this work. The author also acknowledges Manipal University Jaipur for providing the academic environment, institutional

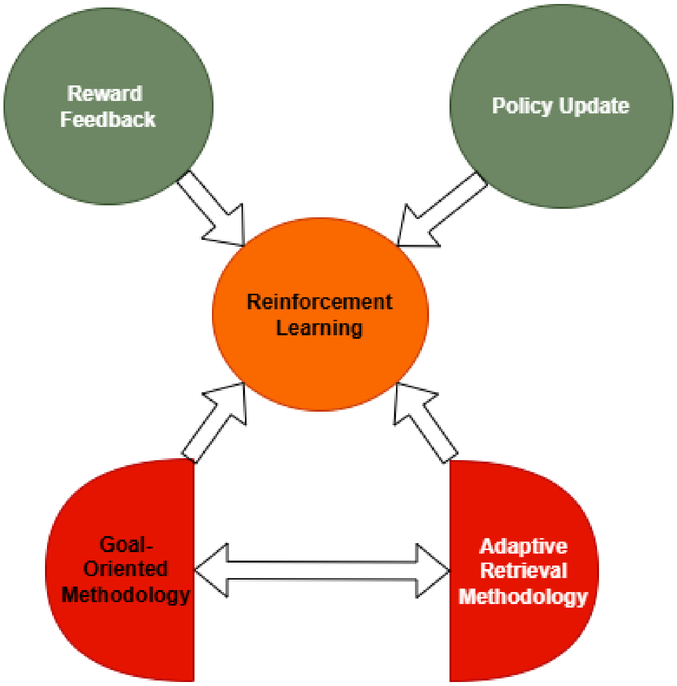

Fig. 4: Reinforcement Learning Framework

support, and resources necessary to carry out this study effectively. Additionally, appreciation is extended to peers and colleagues for their constructive discussions and support, which contributed to refining the ideas and improving the clarity of this research.

References

[1] Y. Zhang *et al.*, "Agentic retrieval-augmented generation: A survey on agentic RAG," arXiv, 2025.
[2] S. Kumar *et al.*, "Retrieval-augmented generation (RAG) for fintech: Agentic design and evaluation," arXiv, 2025.
[3] A. Kumar *et al.*, "Agentic AI: Autonomous intelligence for complex goals—A comprehensive survey," *IEEE Access*, 2025.
[4] X. Li *et al.*, "From generative AI to agentic AI: Foundations, capabilities, and future directions," arXiv, 2025.
[5] "Domain-specific LLMs, retrieval-augmented generation and agentic AI: A unified architecture for specialized intelligence," SSRN, 2025.
[6] A. Singh, "Adaptive agentic RAG systems for improving adaptability and performance in AI-driven information retrieval," SSRN, 2025.
[7] "Efficient and transferable agentic knowledge graph RAG via reinforcement learning," arXiv, 2025.
[8] Y. Li, "Deep reinforcement learning: An overview," arXiv, 2018.
[9] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of Artificial Intelligence Research*, 1996.
[10] K. Zhang, Z. Yang, and T. Basar, "A survey on multi-agent deep reinforcement learning," *Artificial Intelligence Review*, Springer, 2021.
[11] "A comprehensive survey on reinforcement learning-based agentic search: Foundations, roles, optimizations, evaluations, and applications," arXiv, 2025.
[12] H. Luo *et al.*, "Graph-R1: Towards agentic GraphRAG framework via end-to-end reinforcement learning," arXiv, 2025.
[13] Q. Zheng *et al.*, "End-to-end agentic RAG system training for traceable diagnostic reasoning," arXiv, 2025.
[14] H. K. Yoo, "Agentic RAG: Reinforcement learning-based insight-centric search generation," KoreaScience, 2025.
[15] N. Bagan, "Optimizing agentic retrieval-augmented generation using reinforcement learning-based methods," Leiden Institute of Advanced Computer Science (LIACS), 2025.
[16] A. Nagori *et al.*, "Open-source agentic hybrid RAG framework for scientific literature review," arXiv, 2025.

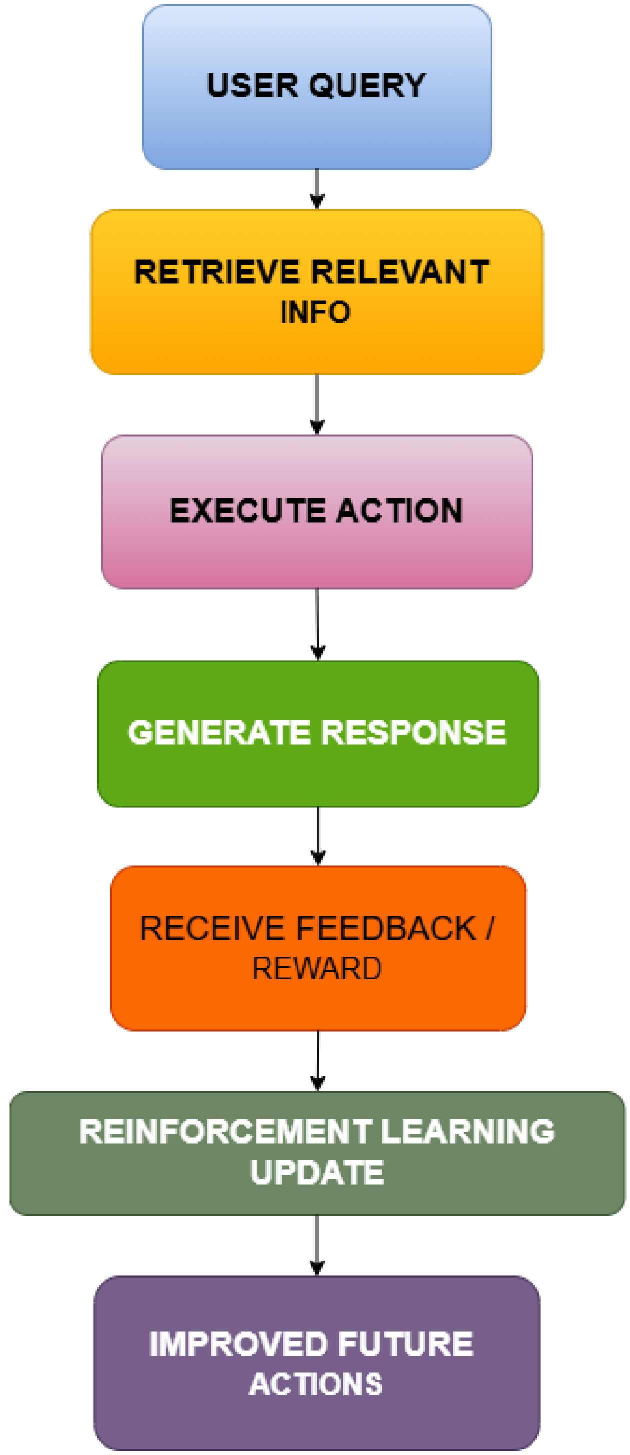

Fig. 5: Reinforcement Learning Workflow