Retrieval Feedback Memory Enhancement Large Model Retrieval Generation Method

Leqian Li[1], **Dianxi Shi**[2*], **Jialu Zhou**[1], **Xinyu Wei**[1], **Mingyue Yang**[1], **Songchang Jin**[3], **Shaowu Yang**[1]

[1]College of Computer Science and Technology, National University of Defense Technology, china
[2]Advanced Institute of Big Data, Beijing, China
[3]Intelligent Game and Decision Lab, Beijing 100091, China
{llq,dxshi,zhoujialu23,xinyuwei,yangmingyue216,shaowu.yang}@nudt.edu.cn

Abstract

Large Language Models (LLMs) have shown remarkable capabilities across diverse tasks, yet they face inherent limitations such as constrained parametric knowledge and high retraining costs. Retrieval-Augmented Generation (RAG) augments the generation process by retrieving externally stored knowledge absent from the model's internal parameters. However, RAG methods face challenges such as information loss and redundant retrievals during multi-round queries, accompanying the difficulties in precisely characterizing knowledge gaps for complex tasks. To address these problems, we propose Retrieval Feedback and Memory Retrieval Augmented Generation(RFM-RAG), which transforms the stateless retrieval of previous methods into stateful continuous knowledge management by constructing a dynamic evidence pool. Specifically, our method generates refined queries describing the model's knowledge gaps using relational triples from questions and evidence from the dynamic evidence pool; Retrieves critical external knowledge to iteratively update this evidence pool; Employs a R-Feedback Model to evaluate evidence completeness until convergence. Compared to traditional RAG methods, our approach enables persistent storage of retrieved passages and effectively distills key information from passages to construct clearly new queries. Experiments on three public QA benchmarks demonstrate that RFM-RAG outperforms previous methods and improves overall system accuracy.

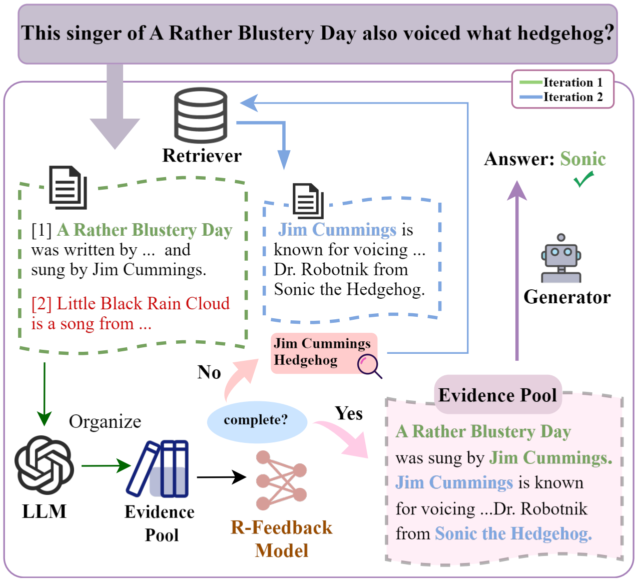

This singer of A Rather Blustery Day also voiced what hedgehog?
Iteration 1
Iteration 2
Retriever
Answer: Sonic
[1] A Rather Blustery Day was written by ... and sung by Jim Cummings.
[2] Little Black Rain Cloud is a song from ...
Jim Cummings is known for voicing ... Dr. Robotnik from Sonic the Hedgehog.
Generator
Jim Cummings Hedgehog
No
complete?
Yes
Evidence Pool
A Rather Blustery Day was sung by Jim Cummings. Jim Cummings is known for voicing ...Dr. Robotnik from Sonic the Hedgehog.
Organize
LLM
Evidence Pool
R-Feedback Model

Figure 1: RFM-RAG employs an LLM to distill knowledge from retrieved results, dynamically updating an evidence pool. A R-Feedback Model then assesses the pool's completeness. If sufficient, evidence is passed to the generation model for final response. Otherwise, we formulates new queries combining evidence pool's content with the question for iterative retrieval. Compared to previous methods, RFM-RAG enables persistent knowledge retention and extracts critical information to retrieve.

Introduction

In recent years, large language models (LLMs) have been widely applied in various natural language processing (NLP) tasks owing to their advanced comprehension and generation capabilities (Radford et al. 2018; Chowdhery et al. 2023; Touvron et al. 2023). However, the parameter knowledge of the model remains static after pre-training. Therefore, when answering questions beyond their pretraining scope or requiring up-to-date domain knowledge, they may generate text that is syntactically fluent but factually ungrounded. This phenomenon is called hallucination (Maynez et al. 2020; Zhou et al. 2020).

To mitigate hallucination issues, Retrieval-Augmented Generation (RAG)(Lewis et al. 2020) retrieves relevant knowledge from external sources in a single pass based on user input and integrating the information into LLM prompts to enhance factual accuracy in responses. While effective for simple knowledge-intensive tasks (Ram et al. 2023), this approach performs poorly in complex scenarios requiring such as multi-step reasoning (Paranjape et al. 2023), fact verification (Thorne et al. 2018), and Long-form generation (Fabbri et al. 2021). Compared to simple tasks, these tasks require higher standards for the knowledge to be acquired. For example, Long-form generation necessitates iterative knowledge gathering throughout the generation process. Multi-hop QA requires step dependent queries where each retrieval re-

[*] Corresponding author.

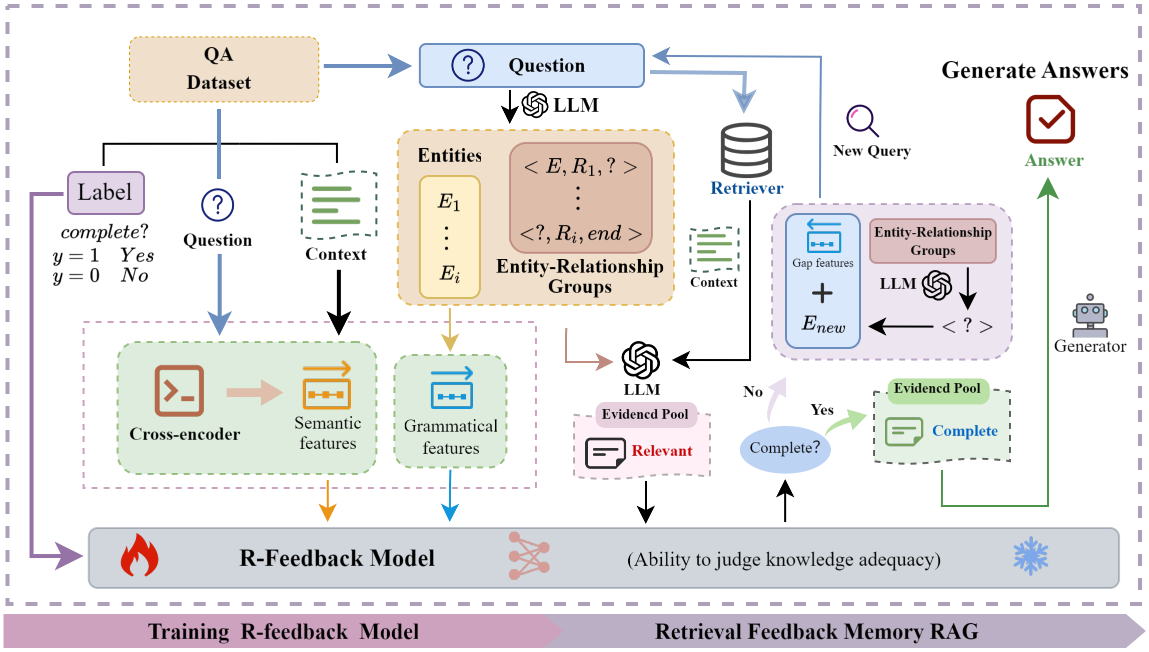


Figure 2: RFM-RAG dynamically constructs an evidence pool by processing retrieved results and formulates targeted queries until termination. The workflow begins with the original question as the initial query. An LLM then curates retrieved passages, filtering noise while integrating relevant knowledge into the evidence pool. Then, the R-Feedback Model evaluates knowledge sufficiency. If deficient, new queries are created from core question entities and evidence-pool information, iteratively enriching the evidence pool through retrieval. Upon achieving comprehensive evidence coverage, the LLM generates the final response.

lies on prior outputs. In contrast, iterative retrieval methods generate multiple retrieval queries and modify the retrieval queries based on feedback information through multi-round of retrieval refinement to obtain the final results(Asai et al. 2024). Dynamic retrieval RAG performs multiple retrievals during the LLM generation process. FLARE (Jiang et al. 2023) uses the part of the last generated sentence to perform retrieval when the LLM's confidence (i.e., the generation probability) on the next token is lower than certain thresholds. Some methods decompose the original question into multiple sub-questions when dealing with multi-step QA problems, retrieve external information separately and integrate multiple pieces of information as the answer.

However, iterative RAG approaches suffer from two critical limitations. Generated outputs depends on retrieved documents. low-quality retrievals introduce noise that reduce the accuracy of response. Repeated calls of the full retrieval-generation pipeline results in unnecessary resource overhead. We believe that when provided with sufficient knowledge, LLMs can generate accurate answers in a single pass. Redundant generation steps in conventional iterative RAG may have hallucinations. Thus, comprehensive knowledge completeness assessment before LLMs input is essential. Furthermore, since knowledge gaps change dynamically during iteration, each retrieval requires precise queries targeting the model's current state.

To address these limitations, we propose Retrieval-Feedback Augmented Memory Enhanced Large Model Retrieval and Generation(RFM-RAG). As shown in **Fig.1**, our method constructs a dynamic evidence pool where LLMs COT prompting (Wei et al. 2022) organize and deduplicate retrieved contexts to eliminate low-quality results. By formulating new retrieval queries through combined integration of the evidence pool and original question, we precisely target knowledge gaps beyond existing evidence. This evidence pool undergoes iterative refinement through successive queries until the Retrieval Feedback Model (R-Feedback Model) considers the evidence collection final. In summary, our main contributions are as follows:

- We propose an iterative retrieval-based dynamic evidence pool construction method, leveraging chain-of-thought prompts to guide LLMs in relevance filtering, structural organization, and deduplication of retrieved context. The refined information serves as validated evidence, progressively building a high-quality knowledge reservoir for final generation.
- We design a targeted query generation mechanism that pinpoints the knowledge gaps of LLMs, enabling precise localization of missing information beyond the evidence pool. Additionally, we innovatively introduce a dedicated R-Feedback Model to evaluate the sufficiency of evidence, and release a specialized dataset for training.
- We conduct comprehensive evaluations of previous RAG methods and RFM-RAG across three benchmark datasets using two distinct LLMs. The experimental results demonstrate improvements achieved by RFM-RAG, confirming the efficacy of our approach.

Related Work

Retrieval-Augmented Generation

RAG effectively mitigates hallucination with single-round retrieval enhancement being the most straightforward approach, retrieving knowledge using the original query, integrating relevant passages and prompting the LLM with augmented input.Foundational studies have extensively explored this paradigm (Khandelwal et al. 2019; Borgeaud et al. 2022; Izacard and Grave 2020; Guu et al. 2020). However, these methods are only suitable for simple tasks or unambiguous queries.

In complex scenarios requiring multi-hop reasoning or inference, single-round retrieval often fails to capture the knowledge necessary for accurate generation precisely. As a result, recent research has focused on advanced RAG strategies. IRCot(Trivedi et al. 2022) employs chain-of-thought reasoning to iteratively generate retrieval queries. Adaptive-RAG(Jiang et al. 2023) categorizes questions into three modes based on complexity and dynamically adjusts retrieval rounds. Self-RAG(Asai et al. 2024) produces reflective tokens to guide retrieval-generation interplay. DRAGIN(Su et al. 2024) performs real-time retrieval activated by LLM uncertainty signals during generation.

Retrieval Quality Assessment Metrics

Evaluating the generated outputs of large language models (LLMs) is a critical step in assessing RAG effectiveness. This process quantifies generation quality using multidimensional metrics including (factual accuracy, answer relevance, and text diversity), which collectively reflect RAG's comprehensive performance (Es et al. 2024). For the core retrieval component of RAG, accurate evaluation can effectively avoid unnecessary retrieval steps. Current mainstream approaches rely on quantitative metrics, which compute statistical similarity between retrieved passages and queries. Methods such as BLEU(Papineni et al. 2002), ROUGE(Lin 2004), and METEOR(Banerjee and Lavie 2005) evaluate relevance through surface term matching (e.g., n-gram overlap) but fundamentally ignore semantic depth. While providing measurable evaluation standards, these approaches face significant limitations in real-world applications due to insufficient understanding of deep semantics.

Methodology

Previous RAG methods suffer from over-reliance on single-round retrieval results and an inherent inability to accurately identify model knowledge gaps, frequently leading to outputs that are factually incorrect. To address these limitations, we introduce the Retrieval-Feedback Memory-enhanced RAG (RFM-RAG) framework, detailed in this section with architectural overview in **Fig.2**. Our methodology is based on three core principles: Dynamically constructing an evidence pool by aggregating and organizing retrieved passages for each retrieval. Using a retrieval feedback model to terminate retrieval loops upon verifying evidence pool sufficiency. Generating iterative queries through relational chain-based knowledge gap detection to address missing information.

Dynamic Evidence Pool Construction

We define the vanilla LLM generation process as $\mathrm{Ans} = \mathrm{LLM}(q)$, where the LLM directly generates answers from queries. Traditional RAG methods follow $\mathrm{Ans} = \mathrm{LLM}(q, e)$ with $e = \mathcal{R}(C \mid q)$, where $\mathcal{R}$ denotes the retriever, e represents relevant knowledge retrieved from corpus C given q, and both q and e are input to the LLM. This paradigm suffers from incomplete retrieval and inaccurate knowledge gap identification due to the limitations of single-round retrieval. To overcome this, RFM-RAG constructs a dynamic evidence pool through iterative retrieval, leveraging R-Feedback Model(As details in the next section) to score evidence completeness and determine termination. The process initializes with the original question q_0. Subsequent retrievals use generated queries q_i, each of which obtains retrieved passages $K_i = \{k_1, k_2, \dots\}$. Using chain-of-thought prompting (Wei et al. 2022), we instruct the LLM (GPT-3.5-turbo)(Brown et al. 2020) to curate retrieved passages (As shown in **Fig.3**). This curation process involves filtering redundancies while extracting question-relevant evidence and incrementally augment the evidence pool. This evidence accumulation process is formally defined as:

$$K_i = \mathcal{R}(C \mid q_i) \quad E_i = LLM_{prompt1}(q_i, K_i) \tag{1}$$

$$E = \{E_0, E_1, \dots, E_i\} \tag{2}$$

Prompt 1: Organize Docs from Retrieve

You will receive two messages: questions and contextual paragraphs. As a professional document processing assistant, your task description is as follows:
Extract and organize information directly related to entities in the problem from contextual paragraphs. Only organize relevant facts for each entity. The output should be presented in the form of organized and simple paragraph sentences.
If there is no information related to the problem in the context paragraph, the output is empty.

Figure 3: Prompt1 for Organizing Passages Using LLMs.

Before the R-Feedback Model decides to terminate the evidence pool construction, each retrieval iteration requires formulating a new query for extracting necessary information from external databases. Most RAG methods leverage query expansion or rewriting techniques. These methods parse semantic features and metadata within queries. However, they fail to capture critical information from retrieved passages. Consequently, we propose a query generation strategy based on knowledge gap detection, designed to more identify missing knowledge in LLM responses by detecting entity deficiencies in the query and newly emerged relevant entities in the evidence pool.

We quantify entity gaps with entity coverage metrics. Specifically, chain-of-thought prompting instructs the LLM to extract key entities k and relational triples r_k from the question, replacing unknown information with placeholders <X>. For the question *Is the director of Move (1970 Film) and the director of Méditerranée (1963 Film) the same country?*, this extracts entities *Move, Méditerranée* and triples *(Move,director,<X>),(Méditerranée, director,<X>), (<X>,country,end)*. The entity coverage feature S_{f_k} is computed as the proportion of knowledge about entity k present in the current evidence pool E. When S_{f_k} falls below preset threshold θ, it indicates insufficient entity information in E, prompting addition to the gap list G' to represent unretrieved entity knowledge.

$$(k, r_k) = LLM_{prompt2}(q_0) \tag{3}$$

$$S_{f_k} = min(\frac{C_{k_E}}{L_E}, 1.0) \tag{4}$$

$$G' = \begin{cases} \text{Add}(G, k) & \text{if} \quad S_{f_k} < \theta \\ G & \text{otherwise} \end{cases} \tag{5}$$

where C_{k_E} is the number of occurrences of entity k in the current evidence pool E, L_E is the length of the evidence pool information. Appendix C introduces the prompt templates used to extract entities and Entity-Relationship groups from the question.

Subsequently, we extract new question-related entities from the evidence pool. We input the extracted relational triples and evidence pool into the large model (As shown in **Fig.4**), enabling the model to retrieve missing information z_k represented by placeholders in the triples from the evidence pool and add to the knowledge gap list $G = \{g_1, g_2, ..., g_n, z_1, z_2, \ldots, z_k\}$. This augmentation captures entities indirectly related to the question that cannot be obtained from the initial question. These entities form the core for subsequent retrievals. Finally, a new query q_i is constructed using lexical items in G, designed to cover the knowledge gaps requiring external knowledge base retrieval for accurate question answering by the LLM.

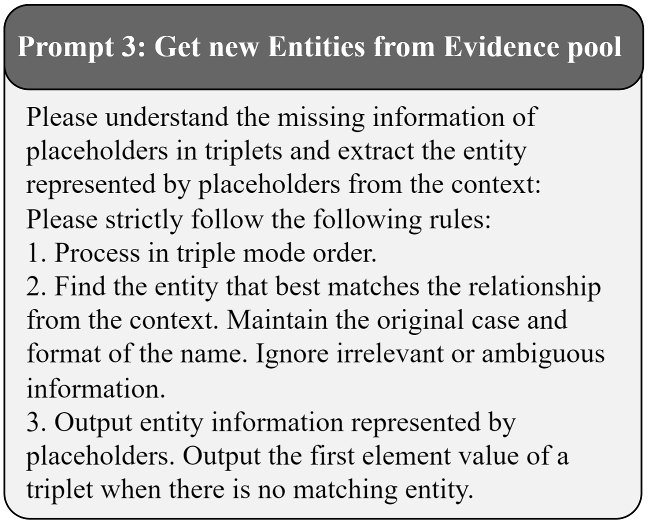
$$z_k = LLM_{prompt3}(r_k, E) \tag{6}$$

R-Feedback Model

We design the R-Feedback Model as a feed-forward network with double hidden layers and activation functions. We compute the syntactic entity coverage feature S_f and semantic relevance feature G_f from the evidence pool E and the initial question q_0. These features serve as input to the R-Feedback Model, which decides when to terminate evidence pool updates.

Therefore, we utilize the entity coverage calculated in Equation 4 for each key entity k. We take the average of the coverage features of all entities to obtain the overall syntactic coverage of the entire question in the evidence pool, which is used to describe the syntactic relevance between the question and the evidence pool. Where $|E|$ is the number of entities extracted from q_0:

$$S_f = \frac{sum(S_{f_k})}{|E|} \tag{7}$$



Prompt 3: Get new Entities from Evidence pool

Please understand the missing information of placeholders in triplets and extract the entity represented by placeholders from the context:
Please strictly follow the following rules:
1. Process in triple mode order.
2. Find the entity that best matches the relationship from the context. Maintain the original case and format of the name. Ignore irrelevant or ambiguous information.
3. Output entity information represented by placeholders. Output the first element value of a triplet when there is no matching entity.

Figure 4: Prompt3 for Extracting Placeholder-Represented Entities from Evidence Pool.

Cross-encoders process queries and paragraphs concurrently through deep attention mechanisms, capturing complex semantic relationships with high accuracy. We derive semantic relevance features G_f by processing the evidence pool E and initial query q_0 using cross-encoder, and then fuse the two features as input to the R-Feedback Model RF_{model}:

$$G_f = Encoder_{cross}(q_0, E) \tag{8}$$

$$Logit_S G = RF_{model}(S_f, G_f) \tag{9}$$

Using the value of $Logit_S G$, R-Feedback Model decides if the condition for updating the evidence pool has been met.

Question: Which film was released more recently, *Die schöne Lurette* or *Sabhash*?	**Context:** Hobby won the Award for Best First Time Director. Karl Geary... **Label:0**
Question: Were **Dan O'Connor** and **Hale Baugh** from the same country?	**Context: Daniel O'Connor** was a **Canadian** politician, businessman... **Hale Baugh** was an **American** modern pentathlete. He competed... **Label: 1**
Question: Which film has the director who was born later, *Il Diavolo In Convento* or **The Enchanting Enemy**?	**Context:** Il diavolo in corpo is an... **The Enchanting Enemy** is an Italian comedy film **directed by Claudio Gora** and starring... **Label: 0**

Table 1: Examples of datasets for R-Feedback Model. Information relevant to the question and context is marked in bold, and entities with missing information in the context are marked in italics.

		2WikiMultihopQA		NaturalQA		StrategyQA	Average	
LLM	**RAG Method**	**EM**	**ACC**	**EM**	**ACC**	**ACC**	**EM**	**ACC**
Gemma-2b	No Retrieval	22.6	43.0	15.0	24.6	56.0	18.8	41.2
	Vanilla RAG	22.8	38.4	11.4	26.0	56.3	17.1	40.2
	Probing-RAG	24.2	**43.6**	21.6	**35.0**	61.8	22.9	**46.8**
	Adaptive RAG	21.6	40.6	11.4	26.2	54.7	16.5	40.5
	DRAGIN	26.4	28.8	18.8	22.2	62.4	22.6	37.8
	RFM-RAG(Ours)	**29.2**	37.6	**30.6**	33.2	**63.2**	**29.9**	44.7
Mistral-7b	No Retrieval	16.4	30.0	13.2	19.8	62.4	14.8	37.4
	Vanilla RAG	21.6	32.6	16.8	35.0	60.7	19.2	42.7
	Probing-RAG	23.0	33.4	20.8	39.4	61.5	21.9	44.7
	Adaptive RAG	22.6	31.6	17.2	37.4	65.4	19.9	44.8
	DRAGIN	23.2	25.8	16.8	37.2	70.3	20.0	44.4
	RFM-RAG(Ours)	**32.1**	**36.7**	**33.4**	**42.8**	**72.6**	**32.7**	**50.7**

Table 2: Experimental results on three different QA datasets. We indicate the highest performance in bold and underline the second highest.

Training R-Feedback Model

Training the retrieval feedback model requires dataset pairs $((q, E), y)_1^N$, where q denotes the question, E represents a knowledge segment, and $y \in 0, 1$ indicates sufficiency of E to answer q. To generate these pairs, we use the evidential chain corresponding to the answer to question q in the dataset to divide the context into supporting evidence and irrelevant information. Sufficient samples ($y = 1$) use gold supporting evidence from the dataset as E, indicating E fully answers q without further retrieval. Insufficient samples ($y = 0$) assign irrelevant information to E, denoting E cannot answer q. Partially sufficient samples ($y = 0$) combine subsets of supporting with irrelevant information as E, simulating scenarios where E contains relevant but incomplete knowledge requiring additional retrieval.

As detailed in Table 1, our training dataset comprises three data categories derived from the public 2WikiMultihopQA (Ho et al. 2020) dataset. To ensure a balanced distribution of positive and negative samples, we randomly selected questions and generated paired samples for each category: sufficient evidence ($y = 1$) and insufficient evidence ($y = 0$). The final dataset contains 10,000 training and 800 validation samples. We trained the R-Feedback Model using this dataset, with cross-entropy loss defined as follows:

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)]$$

We provide details on the hyperparameters for training the R-Feedback Model in Appendix A.

Experimental Setups

Datasets

For performance assessment, we evaluate methods using three open-domain QA datasets, randomly sampling 500 test examples per dataset. Comprehensive dataset and corpus specifications are provided in Appendix B.

2WikimultihopQA (Ho et al. 2020). It contains multi-hop questions that span more than two Wikipedia pages, each provided with 10 paragraphs. The dataset features fine-grained paragraph annotations and a high proportion of distractors, which enables rigorous testing of models' multi-hop reasoning in noisy environments.

NaturalQA (Kwiatkowski et al. 2019). Answers must be found in long documents to locate the exact fragments. Due to the real distribution of user questions and the challenge of locating answers, this task effectively tests the model's ability to extract accurate information from long, open-domain texts.

StrategyQA (Geva et al. 2021). It contains binary questions requiring implicit reasoning strategies without providing explicit evidence paragraphs. Characterized by strategic reasoning requirements, it assesses models' ability to construct evidence chains and perform complex inference.

Baselines

We choose the following Text Generation baselines for comparison. **No Retrieval**. Directly generates answers from the original question without retrieval.**Vanilla RAG**(Lewis et al. 2020). Relevant passages are retrieved from an external corpus based on the initial question. The retrieved passages are then added into the LLM's input. **DRAGIN**(Su et al. 2024). Retrieves when token-level confidence drops, using attention weights to construct queries from contextually salient words. **Adaptive-RAG**(Jeong et al. 2024). Classifies question complexity via fine-tuned classifier to dynamically adjust retrieval steps. **Probing-RAG**(Baek et al. 2024). Leverages intermediate-layer hidden states to determine need for additional retrieval.

All methods were evaluated under few-shot settings: 4-shot on 2WikiMultihopQA and NaturalQA, 6-shot on StrategyQA. Answer extraction used regular expression pattern matching to structure free-form LLM outputs into precise final answers. For evaluation, we used answer-level exact match(EM) and accuracy(ACC) scores to compare extracted answers against reference labels. Given diminishing accuracy gains and significant latency increases beyond three retrieval rounds, we capped maximum number of retrievals at three.

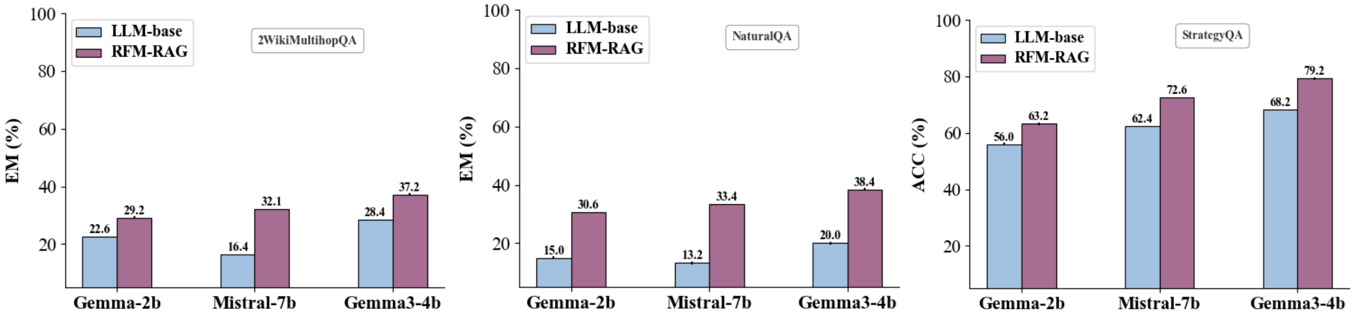


Figure 5: EM and ACC scores for QA without retrieval and RFM-RAG based on Gemma-2b, Mistral-7b, and Gemma3-4b models. RFM-RAG outperforms the generation models themselves on all three datasets and all models.

		NaturalQA			StrategyQA		
	Methods	**R-Step**	**Δ**	**EM**	**R-Step**	**Δ**	**ACC**
Gemma-2b	**RFM**	1.93		**30.6**	2.31		**63.2**
	wo-RFM	3	**1.07**↓	29.5	3	**0.69**↓	62.8
Mistral-7b	**RFM**	2.34		33.4	2.62		**72.6**
	wo-RFM	3	**0.66**↓	**35.8**	3	**0.38**↓	70.2

Table 3: Comparison of averaged retrieval steps and EM, ACC (%) between RFM-RAG and the ablated wo-RFM approach (Evidence pool construction termination fixed at maximum retrieval count 3) using Gemma-2b and Mistral-7b models.

Implementation Details

We employ BM25(Robertson and Jones 1976), a probabilistic sparse retrieval model based on (Robertson, Zaragoza et al. 2009), which demonstrates superior performance in RAG, even surpassing certain dense retrievers. This is implemented via ElasticSearch for all methods to ensure fairness. For 2WikiMultihopQA, we adopt IRCoT's(Trivedi et al. 2022) document corpus. StrategyQA averages 2.7 evidence documents per question and has no official corpus, while NaturalQA provides only answer-containing documents. Consequently, we constructed dedicated corpora using dataset contexts (details in Appendix B). All RAG methods utilize Gemma-2b(Team et al. 2024) and Mistral-7b(Albert Q. Jiang et al. 2023) as QA models. For computing resources, we utilize A100 GPUs with 40GB memory. In addition, due to the significant costs associated with evaluating retrieval-augmented generation models, we conducted experiments with a single run.

Experimental Results

Main Results

Our experiments comprehensively evaluated the performance of RFM-RAG on three datasets against various baselines, with results shown in Table 2. Our observations indicate that in most cases, single-round retrieval RAG consistently outperformed direct LLMs generation in question answering and confirming the efficacy of retrieval augmentation for knowledge-intensive QA tasks. The RFM-RAG method showed excellent performance on the majority of LLMs and datasets. Compared to no retrieval and single-round retrieval methods, on the Gemma-2b model, EM improved by approximately 11.1 and 12.8 percentage points, ACC improved by 3.5 and 4.5 percentage points. On the Mistral-7b model, EM improved by 17.9 and 13.5 percentage points, ACC improved by approximately 13.3 and 8 percentage points. This demonstrates the robustness and effectiveness of RFM-RAG in terms of knowledge collection and organization, as well as its ability to detect knowledge gaps in the model.

Notably, RFM-RAG demonstrates consistent performance gains on Gemma-2b, proving that models with fewer parameters can achieve competitive QA performance when provided with sufficient relevant information. Adaptive-RAG underperforms significantly across datasets. While it adjusts retrieval based on question complexity, the method lacks iterative enhancement targeting model's specific knowledge gaps. The RFM-RAG we propose outperforms all previous adaptive retrieval methods by avoiding redundant generation cycles. By constructing a dynamic evidence pool through detecting model knowledge gaps, our method achieves significant performance improvements.

Analysis

RFM-RAG performance is unaffected by the generation model. To investigate the impact of the generation model's inherent capabilities on the retrieval augmentation methods, we conducted supplementary experiments on the latest model, **Gemma3-4b** (Team et al. 2025). The experimental setup is identical to that of the main experiment: the retriever uses BM25 (implemented in ElasticSearch), the corpus is the same as in Appendix B. The prompt engineering uses the same mind chain template. **Fig.5** compares the ability of the three generative models based solely on parameter-

Question	DRAGIN	RFM-RAG(Ours)
Who is the mother of the director of film Polish-Russian War? (2WikiMultihopQA)	**Query 1:** mother director film Polish-Russian War? director film Polish-Russian War **Knowledge for LLM:** bombs would be like the early. . . The event attracted an audience. . . university leaders to combat a wide array. . . **Query 2:** mother director film Polish-Russian War. . . mother Andrzej Wajda Zofia Wajda. **Knowledge for LLM:** bombs would be like the early. . . The event attracted. . . mentioned a few times in the Torah and references. . . **Answer:** Zofia Wajda **EM: 0**	**Query 1:** Who is the mother of the director of film Polish-Russian War? **Knowledge for LLM:** The director of the film ”Polish-Russian War” is Xawery Żuławski. **Query 2:** Xawery Żuławski. **Knowledge for LLM:** The director of the film ”Polish-Russian War” is Xawery Żuławski. Małgorzata Braunek is the mother of Xawery Żuławski, the Polish film director. **Answer:** Małgorzata Braunek **EM: 1**
what is the name of the rca victor dog? (NaturalQA)	**Query 1:** name rca victor dog Bristol, served model painting Francis Barraud titled Master's. **Knowledge for LLM:** were the first in provincial ...Ginsberg, In this mode perfection is basic,... Nipper(1884–1895)was a dog from Bristol, who served as the model for a painting. **Query 2:** image basis dog-and-gramophone trademark, Berliner's successor Co. Victor Records) **Knowledge for LLM:** were the first in provincial... This image was the basis for the dog-and... Berliner' American successor the Victor Talking Machine Co. (later known as RCA Victor). **Answer:** Berliner **EM: 0**	**Query 1:** what is the name of the rca victor dog? **Knowledge for LLM:** Berliner's successor the Victor Talking Machine Co. (later known as RCA Victor) **Query 2:** Berliner **Knowledge for LLM:** Nipper(1884–1895)was a dog, who served as the model for a painting.This image was the basis for the dog-and-gramophone trademark that was used by Berliner's successor the Victor Talking Machine Co.(later known as RCA Victor). **Answer:** Nipper **EM: 1**

Table 4: Case study with the RFM-RAG and DRAGIN.

ized knowledge with the ability of our RFM-RAG to answer questions on three datasets. For all three models, RFM-RAG outperforms the others across all datasets. Especially for the latest generative model(Gemma3-4b), RFM-RAG improves the EM or ACC score by 8.8 points on 2WikiMultihopQA, 18.4 points on NaturalQA, and 11 points on StrategyQA, relative to the model's inherent generative capability.

Evaluating Retrieval Feedback Model's Iteration Termination Efficacy. Compared to fixed-iteration baselines that terminate without considering knowledge sufficiency, our method employs early termination when sufficient evidence is acquired. This strategy significantly reduces latency and mitigates noise from redundant retrievals. We empirically compared the step counts and Exact Match (EM) scores between the Fixed-iteration baseline (wo-RFM) and RFM-RAG's adaptive termination on NaturalQA and StrategyQA datasets. Table 3 shows that unnecessary retrieval beyond knowledge saturation leads to a reduction in accuracy by 0.4 to 2.4 percentage points on average, while RFM-RAG achieves latency reductions of 12-35% and maintains comparable or superior accuracy, validating the efficacy of our retrieval feedback mechanism.

Case Study. We conducted a case study comparing RFM-RAG and DRAGIN qualitatively on 2WikiMultihopQA and NaturalQA question pairs(Table 4), analyzing retrieval queries, knowledge provisioning, and final answers. In Case 1 (complex multi-hop QA), RFM-RAG extracts key entities from retrieval results as subsequent queries. The second retrieval provides targeted knowledge for accurate answer generation. Conversely, DRAGIN relies on generation-based knowledge inference after first retrieval, introducing uncertainty. DRAGIN extracts missing knowledge from model-generated information after the initial retrieval. However, due to the uncertainty of model generation, its accuracy is weaker than the knowledge extracted from authentic evidence pool related to the question.

In Case 2, which requires information integration from multiple knowledge sources, RFM-RAG processes and retains all retrieved evidence throughout iterations. During final generation, the LLM filters relevant information from the complete evidence pool to formulate answers. DRAGIN fails to retain previously retrieved passages in subsequent retrievals. As a result, even when partial answers are generated from prior knowledge, the lack of critical evidence undermines the integrity of the final conclusion.

Conclusion

In this work, we introduce RFM-RAG, a novel retrieval pipeline that employs a relationship chain-based query generation pattern that enables precise multi-round of retrieval. During this process, the LLM organizes and deduplicates the retrieved results to construct a comprehensive evidence pool. To optimize the retrieval process, RFM-RAG incorporates an R-Feedback Model, which is responsible for determining when to stop updating the evidence pool during the retrieval rounds. This model ensures that retrievals continue only as long as necessary to gather relevant evidence. We introduce both the training dataset and method for the R-Feedback Model and show that RFM-RAG outperforms previous methods for various QA datasets.

References

Albert Q. Jiang, A. M., Alexandre Sablayrolles; et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

Asai, A.; Wu, Z.; Wang, Y.; Sil, A.; and Hajishirzi, H. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection.

Baek, I.; Chang, H.; Kim, B.; Lee, J.; and Lee, H. 2024. Probing-rag: Self-probing to guide language models in selective document retrieval. *arXiv preprint arXiv:2410.13339*.

Banerjee, S.; and Lavie, A. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 65–72.

Borgeaud, S.; Mensch, A.; Hoffmann, J.; Cai, T.; Rutherford, E.; Millican, K.; Van Den Driessche, G. B.; Lespiau, J.-B.; Damoc, B.; Clark, A.; et al. 2022. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, 2206–2240. PMLR.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H. W.; Sutton, C.; Gehrmann, S.; et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113.

Es, S.; James, J.; Anke, L. E.; and Schockaert, S. 2024. Ragas: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 150–158.

Fabbri, A. R.; Kryściński, W.; McCann, B.; Xiong, C.; Socher, R.; and Radev, D. 2021. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9: 391–409.

Geva, M.; Khashabi, D.; Segal, E.; Khot, T.; Roth, D.; and Berant, J. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9: 346–361.

Guu, K.; Lee, K.; Tung, Z.; Pasupat, P.; and Chang, M. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, 3929–3938. PMLR.

Ho, X.; Nguyen, A.-K. D.; Sugawara, S.; and Aizawa, A. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*.

Izacard, G.; and Grave, E. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*.

Jeong, S.; Baek, J.; Cho, S.; Hwang, S. J.; and Park, J. C. 2024. Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity. *arXiv preprint arXiv:2403.14403*.

Jiang, Z.; Xu, F. F.; Gao, L.; Sun, Z.; Liu, Q.; Dwivedi-Yu, J.; Yang, Y.; Callan, J.; and Neubig, G. 2023. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 7969–7992.

Khandelwal, U.; Levy, O.; Jurafsky, D.; Zettlemoyer, L.; and Lewis, M. 2019. Generalization through memorization: Nearest neighbor language models. *arXiv preprint arXiv:1911.00172*.

Kwiatkowski, T.; Palomaki, J.; Redfield, O.; Collins, M.; Parikh, A.; Alberti, C.; Epstein, D.; Polosukhin, I.; Devlin, J.; Lee, K.; et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7: 453–466.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, 74–81.

Maynez, J.; Narayan, S.; Bohnet, B.; and McDonald, R. 2020. On faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2005.00661*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 311–318.

Paranjape, B.; Lundberg, S.; Singh, S.; Hajishirzi, H.; Zettlemoyer, L.; and Ribeiro, M. T. 2023. Art: Automatic multi-step reasoning and tool-use for large language models. *arXiv preprint arXiv:2303.09014*.

Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I.; et al. 2018. Improving language understanding by generative pre-training.

Ram, O.; Levine, Y.; Dalmedigos, I.; Muhlgay, D.; Shashua, A.; Leyton-Brown, K.; and Shoham, Y. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11: 1316–1331.

Robertson, S.; Zaragoza, H.; et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4): 333–389.

Robertson, S. E.; and Jones, K. S. 1976. Relevance weighting of search terms. *Journal of the American Society for Information science*, 27(3): 129–146.

Su, W.; Tang, Y.; Ai, Q.; Wu, Z.; and Liu, Y. 2024. DRAGIN: dynamic retrieval augmented generation based on the information needs of large language models. *arXiv preprint arXiv:2403.10081*.

Team, G.; Kamath, A.; Ferret, J.; Pathak; et al. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Team, G.; Mesnard, T.; Hardin, C.; Dadashi, R.; Bhupatiraju, S.; Pathak; et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Thorne, J.; Vlachos, A.; Christodoulopoulos, C.; and Mittal, A. 2018. FEVER: a large-scale dataset for fact extraction and VERification. *arXiv preprint arXiv:1803.05355*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Trivedi, H.; Balasubramanian, N.; Khot, T.; and Sabharwal, A. 2022. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. *arXiv preprint arXiv:2212.10509*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Zhou, C.; Neubig, G.; Gu, J.; Diab, M.; Guzman, P.; Zettlemoyer, L.; and Ghazvininejad, M. 2020. Detecting hallucinated content in conditional neural sequence generation. *arXiv preprint arXiv:2011.02593*.