Improving Context Fidelity via Native Retrieval-Augmented Reasoning

Suyuchen Wang[1,3]*, Jinlin Wang[2]*, Xinyu Wang[4]*, Shiqi Li[2], Xiangru Tang[5], Sirui Hong[2], Xiao-Wen Chang[4], Chenglin Wu[2]†, Bang Liu[1,3,6]†

[1]DIRO, Université de Montréal [2]MetaGPT [3]Mila - Quebec AI Institute
[4]McGill University [5]Yale University [6]Canada CIFAR AI Chair

Abstract

Large language models (LLMs) often struggle with context fidelity, producing inconsistent answers when responding to questions based on provided information. Existing approaches either rely on expensive supervised fine-tuning to generate evidence post-answer or train models to perform web searches without necessarily improving utilization of the given context. We propose **CARE**, a novel native retrieval-augmented reasoning framework that teaches LLMs to explicitly integrate in-context evidence within their reasoning process with the model's own retrieval capabilities. Our method requires limited labeled evidence data while significantly enhancing both retrieval accuracy and answer generation performance through strategically retrieved in-context tokens in the reasoning chain. Extensive experiments on multiple real-world and counterfactual QA benchmarks demonstrate that our approach substantially outperforms supervised fine-tuning, traditional retrieval-augmented generation methods, and external retrieval solutions. This work represents a fundamental advancement in making LLMs more accurate, reliable, and efficient for knowledge-intensive tasks.[1]

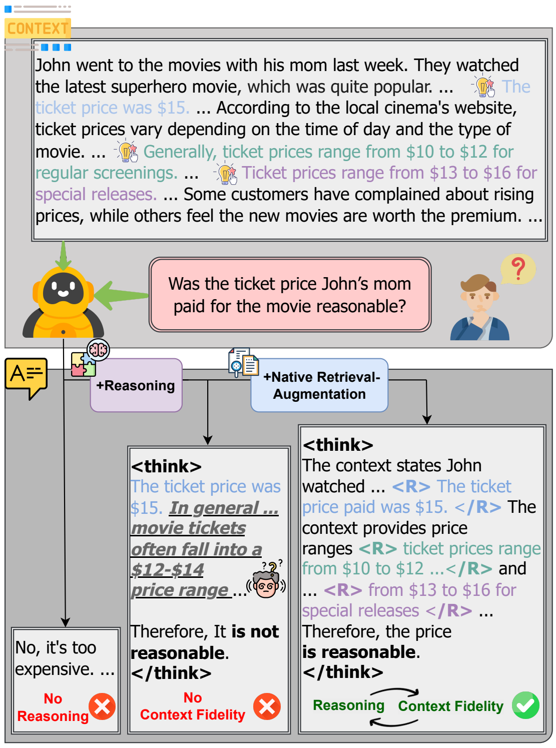

Figure 1: The Comparison among direct generation, reasoning-based generation, and reasoning with integrated in-context facts.

1 Introduction

Large language models (LLMs) have demonstrated impressive performance in a wide range of natural language tasks (Minaee et al., 2024; Liu et al., 2025a), yet continue to struggle with a fundamental challenge: maintaining fidelity to the context provided when answering questions (Talukdar and Biswas, 2024). This *context hallucination problem* (Chang et al., 2024; Hu et al., 2024; Liu et al., 2025b) is particularly pronounced in knowledge-intensive tasks where precise information retrieval and accurate reasoning are paramount. When LLMs generate answers that contradict or fabricate information relative to the input context, user trust declines, and the practical utility of these systems decreases considerably.

Current approaches to addressing this challenge fall into two broad categories, each with significant limitations. The first category employs retrieval-augmented generation (RAG) for evidence retrieval (Variengien and Winsor, 2023; Wang et al., 2024). Although this approach can improve explainability, it usually requires extensive labeled datasets with ground-truth evidence spans, making

*Equal contribution.
†Corresponding authors. Contact via email: bang.liu@umontreal.ca, alexanderwu@deepwisdom.ai.
[1]Homepage: https://foundationagents.github.io/CARE.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 21205–21218
November 4-9, 2025 ©2025 Association for Computational Linguistics

it prohibitively expensive to scale across diverse domains and languages. In addition, the extra retriever module and the vector database create excessiveness to the model architecture.

The second category leverages external retrieval mechanisms, allowing models to search for relevant information beyond their parametric knowledge (Hsu et al., 2024; Nguyen et al., 2024). Although effective in accessing up-to-date or specialized information, these approaches frequently underutilize the rich context already provided by users, which often contains the most relevant information for their specific scenarios. Furthermore, external retrieval introduces additional latency, complexity, and potential inconsistencies between the retrieved content and the original context.

In this paper, we introduce a fundamentally different approach: **native retrieval-augmented reasoning**. Rather than treating retrieval and reasoning as separate processes, our method teaches LLMs to dynamically identify and incorporate relevant evidence from the input context directly within their reasoning chain. This approach leverages the inherent *native* language understanding capabilities of LLMs to perform in-context retrieval without additional indexing or embedding systems, while simultaneously enhancing the reasoning process through explicit evidence integration.

Based on the aforementioned approach, we introduce the **Context-Aware Retrieval-Enhanced reasoning (CARE)** framework. The CARE framework requires limited labeled evidence data and operates through a two-phase training process: an initial supervised fine-tuning (SFT) phase that establishes the evidence integration pattern, followed by a reinforcement learning (RL) phase that refines the self-retrieval mechanism through retrieval-aware rewards. Crucially, we implement a curriculum learning strategy that enables the model to progressively adapt from simple to complex reasoning tasks, extending beyond the initial training distribution without requiring additional labeled data.

Our main contributions are as follows:

- We introduce **native retrieval-augmented reasoning**, a novel paradigm that organically combines in-context retrieval with structured reasoning to improve context fidelity and reduce hallucinations.

- We present a curated dataset for training models to perform evidence-integrated reasoning, which we have open sourced to facilitate further research in this area.

- We propose **CARE**, a comprehensive implementation that combines native retrieval-augmented reasoning with curriculum learning to handle diverse question-answering scenarios without additional labeled data.

- Through extensive experiments across multiple real-world and counterfactual QA benchmarks, we demonstrate that our approach substantially outperforms vanilla SFT, traditional RAG methods, and comparable models lacking in-context retrieval mechanisms in both evidence retrieval and answer accuracy.

Our work represents a significant advancement in making LLMs more accurate, reliable, and efficient for knowledge-intensive tasks, particularly when relevant information is already present in the input context. By teaching models to explicitly retrieve and reason with contextual evidence, we establish a stronger foundation for context-faithful language generation.

2 Related Work

2.1 LLM Reasoning on Question-Answering Tasks

Large language models (LLMs) have demonstrated impressive capabilities in complex reasoning tasks (Wei et al., 2022; Cobbe et al., 2021; Ouyang et al., 2022). Recent work has explored various prompting strategies to improve reasoning, including chain of thought prompting (Wei et al., 2022), which guides models to generate intermediate reasoning steps before producing final answers, and its variants such as zero-shot-CoT (Kojima et al., 2022) and self-consistency (Wang et al., 2022). More structured approaches include tree-of-thought (Yao et al., 2023a), graph-of-thought (Besta et al., 2024), ReAct (Yao et al., 2023b), and least-to-most prompting (Zhou et al., 2022). Despite these advances, LLMs still struggle to maintain context coherence when reasoning about long or noisy inputs (Xu et al., 2023; Li et al., 2024; Fei et al., 2024).

2.2 Retrieval-Augmented Generation

Traditional retrieval-augmented generation (RAG) methods (Guu et al., 2020; Lewis et al., 2020) enhance LLM by retrieving relevant passages from external corpora, alleviating the limitations of fixed parametric memory. This framework has been widely adopted for knowledge-intensive tasks (Xiong et al., 2024; Wang et al., 2025). Recent work has improved retrieval quality through techniques such as query expansion (Wang et al., 2023), re-ranking (Vu et al., 2024), and filtering (Asai et al., 2024), while others focus on robustness to noisy retrievals (Yoran et al., 2024). In-context retrieval methods aim to reuse relevant spans from the input sequence itself (Variengien and Winsor, 2023; Wang et al., 2024). However, both external and in-context RAG fundamentally rely on indexing and embedding-based retrieval pipelines, limiting their adaptability to complex or evolving contexts.

2.3 RL-Enhanced LLM Retrieval

Reinforcement learning (RL) has emerged as a powerful paradigm for optimizing LLM retrieval strategies (Humphreys et al., 2022; Tu et al., 2024; Hsu et al., 2024). Unlike traditional retrieval methods, RL-based approaches can learn adaptive retrieval policies that optimize for task-specific rewards (Kulkarni et al., 2024; Zhuang et al., 2025; Jin et al., 2025). Recent work has explored the use of RL to train retrieval policies that maximize answer accuracy (Hsu et al., 2024; Nguyen et al., 2024), combining the strengths of parametric knowledge and non-parametric retrieval (Mallen et al., 2022; Humphreys et al., 2022; Farahani and Johansson, 2024). Several approaches have used feedback mechanisms to improve retrieval quality, including relevance feedback (Zhou et al., 2023) and iterative refinement (Chen et al., 2024). However, most existing approaches still maintain a separation between the retrieval mechanism and the core reasoning process, potentially limiting the model's ability to integrate retrieved information in a context-aware manner.

3 The CARE Method

3.1 Overview

We present the CARE Method, a reasoning framework that enables LLMs to autonomously conduct native retrieval from the input context without relying on any external retrieval modules or tools, and integrate evidence retrieved by LLM's native capabilities into the reasoning process. By allowing the model to perform native retrieval, CARE better utilizes the language understanding capability of LLMs and user input while reducing the reliance on potentially expensive tool calling, while introducing native retrieval results in the reasoning can both improve the model's context loyalty and improve the reasoning process by utilizing curated evidence. Figure 1 illustrates the comparison between direct generation, reasoning-based inference, and CARE.

Labeling supporting facts for QA datasets is expensive, and thus CARE's design aims at reducing the reliance on such labels. Thus, the framework is designed to consist of two training phases: a supervised fine-tuning (SFT) phase followed by a reinforcement learning (RL) phase.

The first SFT phase is designed to improve the efficiency of RL training by familiarizing the model with the target output format with retrieved facts in the reasoning process. In this phase, the model is fine-tuned on a self-curated dataset comprising reasoning chain enriched with golden in-context retrieval snippets, guiding the model to align reasoning with context-derived evidence (Section 3.3).

The RL phase refines the self-retrieval mechanism through retrieval-aware rewards, reinforcing evidence consistency and logical coherence with the help of native retrieval results from the input context throughout the reasoning process (Section 3.4). This phase further develops the model's ability to identify and integrate supporting facts within the context, ensuring alignment across multi-hop reasoning steps using only question-answer pairs without golden supporting facts.

Together, these two phases form a structured framework that integrates retrieval within reasoning, improving context loyalty, retrieval accuracy, and answer correctness in QA tasks.

3.2 Problem Formulation

We formally define our target problem as $(Q, C) \rightarrow A$, where Q is a user query, C is a long context containing sparse information relevant to answering Q, and A is the generated answer. This formulation specifically targets scenarios where the identification of key information within lengthy contexts and the utilization of sparse key information are the primary bottlenecks.

3.3 The Supervised Fine-Tuning Phase

The SFT phase establishes evidence integration by injecting retrieval tokens within structured reasoning steps. This phase uses an existing QA dataset with labeled supporting facts to ease the "cold-start" problem of the RL training phase, familiarizing the model with the targeted output format, the native re-

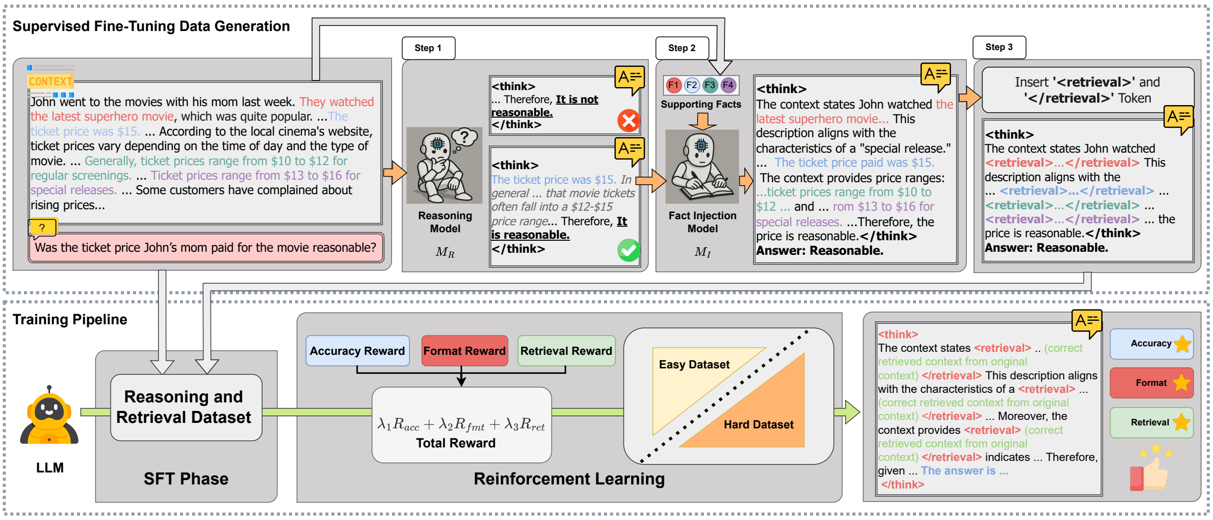

Figure 2: Illustration of the training data creation and two-phase training process of CARE. The upper part depicts the SFT data generation pipeline including fact injection and special tokens insertion within the reasoning content. The lower part shows the SFT training process and the reinforcement learning (RL) training with multiple rewards.

trieval process, and the chain-of-thought reasoning with retrieved facts as support.

Specifically, based on a given QA dataset with context and supporting facts from the context, we introduce a pipeline to generate reasoning chains interleaved with evidence. The data generation pipeline operates sequentially through three stages: reasoning step generation, evidence integration, and retrieval token insertion, which is illustrated in the upper part of Figure 2.

The data generation pipeline processes the input (C, Q), where C is the context and Q is the query, through three stages:

Reasoning Step Generation. The SFT dataset generation is based on an existing training dataset $\mathcal{D}_{\text{original}} = \{(Q_i, C_i, A_i, S_i)\}_{i=1}^{N_{\text{original}}}$, where the i-th instance contains a query Q_i, a context C_i, a ground truth answer A_i and a series of labeled supporting facts from C_i: $S_i = \{s_i^1, s_i^2, ..., s_i^{m_i}\}$. For each instance, a reasoning model M_R generates an initial reasoning response $R_{i,A}$ based on (C_i, Q_i):

$$R_{i,A} = M_R(C_i, Q_i) \tag{1}$$

For each output, we examine whether the generated answer matches the ground truth, and only responses that correctly answer the query Q are retained to ensure logical consistency. However, some correct responses might be derived from the internal knowledge of the model rather than from the input context. This disloyalty to the context potentially increases the risk of hallucinations. To better align the reasoning process with the input context, while R_A establishes a structured reasoning format, it may not align with the evidence of the context, necessitating further integration of the evidence in the next stage. For the i-th selected instance, we extract the reasoning process within the <THINK> </THINK> tokens to form the reasoning chain N_i.

Evidence Integration. To ensure that the reasoning aligns with the input context and to mitigate potential hallucinations from R_A, this stage integrates supporting facts S_i into the reasoning chain N_i. A fact injection model M_I then refines the initial reasoning R_A by incorporating these specific facts F. The output of the evidence integration process is conditioned on the query Q_i, the initial reasoning chain N_i, and the ground truth supporting facts S_i:

$$R_{i,I} = M_I(Q_i, N_i, S_i) \tag{2}$$

In this formulation, M_I focuses on weaving the supporting facts S_i into the existing reasoning structure of N_i to produce $R_{i,I}$. This step explicitly grounds the reasoning process in the supplied evidence, reducing the reliance on the model's internal knowledge when abundant context is given. After generation, we only keep the instances where $R_{i,I}$ contains all the supporting facts provided. For the i-th kept instance, we retrieve the output reasoning chain with evidence integration as E_i, which is more robustly supported by factual statements in the context than N_i.

Retrieval Token Insertion. Lastly, a pair of structural marker tokens is introduced to explicitly denote the supporting fact spans in the reasoning chain to assist the further training process. The newly added tokens <RETRIEVAL> </RETRIEVAL> is inserted around key evidence segments in R_I, forming the final structured response E_I^*.

Ultimately, we obtain our SFT training set $\mathcal{D}_{\text{SFT}} = \{(Q_i, C_i, A_i, E_i^*)\}_{i=1}^{N_{\text{SFT}}}$, establishing a consistent format for the subsequent RL phase.

3.4 Reinforcement Learning Phase

The reinforcement learning phase refines the self-retrieval mechanism established in the SFT phase by aligning the model's outputs with contextual evidence through Group Relative Policy Optimization (GRPO) (Shao et al., 2024). This phase leverages a curriculum learning strategy to gradually transition the model from basic to advanced reasoning tasks while applying retrieval-aware rewards to promote evidence consistency and logical coherence.

The detailed implementation of the reinforcement learning phase, including curriculum adjustment, reward computation, and policy updates, is described in Algorithm 1.

The GRPO Algorithm. GRPO optimizes the policy by evaluating multiple sampled outputs at the group level rather than individual actions. Given a query q, a set of outputs $\{o_1, \ldots, o_G\}$ is sampled from the old policy $\pi_{\theta_{\text{old}}}$. The objective function is defined as:

$$
\begin{aligned}
J_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \Bigg[\Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \\
\min\left[w_{i,t}\hat{A}_{i,t}, \text{clip}\left(r_{i,t}, 1-\epsilon, 1+\epsilon\right)\hat{A}_{i,t}\right] \\
-\beta D_{\text{KL}}\left(\pi_\theta \,\|\, \pi_{\text{ref}}\right)\Bigg]
\end{aligned} \tag{3}
$$

where the importance ratio $w_{i,t}$ is defined as:

$$
w_{i,t} = \frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q, o_{i,<t})} \tag{4}
$$

and the advantage function $\hat{A}_{i,t}$ is defined as:

$$
\hat{A}_{i,t} = \frac{r(q, o_i) - \text{mean}\left(\{r(q, o_i)\}_{i=1}^{G}\right)}{\text{std}\left(\{r(q, o_i)\}_{i=1}^{G}\right)} \tag{5}
$$

The clip function constrains the importance ratio to be within $[1-\epsilon, 1+\epsilon]$ to avoid overconfidence. The KL divergence $D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}})$ serves as a regularization term, preventing excessive divergence from the reference policy. Group-level evaluation in GRPO effectively promotes evidence alignment across multiple outputs, reinforcing the retrieval consistency within the reasoning process.

Algorithm 1 Curriculum RL with CARE Rewards.

Require: Datasets $\mathcal{D}_{\text{easy}}$, $\mathcal{D}_{\text{hard}}$, policy π_θ, reference policy π_{ref}, clip range ϵ, KL coefficient β, initial ratio $\alpha = 1.0$, total steps T
Ensure: Updated policy parameters θ
for each training step t **do**
  Sample query q with probability α from $\mathcal{D}_{\text{easy}}$ and $1-\alpha$ from $\mathcal{D}_{\text{hard}}$
  Sample outputs $\{o_i\}_{i=1}^{G}$ from $\pi_{\theta_{\text{old}}}(q)$
  for each output o_i **do**
    Extract retrieval spans S from o_i
    Compute rewards with Eq. 6
    for each token t in o_i **do**
      Compute importance ratio $r_{i,t} = \frac{\pi_\theta(o_{i,t})}{\pi_{\theta_{\text{old}}}(o_{i,t})}$
      Update objective with Eq. 3
    end for
  end for
  Apply KL penalty:
    $J_{\text{GRPO}} \leftarrow J_{\text{GRPO}} - \beta\sum_t \pi_\theta(o_t)\log\frac{\pi_\theta(o_t)}{\pi_{\text{ref}}(o_t)}$
  Update parameters $\theta \leftarrow \theta + \eta\nabla_\theta J_{\text{GRPO}}$
  Adjust curriculum ratio $\alpha \leftarrow \max(0, 1-\eta t/T)$
end for
return θ

Reward Design. To encourage the model to retrieve relevant information from the context and dynamically integrate them into the reasoning chain, we propose the retrieval reward for CARE training. More specifically, the retrieval reward encourages the model when it outputs the expected <RETRIEVAL> </RETRIEVAL> pair, and all text within these pairs exists in the context. Although a rather loose constraint, the retrieval reward allows the model to make better use of the context in reasoning without ground-truth retrieval data.

Furthermore, since we introduce a new pair of tokens in the reasoning process, we slightly changed the format reward proposed in DeepSeek-AI et al. (2025), which now pushes the model to reason with pairs <THINK> </THINK> and <RETRIEVAL> </RETRIEVAL>. Similarly, the accuracy reward quantifies the correctness of the generated response by calculating the token F1 score between the extracted generated answer and the ground truth answer for the QA tasks.

In general, the reward function in the RL phase is formulated as a weighted sum of three components, each aimed at a distinct aspect of the retrieval and alignment of the reasoning.

$$
R_{\text{total}} = \lambda_1 R_{\text{acc}} + \lambda_2 R_{\text{fmt}} + \lambda_3 R_{\text{ret}} \tag{6}
$$

The weighting coefficients $\lambda_1, \lambda_2, \lambda_3$ control the relative emphasis on factual accuracy, structural consistency, and context fidelity.

Curriculum Learning Strategy. QA datasets exhibit significant variation in context and answer

Model	Method	MFQA	HotpotQA	2WikiMQA	MuSiQue	Average
LLaMA-3.1 8B	Original	<u>45.57</u>	<u>54.64</u>	45.87	32.08	44.54
	ReSearch	/	/	/	/	/
	R1-Searcher	28.44	53.71	<u>67.10</u>	<u>41.41</u>	<u>47.67</u>
	CRAG	44.04	37.88	25.95	24.10	32.99
	CARE	**49.94**	**63.09**	**75.29**	**51.00**	**59.83**
Qwen2.5 7B	Original	46.94	<u>58.47</u>	46.96	30.78	45.79
	ReSearch	32.45	54.24	55.78	**47.61**	47.52
	R1-Searcher	28.36	55.43	<u>65.79</u>	<u>47.09</u>	<u>49.17</u>
	CRAG	<u>47.90</u>	43.97	33.00	28.44	38.33
	CARE	**48.11**	**63.45**	**70.11**	45.57	**56.81**
Qwen2.5 14B	Original	47.58	<u>61.94</u>	<u>59.05</u>	<u>37.99</u>	<u>51.64</u>
	ReSearch	/	/	/	/	/
	R1-Searcher	/	/	/	/	/
	CRAG	**50.89**	44.74	34.68	28.17	39.62
	CARE	<u>48.81</u>	**67.75**	**78.68**	**51.27**	**61.63**

Table 1: Evaluation on the real-world QA datasets. The results are grouped by the base LLM used. The best and second-best results for each base model and dataset are labeled in **bold** and <u>underline</u>, respectively. Slash (/) indicates that the model does not have an official checkpoint or support for this model.

lengths. To gradually adapt our model to diverse dataset characteristics other than the one used for SFT, we implement a curriculum learning strategy transitioning from short-context / short-answer QA to long-context / multihop long-answer QA. This structured progression mitigates catastrophic forgetting while enhancing retrieval capabilities in increasing complexity.

We train with two QA datasets: $\mathcal{D}_{\text{easy}} = \{(Q_i, C_i, A_i)\}_{i=1}^{N_{\text{easy}}}$ and $\mathcal{D}_{\text{hard}} = \{(Q_i, C_i, A_i)\}_{i=1}^{N_{\text{hard}}}$, where $\mathcal{D}_{\text{hard}}$ contains longer contexts, longer answers, and requires more complex reasoning than $\mathcal{D}_{\text{easy}}$. Training begins exclusively with $\mathcal{D}_{\text{easy}}$, then gradually incorporates instances from $\mathcal{D}_{\text{hard}}$.

At each training step t, we sample instances using a Bernoulli trial with a time-varying probability. The mixing ratio α_t decreases linearly according to $\alpha_t = \max(0, 1 - \eta \cdot \frac{t}{T})$, where η is a scaling factor that controls the speed of transition. The sampling probabilities are $p_{\text{easy}} = \alpha_t$ and $p_{\text{hard}} = 1 - \alpha_t$, ensuring that the model maintains short-context retrieval capabilities while learning to aggregate evidence in multiple paragraphs.

4 Experiment Settings

We evaluate our proposed CARE method through comprehensive experiments across multiple LLM families and sizes in two distinct QA categories: real-world long-context QA and counterfactual multihop QA.

4.1 Datasets, Benchmarks and Metrics

Training Datasets. We generate the SFT data mentioned in Section 3.3 based on the HotpotQA training set (Yang et al., 2018) owing to its annotations of supporting facts. During SFT data generation, DeepSeek-R1 (DeepSeek-AI et al., 2025) and DeepSeek-V3 (DeepSeek-AI et al., 2024) are used as the reasoning model M_R and the fact injection model M_I, respectively. The resulting SFT dataset contains 7,739 instances with the retrieval-augmented reasoning chain labeled. For RL training, we select DROP (Dua et al., 2019) as $\mathcal{D}_{\text{easy}}$ and MS MARCO (Nguyen et al., 2016) as $\mathcal{D}_{\text{hard}}$.

Evaluation Datasets. We evaluate in-context retrieval accuracy and whether learned retrieval-augmented reasoning improves answer quality using single-passage and multi-passage datasets from LongBench (Bai et al., 2024), including **MultiFieldQA-En** (Bai et al., 2024), **HotpotQA** (Yang et al., 2018), **2WikiMQA** (Ho et al., 2020), and **MuSiQue** (Trivedi et al., 2022). Following LongBench's protocol, we report F1 scores for all datasets.

Furthermore, to evaluate context fidelity when presented with information contradicting the model's parametric knowledge, we utilize **CofCA** (Wu et al., 2025), a benchmark containing modified counterfactual Wikipedia snippets.

Model	Method	CofCA
LLaMA-3.1 8B	Original	<u>48.14</u>
	R1-Searcher	45.25
	CARE	**61.83**
Qwen2.5 7B	Original	<u>58.38</u>
	ReSearch	47.32
	R1-Searcher	43.61
	CRAG	56.01
	CARE	**64.56**
Qwen2.5 14B	Original	<u>64.40</u>
	CRAG	51.99
	CARE	**67.75**

Table 2: Evaluation on the counterfactual QA task. The results are grouped by the base LLM used. The best and second-best results for each base model and dataset are labeled in **bold** and <u>underline</u>, respectively.

This directly tests whether our native retrieval-augmented reasoning improves adherence to provided context regardless of pre-trained biases. We report F1 performance consistent with the original CofCA evaluation metrics.

4.2 Models and Baselines

We compare CARE with a series of learned reasoning strategies and RAG methods based on three commonly used public LLMs: Qwen-2.5 Instruct 7B and 14B (Qwen et al., 2024), and LLaMA-3.1 8B (Grattafiori et al., 2024), which covers different model families and sizes.

Original Model. For each dataset, we test the performance of the original LLM with their corresponding default system prompt and chat template.

RL-Based Online Retrieval. Existing dynamic retrieval approaches typically leverage reinforcement learning to train models to autonomously conduct web searches rather than directly extract from the provided context. We compare our method against two recent RL-based online search methods: ReSearch (Chen et al., 2025) and R1-Searcher (Song et al., 2025), both of which enable models to strategically access external knowledge during reasoning. Note that in our model selection, ReSearch only provides a checkpoint for Qwen2.5 7B, and R1-Searcher only provides a checkpoint for LLaMA-3.1 8B and Qwen2.5 7B.

RAG Methods. We also compare with CRAG (Yan et al., 2024), a corrective RAG method that uses a lightweight evaluator to improve in-context retrieval with online search. Note that in our model selection, CRAG only provides a checkpoint for Qwen2.5 7B and 14B.

4.3 Experiment Settings

For all three models, the SFT training process follows LLaMA-Factory (Zheng et al., 2024)'s default LoRA SFT setting[2], and the RL training process follows verl (Sheng et al., 2024)'s default GRPO setting[3]. For the Curriculum RL training, we use the hyperparameters $\lambda_1 = 0.7$, $\lambda_2 = 0.1$, $\lambda_3 = 0.2$, and $\eta = 1$. All experiments are done with either $8\times$A800-SXM4-80GB or $8\times$H100 80GB. Detailed experiment settings are included in Appendix B.

5 Results and Analysis

5.1 Question-Answering Performance

Table 1 shows that CARE consistently outperforms baselines in all model sizes. With LLaMA-3.1 8B, our method achieves +15.29% average F1 improvement over the original model, with the strongest gains in multi-hop tasks (2WikiMQA +29.42%, MuSiQue +18.92%). Similar patterns appear with the Qwen2.5 models. Even when not highest (Qwen2.5 7B on MuSiQue and Qwen2.5 14B on MFQA), CARE remain competitive with the best baseline. These results demonstrate that native retrieval-augmented reasoning significantly enhances performance by effectively integrating in-context evidence during reasoning, especially for complex multi-hop questions. We also include a token cost analysis in Appendix A. While CARE generates longer outputs due to the reasoning chains, which is a common characteristic of GRPO-trained models, it eliminates the overhead of external API calls and database retrievals required by baseline methods.

5.2 Counterfactual QA Performance

In Table 2, we report the results on the CofCA counterfactural QA task. CARE consistently delivers the strongest performance, with significant gains on LLaMA-3.1 8B (+13.69%). In particular, traditional online search methods underperform compared to original models in this task, suggesting that external retrieval can be counterproductive when context contradicts parametric knowledge.

[2] https://github.com/hiyouga/LLaMA-Factory/blob/main/examples/train_lora/llama3_lora_sft.yaml

[3] https://github.com/volcengine/verl/blob/main/examples/grpo_trainer/run_qwen2-7b.sh

Settings	SFT	RL	Ret.	Cur.	MFQA	HotpotQA	2WikiMQA	MuSiQue	CofCA	Average
Baseline	✗	✗	✗	✗	<u>46.64</u>	58.47	46.96	30.78	58.38	48.25
SFT Only	✓	✗	✗	✗	42.24	47.08	61.51	33.82	59.21	48.77
No Ret.	✓	✓	✗	✗	37.66	62.59	<u>70.57</u>	43.85	57.26	54.39
No Cur.	✓	✓	✓	✗	38.33	**64.10**	**70.69**	**47.49**	<u>60.60</u>	<u>56.24</u>
CARE	✓	✓	✓	✓	**48.11**	<u>63.45</u>	70.11	<u>45.57</u>	**64.56**	**58.36**

Table 3: Ablation studies on the QA tasks based on Qwen2.5 7B. The best and second-best results for each base model and dataset are labeled in **bold** and <u>underline</u>, respectively. "Ret." stands for retrieval reward in Equation 6, and "Cur." stands for curriculum learning in Algorithm 1.

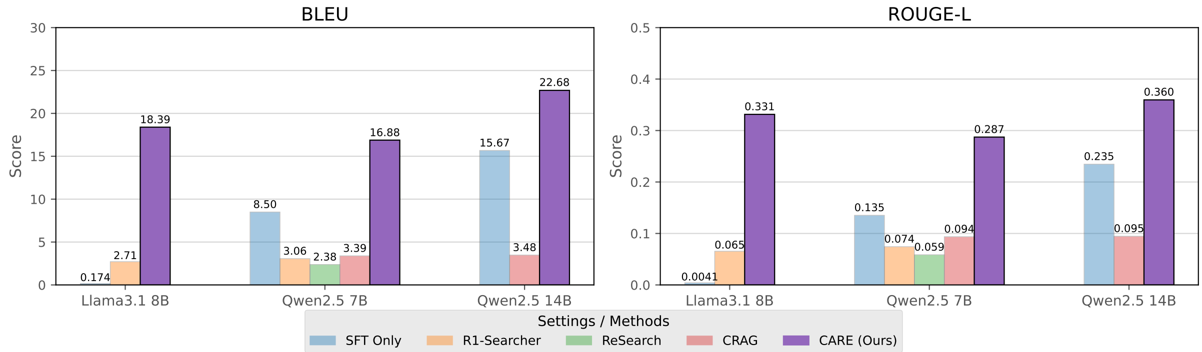

Figure 3: Comparison of model performance across different settings for BLEU and ROUGE-L metrics. Our proposed methods, CARE, demonstrate improved scores.

CARE demonstrates superior context fidelity by explicitly integrating natively extracted in-context evidence in the reasoning process, and can make even greater gains compared to the baselines when encountering unseen information in the context.

5.3 Ablation Studies

We provide results based on Qwen2.5 7B in Table 3. In this table, we include three additional settings: **(1) SFT only**, where the model is only trained with the first SFT phase without RL training; **(2) No retrieval reward**, where after the SFT phase, the model undergoes GRPO training with the same reasoning-encouraging reward used in DeepSeek-R1 (DeepSeek-AI et al., 2025); and **(3) No curriculum learning**, where the RL training phase uses only $\mathcal{D}_{\text{easy}}$.

SFT alone only offers marginal benefits, while adding RL training substantially improves performance, highlighting the importance of reinforcement learning for QA reasoning. Both methods with native in-context reasoning ("No Cur." and CARE) consistently outperform the vanilla R1-like GRPO approach ("No Ret."), showing that retrieval-augmented reasoning enhances performance by grounding reasoning in contextual evidence. While "No Cur." performs well on multi-hop datasets, curriculum learning provides better balance across diverse types of QA, particularly improving performance on long-form answering (MFQA) and counterfactual scenarios (CofCA). This shows that curriculum learning successfully adapts the model to various types of question while maintaining strong performance on complex reasoning tasks, all without requiring additional labeled data beyond the initial SFT phase.

5.4 Evidence Retrieval Evaluation

In this section, we evaluate CARE's ability to accurately retrieve and incorporate supporting evidence for question-answering. Due to the lack of ground-truth supporting fact annotations in standard QA datasets, we focus our evaluation on the LongBench HotpotQA benchmark. For this analysis, we align each instance in LongBench's HotpotQA test set with its corresponding entry in the original HotpotQA dataset, using the original supporting fact annotations as ground truth for evaluation. We report SacreBLEU (Post, 2018) and ROUGE-L F1 (Lin, 2004). Figure 3 presents our comparative results in different model configurations. In all settings, CARE consistently achieves the highest BLEU and ROUGE-L scores. We observe that performance scales with model size across all methods, with Qwen2.5 14B showing the strongest results. However, the relative improvement from CARE remains consistent regardless of the scale and family of the model, suggesting that our approach effectively enhances the context fidelity regardless of the underlying model architecture.

6 Conclusion

We introduce CARE, a native retrieval-augmented reasoning framework that improves context fidelity in LLM by teaching models to dynamically identify and integrate evidence within their reasoning process. This approach improves how LLMs interact with context, while requiring limited labeled evidence. Experiments on multiple general and counterfactual QA benchmarks demonstrated that CARE consistently outperforms existing approaches, including the vanilla SFT method and traditional RAG methods in both answer generation and evidence extraction. This work represents an important step toward more reliable AI systems that make better use of available context without requiring expensive retrieval infrastructure.

Limitations

Although CARE shows significant improvements in context fidelity and question answering performance, several important limitations should be acknowledged. First, the native retrieval-augmented reasoning mechanism, while effective for in-context information, cannot access external knowledge beyond the provided context. For scenarios requiring information not present in the input, our approach would need to be combined with external retrieval systems like RAG, potentially complicating the overall architecture.

Second, while we evaluate comprehensively across multiple QA benchmarks, the evaluation primarily focuses on multi-hop general-domain reasoning questions. The effectiveness of CARE for more abstract reasoning, numerical computation, creative generation tasks, or domain-specific tasks remains to be thoroughly investigated.

Finally, although our method improves context fidelity, it does not completely eliminate the possibility of hallucinations, especially when the input contains ambiguous or contradictory information. Future work should address these challenges while expanding the approach to a broader range of language understanding and generation tasks.

Ethical Considerations

Our research improves context fidelity in language models, potentially reducing hallucinations in critical applications like education and healthcare. The supervised fine-tuning dataset we developed, built upon HotpotQA (which follows the CC BY-SA 4.0 license), will be shared under the same CC BY-SA 4.0 license to promote transparency and reproducibility. Although our method improves fidelity to the provided context, it cannot guarantee complete factual accuracy, especially when the input itself contains inaccuracies or contradictions. We acknowledge both the environmental impact of the computational resources used in training and the possibility that models may inherit biases from training data despite improved context fidelity. Researchers who implement our approach should perform appropriate fairness evaluations before deployment in sensitive applications.

Acknowledgement

This research was enabled in part by the support provided by Calcul Québec[4] and the Digital Research Alliance of Canada[5].

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *ICLR*. OpenReview.net.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024. LongBench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3119–3137, Bangkok, Thailand. Association for Computational Linguistics.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and 1 others. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17682–17690.

Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. Booookscore: A systematic exploration of book-length summarization in the era of LLMs. In *The Twelfth International Conference on Learning Representations*.

[4] https://www.calculquebec.ca/.

[5] https://alliancecan.ca/.

Mingyang Chen, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z Pan, Wen Zhang, Huajun Chen, Fan Yang, and 1 others. 2025. Research: Learning to reason with search for llms via reinforcement learning. *arXiv preprint arXiv:2503.19470*.

Yunmo Chen, Tongfei Chen, Harsh Jhamtani, Patrick Xia, Richard Shin, Jason Eisner, and Benjamin Van Durme. 2024. Learning to retrieve iteratively for in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7156–7168.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv: 2501.12948*.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv: 2412.19437*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378, Minneapolis, Minnesota. Association for Computational Linguistics.

Mehrdad Farahani and Richard Johansson. 2024. Deciphering the interplay of parametric and non-parametric memory in retrieval-augmented language models. *arXiv preprint arXiv:2410.05162*.

Weizhi Fei, Xueyan Niu, Guoqing Xie, Yanhua Zhang, Bo Bai, Lei Deng, and Wei Han. 2024. Retrieval meets reasoning: Dynamic in-context editing for long-text understanding. *arXiv preprint arXiv:2406.12331*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv: 2407.21783*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval augmented language model pre-training. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Sheryl Hsu, Omar Khattab, Chelsea Finn, and Archit Sharma. 2024. Grounding by trying: Llms with reinforcement learning-enhanced retrieval. *arXiv preprint arXiv:2410.23214*.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Xiangkun Hu, Dongyu Ru, Lin Qiu, Qipeng Guo, Tianhang Zhang, Yang Xu, Yun Luo, Pengfei Liu, Yue Zhang, and Zheng Zhang. 2024. Refchecker: Reference-based fine-grained hallucination checker and benchmark for large language models. *arXiv preprint arXiv: 2405.14486*.

Peter Humphreys, Arthur Guez, Olivier Tieleman, Laurent Sifre, Théophane Weber, and Timothy Lillicrap. 2022. Large-scale retrieval for reinforcement learning. *Advances in Neural Information Processing Systems*, 35:20092–20104.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Mandar Kulkarni, Praveen Tangarajan, Kyung Kim, and Anusua Trivedi. 2024. Reinforcement learning for optimizing rag for domain chatbots. *arXiv preprint arXiv:2401.06800*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *NeurIPS*.

Huayang Li, Pat Verga, Priyanka Sen, Bowen Yang, Vijay Viswanathan, Patrick Lewis, Taro Watanabe, and Yixuan Su. 2024. Alr^2: A retrieve-then-reason framework for long-context question answering. *arXiv preprint arXiv: 2410.03227*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Bang Liu, Xinfeng Li, Jiayi Zhang, Jinlin Wang, Tanjin He, Sirui Hong, Hongzhang Liu, Shaokun Zhang, Kaitao Song, Kunlun Zhu, Yuheng Cheng, Suyuchen Wang, Xiaoqiang Wang, Yuyu Luo, Haibo Jin, Peiyan Zhang, Ollie Liu, Jiaqi Chen, Huan Zhang, and 28 others. 2025a. Advances and challenges in foundation agents: From brain-inspired intelligence to evolutionary, collaborative, and safe systems. *arXiv preprint arXiv: 2504.01990*.

Siyi Liu, Kishaloy Halder, Zheng Qi, Wei Xiao, Nikolaos Pappas, Phu Mon Htut, Neha Anna John, Yassine Benajiba, and Dan Roth. 2025b. Towards long context hallucination detection. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 7827–7835, Albuquerque, New Mexico. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey. *arXiv preprint arXiv: 2402.06196*.

Minh Nguyen, Toan Quoc Nguyen, Kishan KC, Zeyu Zhang, and Thuy Vu. 2024. Reinforcement learning from answer reranking feedback for retrieval-augmented answer generation. In *Proceedings of INTERSPEECH*.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv: 2412.15115*.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2019. Zero: Memory optimizations toward training trillion parameter models. *arXiv preprint arXiv: 1910.02054*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *European Conference on Computer Systems*.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. 2025. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *arXiv preprint arXiv:2503.05592*.

Wrick Talukdar and Anjanava Biswas. 2024. Improving large language model (llm) fidelity through context-aware grounding: A systematic approach to reliability and veracity. *arXiv preprint arXiv: 2408.04023*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. MuSiQue: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Chien-Hung Tu, Hsien-Jung Hsu, and Shih-Wen Chen. 2024. Reinforcement learning for optimized information retrieval in llama.

Alexandre Variengien and Eric Winsor. 2023. Look before you leap: A universal emergent decomposition of retrieval tasks in language models. *CoRR*, abs/2312.10091.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry W. Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc V. Le, and Thang Luong. 2024. Freshllms: Refreshing large language models with search engine augmentation. In *ACL (Findings)*, pages 13697–13720. Association for Computational Linguistics.

Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query expansion with large language models. In *EMNLP*, pages 9414–9423. Association for Computational Linguistics.

Liang Wang, Nan Yang, and Furu Wei. 2024. Learning to retrieve in-context examples for large language models. In *EACL (1)*, pages 1752–1767. Association for Computational Linguistics.

Xinyu Wang, Jijun Chi, Zhenghan Tai, Tung Sum Thomas Kwok, Muzhi Li, Zhuhong Li, Hailin He, Yuchen Hua, Peng Lu, Suyuchen Wang, Yihong Wu, Jerry Huang, Jingrui Tian, and Ling Zhou. 2025. Finsage: A multi-aspect rag system for financial filings question answering. *Preprint*, arXiv:2504.14493.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv: 1910.03771*.

Jian Wu, Linyi Yang, Zhen Wang, Manabu Okumura, and Yue Zhang. 2025. CofCA: A STEP-WISE counterfactual multi-hop QA benchmark. In *The Thirteenth International Conference on Learning Representations*.

Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. 2024. Benchmarking retrieval-augmented generation for medicine. In *ACL (Findings)*, pages 6233–6251. Association for Computational Linguistics.

Peng Xu, Wei Ping, Xianchao Wu, Lawrence McAfee, Chen Zhu, Zihan Liu, Sandeep Subramanian, Evelina Bakhturina, Mohammad Shoeybi, and Bryan Catanzaro. 2023. Retrieval meets long context large language models. In *The Twelfth International Conference on Learning Representations*.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. In *ICLR*. OpenReview.net.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. 2023. Pytorch FSDP: experiences on scaling fully sharded data parallel. *Proc. VLDB Endow.*, 16(12):3848–3860.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410, Bangkok, Thailand. Association for Computational Linguistics.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and 1 others. 2022. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

Yujia Zhou, Zhicheng Dou, and Ji-Rong Wen. 2023. Enhancing generative retrieval with reinforcement learning from relevance feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12481–12490.

Shengyao Zhuang, Xueguang Ma, Bevan Koopman, Jimmy Lin, and Guido Zuccon. 2025. Rank-r1: Enhancing reasoning in llm-based document rerankers via reinforcement learning. *arXiv preprint arXiv:2503.06034*.

A Token Cost Efficiency Analysis

In this section, we provide a detailed token cost analysis for the real-world QA experiments (Section 5.1) in Table 4. While CARE generates longer outputs due to the reasoning chains, which is a common characteristic of GRPO-trained models, it eliminates the overhead of external API calls and database retrievals required by baseline methods.

Model	Method	MFQA	HotpotQA	2Wiki	MuSiQue
LLaMA 8B	Original	19.5	8.5	7.5	7.4
	R1-S.	296+2012	278+2058	293+2125	313+2436
	CARE	564	656	608	848
Qwen 7B	Original	24.8	6.0	6.1	9.3
	CRAG	24+212	7+411	9+201	8+470
	ReSearch	276+2054	275+2271	308+2814	290+2492
	CARE	566	633	560	942

Table 4: Average output tokens per query on each real-world QA dataset. Numbers in format x+y indicate model output + retrieval overhead. R1-S. stands for R1-Searcher.

B Experiment Details

B.1 Implementation Details

All models are implemented based on pre-trained checkpoints provided by the Huggingface Transformers library (Wolf et al., 2019). We use LLaMA-Factory (Zheng et al., 2024) for the SFT phase. In this phase, we train each model on our curated SFT dataset for 3 epochs with the AdamW optimizer (Loshchilov and Hutter, 2019). The training progress adopts a warmup cosine scheduler with a maximum learning rate 0.0001 and a warmup ratio of 0.1. The effective batch size is 64. LoRA (Hu et al., 2022) is applied with $r = 8$ and $\alpha = 16$. The training process uses the ZeRO-2 optimizer (Rajbhandari et al., 2019). For the RL phase, we adopt the verl framework (Sheng et al., 2024) for GRPO training. We used a training batch size of 1024. The Adam optimizer was employed with a learning rate of 1e-6. For policy optimization, we use GRPO as the advantage estimator and incorporated KL divergence regularization with a coefficient of 0.001 using the low-variance KL estimator. We set the mini-batch size to 256. The model was trained for 350 steps with 5 response samples per prompt. For distributed training, we deployed Fully Sharded Data Parallel (FSDP) (Zhao et al., 2023) across 8 GPUs on a single node with tensor parallelism of size 2. All experiments are done with either 8×A800-SXM4-80GB or 8×H100 80GB.

C System Prompts

We provide the system prompts used in the dataset creation process and the CARE in the following.

Prompt used for M_R's generation of reasoning chains for SFT data creation.

You're an expert reader. Your goal is to read a context to answer a question. Note that during your thinking process, before you make *any reasoning step that requires retrieving information from the context*, summarize what information you would need to complete this reasoning step, such as "I need to know X for this" or similar phrases before you reason about the context. This will help you to be more systematic in your reasoning process. Put your final answer as a minimum phrase or word at the end after "Answer:".
Context: {context}
Question: {question}

Prompt used for M_I's evidence integration for SFT data creation.

I'll provide you with a question, a reasoning process to solve this question, and several evidence sentences. Insert *all* evidence sentences into the reasoning process at appropriate locations and give me the updated reasoning process. Each evidence sentence usually should be placed just before any conclusions or deductions that depend on it. The evidence sentences may need to be distributed throughout different parts of the reasoning and may appear more than once. *Do not modify any evidence sentences* - insert them exactly as provided. Return only the completed reasoning process without explanations or additional text scaffolds.
Question: {question}
Reasoning process: {reasoning_content}
Evidence sentences (One sentence per line): {evidence_sentence_string}
The rewritten reasoning process:

System prompt for CARE. The actual system prompt for each model prepends the corresponding model's original system prompt before this prompt.

You FIRST think about the reasoning process as an internal monologue and then provide the final answer. The reasoning process MUST BE enclosed within <think> </think> tags. WITHIN the thinking process, make reference to the relevant texts in the prompt that provide critical information to move the reasoning process forward. The referenced texts MUST BE enclosed within <retrieval> </retrieval> tags, and MUST BE placed within the reasoning process only. The final answer MUST BE put at the end of the response after "Answer:".

D Case Study

We select two samples from LongBench's HotpotQA dataset and provide the complete question, answer, supporting facts, prediction, and whether the generated answer is correct.

Question	Answer	Supporting Facts	Prediction	Correct
The song that came out ahead of The Wanted's “Lightning” on the UK Singles Chart was featured on which album?	Talk That Talk	["“We Found Love” is a song recorded by Barbadian singer Rihanna from her sixth studio album, “Talk That Talk” (2011).", "It was written and produced by, and features Scottish DJ Calvin Harris; the song was also featured on his third album “18 Months” (2012).", "“We Found Love” premiered on September 22, 2011, on the Capital FM radio station in the United Kingdom, and was made available to download on the same day as the lead single from “Talk That Talk”.", "“We Found Love” is an uptempo electro house song, with elements of techno and Europop.", "The song's lyrics speak of a couple who “found love in a hopeless place”."]	<think>\nOkay, let's try to figure out the answer to this question. The question is asking which album featured the song that came out ahead of The Wanted's “Lightning” on the UK Singles Chart.\n\nFirst, I need to find out the details about The Wanted's “Lightning” from the passages provided. Looking at the passage for The Wanted's discography, it mentions that their third single from the second album “Battleground” was “Lightning,” released in October 2011. <retrieval>The song debuted at number two on the UK Singles Chart, behind Rihanna's “We Found Love”.</retrieval> So, that means the song that was number one before “Lightning” was “We Found Love” by Rihanna.\n\nNow, the question is about the album that features “We Found Love.” I need to check where “We Found Love” is mentioned. In the passages, there's a section about Rihanna's “We Found Love” from her album “Talk That Talk (2011.” The album is listed as her sixth studio album. The song itself is part of that album. <retrieval>“We Found Love” is a song by Barbadian singer Rihanna from her sixth studio album, Talk That Talk (2011.</retrieval> So the answer should be “Talk That Talk.”\n\nWait, but let me confirm. The relevant information from the third passage (The Rihanna section for “We Found Love” clearly states that it's from “Talk That Talk (2011),” and that's her sixth studio album. The song's debut was with Rihanna's album. Therefore, the answer is “Talk That Talk.”\n</think>\n\nAnswer: Talk That Talk	Yes
Yau Ma Tei North is a district of a city with how many citizens?	7.2 million	["Hong Kong, officially the Hong Kong Special Administrative Region of the People's Republic of China, is an autonomous territory south to Mainland China and east to Macao in East Asia.", " With around 7.2 million Hong Kongers of various nationalities in a territory of 1,104 km, Hong Kong is the world's fourth most densely populated country or territory."]	<think>\nOkay, let's tackle this question. The question is asking how many citizens Yau Ma Tei North is a district of. First, I need to figure out which city Yau Ma Tei North is a district of. From the context, the term “Yau Ma Tei North” is mentioned in several passages. Let me check.\n\nLooking at the passages, the one titled “Yau Ma Tei North (constituency)” states that Yau Ma Tei North is one of the 19 constituencies in the Yau Tsim Mong District of Hong Kong. \n<retrieval>Yau Ma Tei North is one of the 19 constituencies in the Yau Tsim Mong District of Hong Kong which was first created in 1982 and recreated in 2015.</retrieval>\nThat's in the context of Hong Kong's administrative divisions. The next step would be to find out the population of Hong Kong. \n\nI recall that there's a detailed section in the “Hong Kong” passages about the population. Let me check that part. The Hong Kong entry mentions that the Census and Statistics Department estimated Hong Kong's population at 7,413,070 in 2021.\n<retrieval>The Census and Statistics Department estimated Hong Kong's population at 7,413,070 in 2021. The overwhelming majority (91.6%) is Han Chinese, most of whom are Taishanese, Teochew, Hakka, and other Cantonese peoples.</retrieval>\n\nWait, but the question is about Yau Ma Tei North, which is a district within Hong Kong. The population of Hong Kong as a whole is 7.4 million, but the key point here is whether Yau Ma Tei North itself has a population figure. The Yau Ma Tei North constituency's entry says it covers Yau Ma Tei and has an estimated population of 12,817. \n<retrieval>The constituency loosely covers Yau Ma Tei with the estimated population of 12,817.</retrieval>\nSo that's 12,817 people. But the question is phrased as “Yau Ma Tei North is a district of a city with how many citizens?” So the target is the parent city of Yau Ma Tei North. Since Yau Ma Tei North is a district in Hong Kong, Hong Kong is the city. Therefore, the population of Hong Kong is the answer, which is 7,413,070 as per the latest data provided.\n</think>\n\nTo determine the population of the city (Hong Kong) containing Yau Ma Tei North, I need to know the administrative level of Yau Ma Tei North. The “Yau Ma Tei North (constituency)” section states it belongs to Yau Tsim Mong District. Next, the “Hong Kong” passages mention the district is part of the Yau Tsim Mong District and its population. The main Hong Kong entry (under “Demographics”) gives the 2021 population as 7,413,070. However, Yau Ma Tei North is a specific constituency (the 19th out of 19) within Yau Tsim Mong District, with an estimated population of 12,817. The question likely refers to the broader city of Hong Kong's population, not the smaller constituency. \nAnswer: 7413070	No

Table 5: Case study examples from LongBench's HotpotQA test set showing CARE's reasoning process. The table presents the original questions, ground-truth answers and supporting facts, along with the complete model predictions including the reasoning chains within <think> tags and integrated retrieval spans within <retrieval> tags. The "Correct" column indicates whether the model's final answer matches the ground truth using F1 score evaluation.