

Learning to Extract Rational Evidence via Reinforcement Learning for Retrieval-Augmented Generation

Xinping Zhao^{1,2}, Shouzheng Huang¹, Yan Zhong³, Xinshuo Hu,
Meishan Zhang^{1✉}, Baotian Hu^{1,2}, Min Zhang^{1,2}

¹Harbin Institute of Technology (Shenzhen), ²Shenzhen Loop Area Institute, ³Peking University
{zhaoxinping, 210110129}@stu.hit.edu.cn,
zhongyan@stu.pku.edu.cn, yanshek.woo@gmail.com,
mason.zms@gmail.com, {hubaotian, zhangmin2021}@hit.edu.cn

Abstract

Retrieval-Augmented Generation (RAG) effectively improves the accuracy of Large Language Models (LLMs). However, retrieval noises significantly undermine the quality of LLMs' generation, necessitating the development of denoising mechanisms. Previous works extract evidence straightforwardly without deep thinking, which may risk filtering out key clues and struggle with generalization. To this end, we propose EviOmni, which learns to extract rational evidence via reasoning first and then extracting. Specifically, EviOmni integrates evidence reasoning and evidence extraction into one unified trajectory, followed by knowledge token masking to avoid information leakage, optimized via on-policy reinforcement learning with verifiable rewards in terms of *answer*, *length*, and *format*. Extensive experiments on five benchmark datasets show the superiority of EviOmni, which provides compact and high-quality evidence, enhances the accuracy of downstream tasks, and supports both traditional and agentic RAG systems¹.

1 Introduction

Retrieval-Augmented Generation (RAG) prevails in Large Language Models (LLMs). It has demonstrated strong effectiveness across a wide array of knowledge-intensive tasks (Lewis et al., 2020; Wu et al., 2022), such as open-domain question answering (QA) (Shi et al., 2024; Trivedi et al., 2023), fact-checking (Du et al., 2023; Zhao et al., 2024b), and dialog generation (Izacard et al., 2023; Thoppilan et al., 2022), to produce faithful, reliable, and accurate outputs. Ideally, LLMs should be grounded in purely relevant content to generate accurate output and facilitate inference speed. However, due

[✉]Corresponding author.

¹Models are available at https://huggingface.co/HIT-TMG/EviOmni-nq_train-1.5B and https://huggingface.co/HIT-TMG/EviOmni-nq_train-7B. Code is available at <https://github.com/HITsz-TMG/EviOmni>.

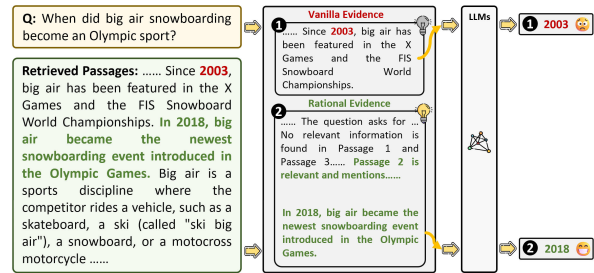


Figure 1: Motivating example, where key clues are marked in green: ❶ The key clue is omitted, leading to an incorrect answer; ❷ Guided by evidence reasoning, the key clue is extracted, leading to a correct answer.

to imperfect retrieval systems and noisy retrieval corpus (Wang et al., 2023; Wei et al., 2024; Zhao et al., 2024a), retrieval passages often contain many irrelevant or noisy snippets, distracting LLMs' attention and degrading generation quality. As such, it is necessary to extract evidence and filter out noise for RAG to achieve superior performance.

Recently, several studies have attempted to address this issue. They can be broadly divided into two categories: (1) Reranking more relevant passages to the top of the retrieval list (Hwang et al., 2024b; Mortaheb et al., 2025; Mao et al., 2021); and (2) Summarizing retrieval contents into a coherent and relevant narrative (Wang et al., 2023; Zhao et al., 2024a; Zhu et al., 2024; Xu et al., 2024). The former heavily relies on the performance of the reranker itself and may disrupt context after reranking. It usually performs worse than the latter due to a lack of contextual understanding. The latter aims to train LLMs as filtering models via supervised fine-tuning (SFT) (Wang et al., 2023) or preference optimization (Zhao et al., 2024a). However, these methods typically rely on heuristically constructed training data, e.g., String Inclusion that measures whether the golden answer is included in the passage, and Lexical Overlap that calculates the unigram overlap between the answer and passage.

Despite their effectiveness, existing methods rerank or summarize retrieval contents straightforwardly without deep thinking, which may risk filtering out key clues due to insufficient contextual understanding of retrieved contents. Figure 1 presents a comparison between vanilla evidence and rational evidence for QA: ❶ The vanilla evidence omits the key clue and leads to an incorrect answer; and ❷ In contrast, evidence reasoning guides the subsequent evidence extraction, *i.e.*, “*Passage 2 is relevant and mentions.....*”, so the rational evidence preserves the key clue, leading to a correct answer. This finding motivates us to develop rational evidence extraction for RAG, which first performs deep reasoning and then consciously extracts evidence.

In this work, we propose EviOmni, a framework that learns to extract rational evidence for RAG via reinforcement learning with verifiable rewards. Specifically, EviOmni integrates evidence reasoning (enclosed within `<reason>` and `</reason>`) and extraction (enclosed within `<extract>` and `</extract>`) into one response enabling end-to-end training. The model generates multiple responses to collect informative trajectories containing both positive and negative experiences. Subsequently, EviOmni applies knowledge token masking on each response to avoid information leakage when generating rationale-based and evidence-based answers (enclosed within `<answer>` and `</answer>`) to assess their quality, respectively. Finally, we design three types of rule-based verifiable reward functions, *i.e.*, *answer*, *length*, and *format*, to guide model optimization via Group Relative Policy Optimization (GRPO) (Shao et al., 2024) towards the desired properties. As a result, EviOmni can reason over retrieval contents and consolidate key clues into a concise yet informative narrative, thereby improving LLMs’ generation accuracy. The main contributions of this work can be summarized as:

- We unveil and empirically validate the major issue hindering evidence extraction for RAG, *i.e.*, insufficient contextual understanding of retrieval contents, which risks filtering out key clues.
- We propose a novel rational evidence extraction learning framework, EviOmni, which first reasons and then extracts, optimized through reinforcement learning with verifiable rewards.
- Extensive experiments on five benchmarks show that EviOmni is effective for both traditional and

agentic RAG, with superior accuracy, generalization, efficiency, and robustness to retrieval noise.

2 Preliminaries

2.1 Problem Statement

In RAG (Lewis et al., 2020), LLMs are given an query q and top- k retrieved passages $P = \{p_1, p_2, \dots, p_k\}$, to generate an output o that approximates the golden answer a . In the traditional RAG, retrieved passages P are directly fed into LLMs. However, these passages may contain irrelevant or noisy contents (Wang et al., 2023; Zhao et al., 2024a), substantially degrading LLMs’ generation accuracy and efficiency. To this end, evidence extraction is introduced to condense P into a concise, query-relevant context e , thereby improving generation quality and speed. The paradigm of “RAG with evidence extraction” can be formulated as:

$$e = \mathcal{M}_{\mathcal{E}}(\cdot|q, P), \quad o = \mathcal{M}_{\mathcal{G}}(\cdot|q, e), \quad (1)$$

where $\mathcal{M}_{\mathcal{E}}(\cdot)$ denotes the evidence extractor; e is the extracted evidence; $\mathcal{M}_{\mathcal{G}}(\cdot)$ denotes the answer generator. Albeit effective, the vanilla paradigm may risk filtering out key clues due to insufficient contextual understanding of retrieved passages. To address this issue, we propose a novel paradigm of “RAG with rational evidence extraction”:

$$\begin{aligned} e &= \mathcal{M}_{\mathcal{E}}(\cdot|q, P, r) \cdot \mathcal{M}_{\mathcal{E}}(r|q, P), \\ o &= \mathcal{M}_{\mathcal{G}}(\cdot|q, e), \end{aligned} \quad (2)$$

where r denotes the rationale, which explicitly and thoroughly identifies any clues lying in retrieved passages (P) to guide rational evidence extraction (e). We train the model in an on-policy manner, meaning that $\mathcal{M}_{\mathcal{E}}(\cdot)$ and $\mathcal{M}_{\mathcal{G}}(\cdot)$ are identical during training. The overall objective is to learn a policy $\pi_{\theta} : e \sim \mathcal{M}_{\mathcal{E}}(\cdot|q, P, r) \cdot \mathcal{M}_{\mathcal{E}}(r|q, P)$, to extract rational evidence e that maximizes the expected utility $\mathcal{U}(a, o)$ over the data distribution \mathcal{D} :

$$\begin{aligned} \max_{\pi_{\theta}} \mathbb{E}_{a \sim \mathcal{M}_{\mathcal{G}}(\cdot|q, e), e \sim \pi_{\theta}(q, P), q \sim \mathcal{D}} [\mathcal{U}(a, o)], \\ \text{s.t. } L_e \ll L_P \end{aligned} \quad (3)$$

where $\mathcal{U}(\cdot)$ is a utility function that measures the quality of outputs conditioned on a ; L_e and L_P are the length of evidence and passages, respectively.

2.2 Empirical Study

Experimental Setup. We conduct an empirical study to verify the assumption that evidence extractors can retain as many key clues as possible via

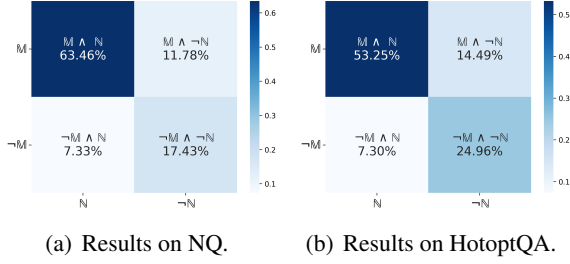


Figure 2: The results *w.r.t.* AR on the in-distribution NQ as well as out-of-distribution HotoptQA datasets.

Dataset	Vanilla Evidence	Rational Evidence	Rationale
NQ	70.79%	75.24%	77.30%
HotoptQA	60.55%	67.74%	71.48%

Table 1: AR results of Vanilla Evidence, Rational Evidence, and Rationale on NQ and HotpotQA datasets.

reasoning first and then extracting. We construct a synthetic instruction dataset by utilizing DeepSeek-R1 (DeepSeek-AI et al., 2025), where we sample 1K instances from the training set of Natural Question (NQ) (Kwiatkowski et al., 2019). The output of each instance in the dataset contains three parts: (1) `<reason>` evidence reasoning `</reason>`; (2) `<extract>` evidence extraction `</extract>`; and (3) `<answer>` final answer `</answer>`. We filter out instances with **incorrect** final answers, resulting in about 620 instances. Then, we create two variants of the dataset, where the output of the first one consists of “`<reason>` ... `</reason><extract>` ... `</extract>`”; that of the second one only consists of “`<extract>` ... `</extract>`”. Finally, we fine-tune Qwen2.5-1.5B-Instruct on these two datasets via Low-Rank Adaptation (LoRA), respectively. The LoRA Rank is set to 16; the number of epochs is set to 3. We tested on in-distribution NQ and out-of-distribution (OOD) HotoptQA (Yang et al., 2018) datasets, excluding instances from the test set, where the retrieved passages did not contain the golden answer. For a fair comparison, the maximum length of the extracted evidence is set to 100.

Analysis and Discussion. Table 1 and Figure 2 present the results *w.r.t.* Answer Recall (AR) (Zhao et al., 2025; Jiang et al., 2024), measuring the recall of the golden answer string in the evidence. The evidence extracted by the model trained on the first and second datasets is denoted as “rational evidence” (marked as \mathbb{M}) and “vanilla evidence” (marked as \mathbb{N}), respectively. Here, “rationale” refers to the evidence reasoning. Taking

$\mathbb{M} \wedge \neg \mathbb{N}$ as an example, it means the answer string is recalled by the rational evidence but not by the vanilla one. The results show: (1) Performance with rational evidence consistently outperforms that with vanilla one, whether on in-distribution or OOD datasets; (2) Performance with rational evidence is slightly lower than that with rationale, indicating the necessity of reasoning first and better optimization needed to bridge this gap; and (3) The percentage of $\mathbb{M} \wedge \neg \mathbb{N}$ is considerably higher than that of $\neg \mathbb{M} \wedge \mathbb{N}$, e.g., 14.49% vs. 7.30% on HotpotQA, showing the rational evidence’s superiority.

Case Study. Detailed case studies on traditional and agentic RAG are provided in Appendices H and I. Briefly, studies on traditional RAG show that the limitations of vanilla evidence versus rational one lie in the “**3I**” issues: (1) **Incompleteness**: lack of key clues; (2) **Irrelevance**: irrelevant information; and (3) **Inaccuracy**: incorrect information. In contrast, rational evidence benefits agentic RAG via the “**4R**” properties: (1) **Reduction** of redundant context; (2) **Refusal** when evidence is insufficient; (3) **Robustness** to noise; and (4) **Right timing**, which early stops once evidence is enough. Given the above **qualitative** and **quantitative** analyses, it is necessary and valuable to learn to extract rational evidence for improving overall RAG performance.

3 Methodology

The overall framework of EviOmni is illustrated in Figure 3 including rational evidence acquisition (§3.1), modeling verifiable reward (§3.2), and online policy optimization (§3.3). Algorithms 1 and 2 illustrate how EviOmni enhances the workflows of traditional and agentic RAG, respectively.

3.1 Obtaining Rational Evidence

Rational Evidence Extraction. As stated in §1, vanilla evidence risks filtering out key clues (Wang et al., 2023; Zhao et al., 2024a). An empirical study in §2.2 further verifies our assumption, where rational evidence performs better than vanilla one. To this end, we aim to optimize the evidence extractor $\mathcal{M}_{\mathcal{E}}(\cdot)$ to learn to extract rational evidence for RAG, formulated as: $e \sim \mathcal{M}_{\mathcal{E}}(\cdot|q, P, r) \cdot \mathcal{M}_{\mathcal{E}}(r|q, P)$. Specifically, we feed the query q and its corresponding retrieved passages P into the evidence extractor $\mathcal{M}_{\mathcal{E}}(\cdot)$ and instruct the evidence extractor to reason first and then extract. Evidence reasoning and evidence extraction are enclosed within the special “reason” and “extract” tags, re-

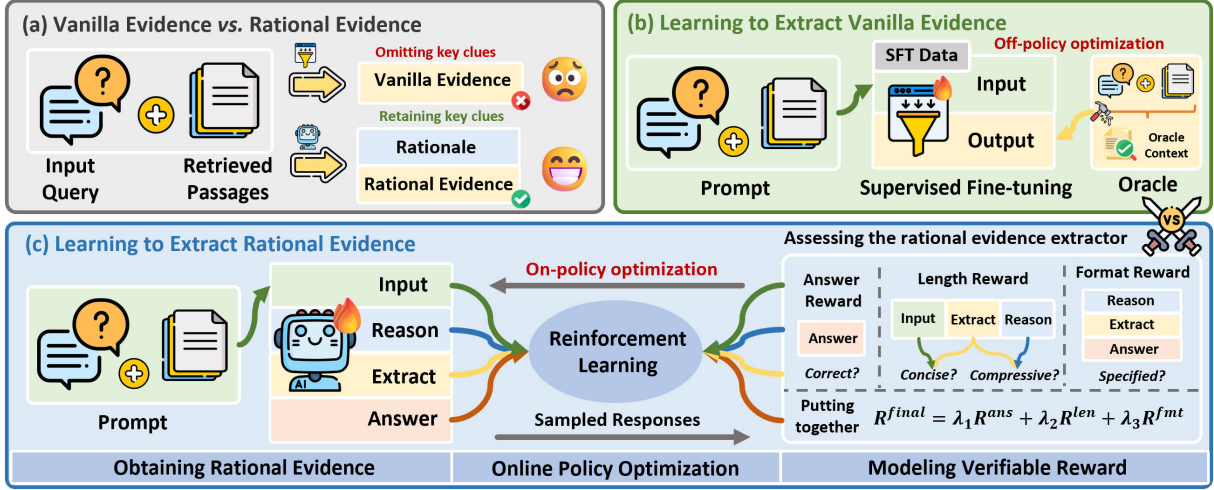


Figure 3: The overall framework of EviOmni and previous vanilla one. (a) Compared to vanilla evidence, rational evidence is more skilled in extracting key clues. (b) Existing works typically learn to extract vanilla evidence via SFT, where the output, *i.e.*, oracle context containing key clues, is usually created by hand-crafted strategies. (c) Our EviOmni incentivizes rational evidence extraction capability in the extractor π_θ via RL with verifiable rewards.

spectively, *i.e.*, `<reason> ... </reason>` and `<extract> ... </extract>`. The prompt used for rational evidence generation is provided in Appendix G. Given the rationale and the rational evidence, a question arises: *How to assess their quality?* In §2.2, we use answer recall, a heuristic metric; similarly, previous works (Wang et al., 2023) employ String Inclusion or Lexical Overlap to measure answer recall or unigram overlap, respectively. However, these metrics do not directly align with the ultimate goal of RAG, namely, generating output as close to the golden answer as possible.

Knowledge Token Masking. To this end, we propose evaluating the quality of the rationale and rational evidence by assessing the generated outputs that are conditioned on them. As is well known, causal attention mechanisms in LLMs (Radford et al., 2018; Vaswani et al., 2017) aggregate information from preceding contexts, raising the question: *How to generate the output conditioned on them respectively without information leakage?* To address this issue, we apply knowledge token masking on each response to disentangle the rationale and rational evidence: (1) Masking the rational evidence e to assess the quality of the rationale r ; (2) Masking both the retrieved passages P and the rationale r to assess the quality of the rational evidence e . We adopt hard masking on the input rather than soft masking on the attention, because the latter may cause information leakage due to the hidden states already aggregating information from preceding contexts. After token masking, we first

prefill input contexts from scratch and then generate three outputs conditioned on different contexts:

$$\begin{aligned}
 o_r &= \mathcal{M}_G(\cdot | q, P, r, \mathbb{R}), \\
 o_e &= \mathcal{M}_G(\cdot | q, \mathbb{R}, \mathbb{R}, e), \\
 o_f &= \mathcal{M}_G(\cdot | q, P, r, e),
 \end{aligned} \tag{4}$$

where o_r , o_e , and o_f represent the outputs conditioned on the rationale, rational evidence, and full context, respectively; \mathbb{R} denotes the hard masking during input. Note that we also generate the output o_f conditioned on the full context, since introducing o_f into optimization can facilitate convergence.

3.2 Modeling Verifiable Reward

The reward functions play a key role in Reinforcement Learning (RL) (Ouyang et al., 2022; Wang et al., 2024; DeepSeek-AI et al., 2025), which can guide the optimization towards desired properties. To train EviOmni via RL, we design three types of verifiable reward functions *w.r.t.* three primary desired properties for evidence extraction: (1) The correctness of generated outputs; (2) The comprehensiveness of rationale and the conciseness of rational evidence; and (3) The format specification of responses. For reward modeling, we first collect a set of PREAO-tuple $\langle P, r, e, a, o_{r,e,f} \rangle$. Each PREAO-tuple consists of retrieved Passages P , rationale r , rational evidence e , golden answer a , and generated output $o_{r,e,f}$. Then, we design three types of reward functions to assess the rational evidence extractor against the three properties:

Answer Reward. It focuses on the correctness of the outputs enclosed within `<answer>` and `</answer>` tags. However, different downstream tasks (e.g., QA, fact verification, and dialog generation) of RAG use different metrics to evaluate correctness, e.g., Exact Match (EM) and F_1 score. Given that, a question naturally arises: *How to uniformly measure the answer reward across various RAG downstream tasks?* To address this issue, we employ the unigram F_1 to measure the answer rewards in a unified and balanced manner: $R_*^{ans} = F_1(a, o_*)$, $* \in \{r, e, f\}$, where $R_*^{ans} \in [0, 1]$ is the answer reward for the output o_* . If o_* is analogous to a , then R_*^{ans} is close to 1.0; otherwise, 0.0.

Length Reward. It focuses on two aspects. The first one is the comprehensiveness of rationale, enclosed within `<reason>` and `</reason>` tags, where the rationale typically needs to be relatively long to identify any clues lying in retrieval contents. The second one is the conciseness of rational evidence, enclosed within `<extract>` and `</extract>` tags. In contrast, the rational evidence needs to be relatively short to accelerate inference speed. To quantify “long” or “short”, a reference system is required. Following prior work (Wang et al., 2023; Zhao et al., 2024a; Zhu et al., 2024), we use the retrieved passages P as the reference system for rational evidence e , encouraging it to be significantly shorter than P while preserving key clues. Conversely, e serves as the reference for the rationale r , encouraging it to be comprehensive enough to capture all key clues in the retrieved passages. Based on these principles, we define length rewards for both the rationale and rational evidence, denoted as R_r^{len} and R_e^{len} , respectively, where $R_r^{len}, R_e^{len} \in (0, 1]$. Detailed formulations and analyses of the length rewards are provided in Appendices B and C.

Format Reward. It focuses on whether the response obeys the defined format. Specifically, the model’s evidence reasoning, evidence extraction, as well as final output should be enclosed within the `<reason> ... </reason>`, `<extract> ... </extract>`, and `<answer> ... </answer>` tags, respectively. Therefore, the format reward R^{fmt} is assigned a value of 1 only if responses fully adhere to the formatting requirements; otherwise, it is set to 0.

Putting them together. After obtaining the answer, length, and format rewards, we compute the final reward via a linear weighted sum, as complex weighting is not the focus of this work, and a linear

one generally leads to satisfactory performance:

$$R^{final} = \lambda_1 R^{ans} + \lambda_2 R^{len} + \lambda_3 R^{fmt}, \quad (5)$$

where R^{ans} is the average answer reward; R^{len} is the average length reward; λ_* is the weighting coefficient; R^{final} is the final reward used in RL.

3.3 Online Policy Optimization

Having obtained the final rewards for each response, we optimize EviOmni using the policy optimization algorithm GRPO (Shao et al., 2024; DeepSeek-AI et al., 2025), thereby incentivizing rational evidence extraction capability in it via reinforcement learning. Specifically, for each input question q , GRPO first samples a group of responses $\mathcal{Y} = \{y_1, y_2, \dots, y_G\}$, where G is the group size, and each response comprises the rationale r , rational evidence e , and three conditional outputs o_r , o_e , and o_f (§3.1). Subsequently, GRPO evaluates these responses using the verifiable reward functions (§3.2) and obtains final rewards for each response, denoted as $\mathcal{R} = \{R_1, R_2, \dots, R_G\}$, where we omit the superscript “final” for brevity. Unlike PPO (Schulman et al., 2017), GRPO directly compares the final rewards of candidate responses within the same group without requiring an additional critic model. The advantage of the i -th response is determined by normalizing its reward R_i using the mean and the standard deviation of rewards in \mathcal{R} , which can be formulated as follows:

$$A_i = \frac{R_i - \text{mean}(\mathcal{R})}{\text{std}(\mathcal{R})}, \quad (6)$$

where $\text{mean}(\cdot)$ and $\text{std}(\cdot)$ compute the average and standard deviation of the input set, respectively. However, GRPO’s group normalization may overly magnify minor numerical fluctuations. For example, considering $\mathcal{R} = \{0.49, 0.51\}$, the $\text{mean}(R)$ is 0.5, the $\text{std}(R)$ is 0.01, and the resulting advantages are $\{-1.0, 1.0\}$, which actually magnifies minor fluctuations. To address this issue, we propose clipping $\text{std}(R)$ such that it is at least ϵ_{std} , ensuring the denominator does not become too small:

$$\tilde{A}_i = \frac{R_i - \text{mean}(\mathcal{R})}{\text{clip_std}(\mathcal{R})}, \quad (7)$$

where $\text{clip_std}(\mathcal{R}) = \max(\text{std}(R), \epsilon_{std})$; ϵ_{std} is a clipping value. After obtaining the refined advantages $\{\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_G\}$, we optimize the current policy model π_θ (the evidence extractor $\mathcal{M}_{\mathcal{E}}(\cdot)$)

Datasets	Metrics	WR		VAR					RAR	
		Zero	Full	SeleCtx	LLMLingua-2	RECOMP	FilCo	SEER	CoT	EviOmni
1.5B size model										
NQ	EM	13.74	41.97	24.68	29.36	37.40	36.62	36.93	37.70	<u>41.14</u>
	F1	53.16	<u>70.07</u>	59.52	62.82	67.18	66.63	67.11	68.09	70.77
	CR	-	1.0x	3.44x	4.51x	5.43x	<u>16.3x</u>	13.2x	4.56x	38.1x
TQA [†]	EM	29.10	57.02	45.13	48.67	56.56	54.06	54.57	54.29	<u>56.84</u>
	F1	64.62	<u>80.13</u>	73.40	75.29	79.85	78.56	78.81	79.56	80.85
	CR	-	1.0x	3.38x	4.52x	5.35x	8.55x	<u>10.3x</u>	5.02x	38.8x
HotpotQA [†]	EM	12.36	19.20	15.30	16.64	18.52	18.18	18.60	<u>19.52</u>	20.46
	F1	48.52	53.04	49.65	51.26	52.92	52.15	52.79	<u>53.43</u>	54.20
	CR	-	1.0x	3.40x	4.52x	5.44x	<u>18.3x</u>	15.5x	4.17x	33.0x
7B size model										
NQ	EM	25.04	48.78	33.91	36.51	43.77	44.79	45.01	44.49	<u>46.95</u>
	F1	60.88	74.40	65.71	67.77	71.61	72.30	72.76	73.09	<u>73.99</u>
	CR	-	1.0x	3.44x	4.51x	5.43x	<u>15.4x</u>	12.4x	3.36x	17.9x
TQA [†]	EM	47.31	65.34	58.72	60.50	64.40	63.46	64.20	63.91	<u>64.76</u>
	F1	74.36	84.83	80.95	81.94	84.12	83.66	84.14	84.45	<u>84.74</u>
	CR	-	1.0x	3.38x	4.52x	5.35x	7.77x	<u>9.70x</u>	3.33x	17.6x
HotpotQA [†]	EM	17.95	25.82	21.27	23.45	24.84	25.27	25.81	<u>27.25</u>	28.02
	F1	53.07	58.50	54.90	56.65	58.02	58.15	58.63	<u>59.84</u>	60.30
	CR	-	1.0x	3.40x	4.52x	5.44x	17.5x	14.3x	3.32x	<u>16.5x</u>

Table 2: Overall performance comparison, where the best results are **boldfaced** and the second-best results are underlined. † denotes out-of-domain (OOD) datasets for our EviOmni, since it is only trained on the NQ dataset. The improvements of EviOmni over other VAR and RAR methods are statistically significant with p -value < 0.05 .

via maximizing the following objective function:

$$\begin{aligned}
J_{\text{GRPO}}(\theta) = & \mathbb{E}_{x \sim \mathcal{D}, \{y_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot|x)} \\
& \left[\frac{1}{G} \sum_{i=1}^G \min\left(\frac{\pi_{\theta}(\tilde{y}_i|x)}{\pi_{\theta_{\text{old}}}(\tilde{y}_i|x)} \tilde{A}_i, \right. \right. \\
& \left. \left. \text{clip}\left(\frac{\pi_{\theta}(\tilde{y}_i|x)}{\pi_{\theta_{\text{old}}}(\tilde{y}_i|x)}, 1 - \epsilon, 1 + \epsilon \right) \tilde{A}_i \right) \right. \\
& \left. - \beta D_{\text{KL}}(\pi_{\theta} \parallel \pi_{\text{ref}}) \right], \quad (8)
\end{aligned}$$

where $x = \{q, P\}$ denotes input samples drawn from the dataset \mathcal{D} ; y is the model’s response consisting of $\{r, e, o_r, o_e, o_f\}$, sampled from the old policy $\pi_{\theta_{\text{old}}}$; \tilde{y} is a response consisting of $\{r, e, o_f\}$, where o_f is an output conditioned on full context. This makes \tilde{y} self-contained, and it is therefore used for training; ϵ and β are the PPO clipping hyperparameters and KL weighting coefficient, respectively; π_{ref} represents the reference policy.

4 Experiment

In this section, we conduct extensive experiments on five benchmark datasets to answer the following Research Questions (RQs):

- **RQ1:** How does rational evidence perform compared to vanilla evidence?
- **RQ2:** How do the properties of rational evidence vary with the RL training process?

- **RQ3:** How do different parts of the answer rewards contribute to the final model performance?
- **RQ4:** How does the inference efficiency of EviOmni compare to different types of methods?
- **RQ5:** Can rational evidence perform robustly against retrieval noise?
- **RQ6:** How does rational evidence perform on challenging multi-hop QA? (see Appendix D)
- **RQ7:** Can rational evidence be effective for agentic RAG? (see Appendices E and F)

Dataset	#Train	#Dev	#Test
NQ (Kwiatkowski et al., 2019)	79.1k	8.7k	3.6k
TQA (Joshi et al., 2017)	78.7k	8.8k	11.3k
HotpotQA (Yang et al., 2018)	88.9k	5.6k	5.6k
2WikiQA (Ho et al., 2020)	167.4k	12.5k	12.5k
Musique (Trivedi et al., 2022)	19.9k	2.4k	2.4k

Table 3: Statistics of the datasets.

4.1 Experimental Setup

We experiment on five benchmark QA datasets, including Natural Questions (NQ) (Kwiatkowski et al., 2019), TriviaQA (TQA) (Joshi et al., 2017), HotpotQA (Yang et al., 2018), 2WikiQA (Ho et al., 2020), and MuSiQue (Trivedi et al., 2022), where

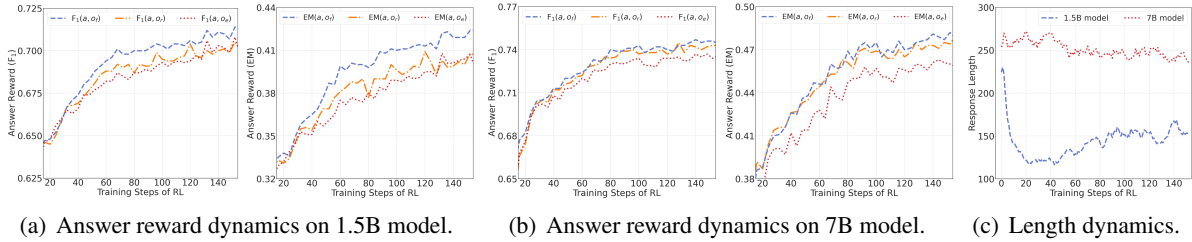


Figure 4: Training dynamics *w.r.t.* answer reward and response length (including ‘reason’, ‘extract’, and ‘answer’).

Models	Rationale	Evidence	Full Ctx	Evidence+Rationale	Evidence+Full Ctx	Evidence+Rationale+Full Ctx
NQ	39.89	40.27	39.53	40.02	<u>40.44</u>	41.14
TQA	56.97	57.38	56.17	<u>57.31</u>	57.26	56.84
HotpotQA	20.30	19.73	20.13	<u>20.34</u>	20.23	20.46
Average	39.05	39.13	38.61	39.22	<u>39.31</u>	39.48

Table 4: Performance comparison (exact match) of EviOmni trained on different combinations of answer rewards.

the first two are open-domain QA, and the last three are multi-hop QA. The detailed statistics of these datasets are presented in Table 3. We use the Exact Match (EM \uparrow) and unigram F $_1\uparrow$ to evaluate QA performance, and use the Compression Ratio (CR \downarrow) to measure efficiency. We employ Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct (Yang et al., 2025) as the initial models and train these two models to extract rational evidence via full parameter fine-tuning for 1 epoch on the NQ dataset. For retrieval, we use Dense Passage Retriever (Karpukhin et al., 2020) to fetch top-5 passages from all Wiki passages. For comparison, we compare EviOmni with the three groups of competitive baselines: **(1) Without Refinement (WR):** (i) Zero-shot (Zero); (ii) Full Passage (Full); **(2) Vanilla Refinement (VAR):** (i) Select Context (SeleCtx) (Li et al., 2023); (ii) LLMLingua-2 (Pan et al., 2024); (iii) RECOMP (Xu et al., 2024); (iv) FILCO (Wang et al., 2023); (v) SEER (Zhao et al., 2024a); **(3) Rational Refinement (RAR):** (i) Chain-of-Thought (CoT) (Wei et al., 2022). Detailed experimental settings and prompts can be found in **Appendices A** and **G**.

4.2 Main Comparison (RQ1)

The overall comparison results are presented in Table 2. From the results, we have the following observations. **(1)** In all cases (18/18), EviOmni achieves the best or second-best results, indicating the superiority of supplementing RAG with rational evidence. **(2)** Surprisingly, directly applying EviOmni trained on NQ and tested on OOD datasets also yields impressive performance, indicating that RL can endow EviOmni with superior

generalization. **(3)** Compared to ‘Full’, EviOmni achieves an extremely high compression ratio (*e.g.*, 38.1x with 1.5B size model on NQ), while its performance is very close to or even surpasses that of ‘Full’. In contrast, although competitive baselines have relatively high compression ratios, their compression is accompanied by large performance degradation. **(4)** By comparing the 1.5B and 7B variants of EviOmni, we find that the 7B one tends to extract more informative evidence than the 1.5B one. **(5)** EviOmni considerably outperforms VAR methods in nearly all cases and provides more compact evidence, showing the necessity of evidence reasoning. **(6)** In HotpotQA, RAR methods largely outperform VAR ones and even surpass ‘Full’, indicating that rational refinement is important to multi-hop QA. These observations fully manifest the effectiveness and efficiency of rational evidence.

4.3 Training Dynamics (RQ2)

The RL training dynamics *w.r.t.* the answer rewards and response length are shown in Figure 4. The results show that the answer rewards of full context (o_f), rationale (o_r), and rational evidence (o_e) are consistently improved during the process of RL. Unsurprisingly, the answer reward of full context generally performs best. Specifically, on the 1.5B model, rationale and rational evidence perform very closely, while on the 7B model, rational evidence performs slightly worse than rationale. We think the main reason is that the 7B model can infer answers based on implicit clues, whereas rational evidence may omit some of them. For the dynamics of response length, the response length of the

Models	NQ	TQA	HotpotQA	Avg
FILCO	0.64	0.82	0.59	0.68
CoT	0.55	0.70	0.71	0.65
EviOmni	0.35	0.41	0.43	0.40

Table 5: Inference latency (seconds/query) on 1.5B.

1.5B model decreases rapidly at the beginning of training and then increases slowly, while that of the 7B model decreases slowly all the time. We think the main reason is that a moderate length is beneficial for improving answer rewards, as an overlong reasoning may confuse answer generation, and a too short reasoning may omit key clues. Therefore, as training continues, the response lengths of the 1.5B model and the 7B model tend to converge. In all, the above observations provide several useful insights for reward modeling as well as designing.

4.4 Ablation Study (RQ3)

To evaluate the impact of different answer rewards on model performance, we train the 1.5B variant of EviOmni models using different combinations of answer rewards with average weighting. As shown in Table 4, among models trained on a single answer reward, training with evidence’s answer reward R_e^{ans} achieves the highest average performance, indicating that the quality of evidence is the most important. For two-answer reward combinations, ‘Evidence+Full Ctx’ slightly outperforms ‘Evidence+Rationale’, with significant gains on NQ, indicating that the optimization of ‘Full Ctx’ matters. Notably, using all answer rewards yields the best performance on NQ and HotpotQA, and the highest average score, indicating the benefits of multi-faceted rewarding. The results manifest the contribution and necessity of each answer reward.

4.5 Inference Efficiency (RQ4)

Table 5 presents the inference latency of EviOmni compared to FILCO (a representative VAR method) and CoT (a representative RAR method) on the 1.5B model. All experiments are conducted on an A800 GPU, with the batch size and max new tokens set to 64 and 768, respectively. The results show that the latency of evidence extraction using EviOmni is considerably shorter than FILCO and CoT. Surprisingly, the average latency of FILCO is slightly longer than that of CoT, while FilCo generates fewer new tokens on average. Our analysis shows that the length’s std of evidence extracted by FILCO is largely higher than that of CoT, where the

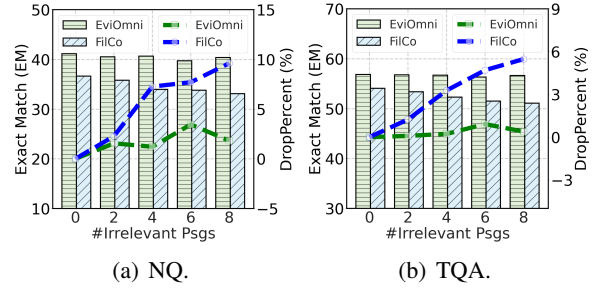


Figure 5: Performance comparisons *w.r.t.* data noise, where Psqs is the abbreviation of passages.

std of FILCO on NQ, TQA, and HotpotQA is 107.5, 136.1, and 81.7, while that of CoT is 67.9, 61.6, and 63.9. The unstable evidence length of FILCO disrupts efficient batching. The above results show the efficiency of EviOmni during deployment.

4.6 Robustness Analysis (RQ5)

In real-world applications, RAG systems often suffer from data noise caused by imperfect retrieval. To simulate this scenario, we randomly sample a certain number (*i.e.*, 0, 2, 4, 6, and 8) of irrelevant passages for each test query. Then, each query is paired with 5 retrieved relevant passages and sampled irrelevant passages. We experiment on the 1.5B model, and the results are presented in Figure 5. The results show that adding noise considerably degrades the performance of FILCO, while the performance drop of EviOmni is relatively small, where the green line is always below the blue one. Particularly, EviOmni with 8 irrelevant passages (psqs) outperforms FILCO without noise. This fully manifests the robustness of EviOmni against noise.

5 Related Works

5.1 Retrieval-Augmented Generation

RAG prevails in LLMs, enhancing LLMs with external non-parametric knowledge to remedy their outdated, incorrect, or incomplete internal parametric knowledge (Lewis et al., 2020; Wu et al., 2022; Guu et al., 2020; Izacard et al., 2023; Asai et al., 2024; Chen et al., 2024; Rao et al., 2023, 2025). The pioneering attempts (Lewis et al., 2020; Guu et al., 2020) demonstrated that augmenting the input context of LLMs with retrieved passages yields significant improvements, especially in knowledge-intensive tasks. While prior works usually retrieve straightforwardly at the beginning, dynamic RAG (Asai et al., 2024; Jeong et al., 2024; Jiang et al., 2023) proposed to adaptively retrieve passages

based on the demand of generation and the complexity of the query. Recently, agentic RAG (Zhang et al., 2025b) proposed to interleave retrieval and reasoning to tackle complex queries. In particular, this kind of work (Song et al., 2025; Jin et al., 2025; Trivedi et al., 2023) focuses on prompting or fine-tuning LLMs as search agents that interact with search tools to fetch external knowledge.

5.2 RAG with Context Compression

Traditional RAG systems typically concatenate all retrieved passages as the context of LLMs (Lewis et al., 2020; Zhao et al., 2026). However, this may introduce data noise and computational overhead due to imperfect retrieval and overlong context (Wang et al., 2023). Recently, many works have attempted to compress context and retain important clues, mainly including two categories: (1) Reranking methods, which rerank retrieved passages and retain top-ranked passages (Hwang et al., 2024b; Mortaheb et al., 2025; Mao et al., 2021; Chen et al., 2025); (2) Summarization methods, which extract relevant information from retrieved passages and consolidate them into a narrative (Wang et al., 2023; Zhao et al., 2024a; Zhu et al., 2024; Xu et al., 2024). Despite their effectiveness, existing works overlook evidence reasoning, thereby risking filtering out key clues as well as struggling with generalization and robustness.

6 Conclusion

In this paper, we first unveil the limitations of vanilla evidence, which risks filtering out crucial clues, and then empirically validate the superiority of rational evidence in substantially retaining key clues. To this end, we propose a rational evidence extraction learning framework, EviOmni, which first reasons and then extracts, is optimized by reinforcement learning with verifiable rewards, incorporates knowledge token masking to avoid information leakage, and ultimately produces compact, high-quality evidence. Extensive experiments across both traditional and agentic RAG scenarios demonstrate the superiority of EviOmni in end-to-end performance, out-of-domain generalization, inference efficiency, and robustness to retrieval noise.

Limitations

Despite our innovations and improvements, EviOmni still has certain limitations, especially cascaded generation between rationale and rational

evidence. This indicates that the rationale must be generated first, followed by the rational evidence, which increases the inference latency to some extent. Although the results in Table 5 show that the inference efficiency of EviOmni is superior to that of the baselines, we believe that further optimization of reasoning efficiency is necessary to achieve higher inference speed. One promising direction is reasoning when necessary (Jiang et al., 2025). For example, for simple queries, the model could directly generate rational evidence without reasoning, thereby reducing redundant computation. We leave this adaptive framework for future research.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (Grant No. 62336008), Shenzhen Basic Research Program (Grant No. JCYJ20241202123503005), and Shenzhen Basic Research Program (Grant No. JCYJ20240813105111016). We sincerely thank all reviewers for their insightful comments and valuable suggestions, which have significantly improved our work.

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 *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.
- Huiyao Chen, Yi Yang, Yinghui Li, Meishan Zhang, Baotian Hu, and Min Zhang. 2025. *Beyond chunking: Discourse-aware hierarchical retrieval for long document question answering*. *arXiv preprint arXiv:2506.06313*.
- Huiyao Chen, Yu Zhao, Zulong Chen, Mengjia Wang, Liangyue Li, Meishan Zhang, and Min Zhang. 2024. *Retrieval-style in-context learning for few-shot hierarchical text classification*. *Trans. Assoc. Comput. Linguistics*, 12:1214–1231.
- 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 81 others. 2025. *Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning*. *CoRR*, abs/2501.12948.
- Yanrui Du, Sendong Zhao, Haochun Wang, Yuhan Chen, Rui Bai, Zewen Qiang, Muzhen Cai, and Bing Qin. 2023. *Make your decision convincing! A unified*

- two-stage framework: Self-attribution and decision-making. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 1101–1112. Association for Computational Linguistics.
- Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. **Tevatron: An efficient and flexible toolkit for neural retrieval**. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 3120–3124. ACM.
- Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. **REALM: retrieval-augmented language model pre-training**. *CoRR*, abs/2002.08909.
- 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 *COLING*, pages 6609–6625. International Committee on Computational Linguistics.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. **Lora: Low-rank adaptation of large language models**. *ICLR*, 1(2):3.
- Taeho Hwang, Sukmin Cho, Soyeong Jeong, Hoyun Song, SeungYoon Han, and Jong C. Park. 2024a. **EXIT: context-aware extractive compression for enhancing retrieval-augmented generation**. *CoRR*, abs/2412.12559.
- Taeho Hwang, Soyeong Jeong, Sukmin Cho, Seungyoon Han, and Jong C. Park. 2024b. **DSLRL: document refinement with sentence-level re-ranking and reconstruction to enhance retrieval-augmented generation**. *CoRR*, abs/2407.03627.
- Gautier Izacard, Patrick S. H. Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2023. **Atlas: Few-shot learning with retrieval augmented language models**. *J. Mach. Learn. Res.*, 24:251:1–251:43.
- Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. 2024. **Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity**. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 7036–7050. Association for Computational Linguistics.
- Lingjie Jiang, Xun Wu, Shaohan Huang, Qingxiu Dong, Zewen Chi, Li Dong, Xingxing Zhang, Tengchao Lv, Lei Cui, and Furu Wei. 2025. **Think only when you need with large hybrid-reasoning models**. *CoRR*, abs/2505.14631.
- Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. **Active retrieval augmented generation**. In *EMNLP*, pages 7969–7992. Association for Computational Linguistics.
- Ziyan Jiang, Xueguang Ma, and Wenhu Chen. 2024. **Longrag: Enhancing retrieval-augmented generation with long-context llms**. *CoRR*, abs/2406.15319.
- Bowen Jin, Hansi Zeng, Zhenrui Yue, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. **Search-r1: Training llms to reason and leverage search engines with reinforcement learning**. *CoRR*, abs/2503.09516.
- Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. **Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension**. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1601–1611. Association for Computational Linguistics.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. **Dense passage retrieval for open-domain question answering**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics.
- Muhammad Khalifa, Lajanugen Logeswaran, Moon-tae Lee, Honglak Lee, and Lu Wang. 2023. **Few-shot reranking for multi-hop QA via language model prompting**. In *Proceedings of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 15882–15897. Association for Computational Linguistics.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. **Natural questions: a benchmark for question answering research**. *Trans. Assoc. Comput. Linguistics*, 7:452–466.
- Patrick S. H. 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 *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.
- Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. 2025. **Search-o1: Agentic search-enhanced**

- large reasoning models. In *EMNLP*, pages 5420–5438. Association for Computational Linguistics.
- Yucheng Li, Bo Dong, Frank Guerin, and Chenghua Lin. 2023. [Compressing context to enhance inference efficiency of large language models](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 6342–6353. Association for Computational Linguistics.
- Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2021. [Reader-guided passage reranking for open-domain question answering](#). In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 344–350. Association for Computational Linguistics.
- Matin Mortaheb, Mohammad Ali Amir Khojastepour, Srimat T. Chakradhar, and Sennur Ulukus. 2025. [Re-ranking the context for multimodal retrieval augmented generation](#). *CoRR*, abs/2501.04695.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. [Training language models to follow instructions with human feedback](#). In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.
- Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Dongmei Zhang. 2024. [Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression](#). In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 963–981. Association for Computational Linguistics.
- Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, and 1 others. 2018. [Improving language understanding by generative pre-training](#).
- Jun Rao, Liang Ding, Shuhan Qi, Meng Fang, Yang Liu, Li Shen, and Dacheng Tao. 2023. [Dynamic contrastive distillation for image-text retrieval](#). *IEEE Trans. Multim.*, 25:8383–8395.
- Jun Rao, Zepeng Lin, Xuebo Liu, Xiaopeng Ke, Lian Lian, Dong Jin, Shengjun Cheng, Jun Yu, and Min Zhang. 2025. [APT: improving specialist LLM performance with weakness case acquisition and iterative preference training](#). In *ACL (Findings)*, Findings of ACL, pages 20958–20980. Association for Computational Linguistics.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. [Proximal policy optimization algorithms](#). *CoRR*, abs/1707.06347.
- 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.
- Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. [REPLUG: retrieval-augmented black-box language models](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 8371–8384. Association for Computational Linguistics.
- 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](#). *CoRR*, abs/2503.05592.
- Hao Sun, Zile Qiao, Jiayan Guo, Xuanbo Fan, Yingyan Hou, Yong Jiang, Pengjun Xie, Yan Zhang, Fei Huang, and Jingren Zhou. 2025. [Zerosearch: Incentivize the search capability of llms without searching](#). *CoRR*, abs/2505.04588.
- Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, YaGuang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, and 38 others. 2022. [Lamda: Language models for dialog applications](#). *CoRR*, abs/2201.08239.
- Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. [Musique: Multi-hop questions via single-hop question composition](#). *Trans. Assoc. Comput. Linguistics*, 10:539–554.
- Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. [Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 10014–10037. Association for Computational Linguistics.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

- Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. [Text embeddings by weakly-supervised contrastive pre-training](#). *CoRR*, abs/2212.03533.
- Shuhe Wang, Shengyu Zhang, Jie Zhang, Runyi Hu, Xiaoya Li, Tianwei Zhang, Jiwei Li, Fei Wu, Guoyin Wang, and Eduard H. Hovy. 2024. [Reinforcement learning enhanced llms: A survey](#). *CoRR*, abs/2412.10400.
- Zhiruo Wang, Jun Araki, Zhengbao Jiang, Md. Rizwan Parvez, and Graham Neubig. 2023. [Learning to filter context for retrieval-augmented generation](#). *CoRR*, abs/2311.08377.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.
- Zhepei Wei, Wei-Lin Chen, and Yu Meng. 2024. [Instrutrag: Instructing retrieval-augmented generation with explicit denoising](#). *CoRR*, abs/2406.13629.
- Yuxiang Wu, Yu Zhao, Baotian Hu, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2022. [An efficient memory-augmented transformer for knowledge-intensive NLP tasks](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 5184–5196. Association for Computational Linguistics.
- Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2024. [RECOMP: improving retrieval-augmented llms with context compression and selective augmentation](#). In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.
- 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, Jixi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 23 others. 2025. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. [Hotpotqa: A dataset for diverse, explainable multi-hop question answering](#). In *EMNLP*, pages 2369–2380. Association for Computational Linguistics.
- Qianchi Zhang, Hainan Zhang, Liang Pang, Hongwei Zheng, Yongxin Tong, and Zhiming Zheng. 2025a. [Finelfilter: A fine-grained noise filtering mechanism for retrieval-augmented large language models](#). *CoRR*, abs/2502.11811.
- Weizhi Zhang, Yangning Li, Yuanchen Bei, Junyu Luo, Guancheng Wan, Liangwei Yang, Chenxuan Xie, Yuyao Yang, Wei-Chieh Huang, Chunyu Miao, Henry Peng Zou, Xiao Luo, Yusheng Zhao, Yankai Chen, Chunkit Chan, Peilin Zhou, Xinyang Zhang, Chenwei Zhang, Jingbo Shang, and 4 others. 2025b. [From web search towards agentic deep research: Incentivizing search with reasoning agents](#). *CoRR*, abs/2506.18959.
- Xinping Zhao, Xinshuo Hu, Zifei Shan, Shouzheng Huang, Yao Zhou, Xin Zhang, Zetian Sun, zhenyu liu, Dongfang Li, Xinyuan Wei, Youcheng Pan, Yang Xiang, Meishan Zhang, Haofen Wang, Jun Yu, Baotian Hu, and Min Zhang. 2026. [KaLM-embedding-v2: Superior training techniques and data inspire a versatile embedding model](#). In *The Fourteenth International Conference on Learning Representations*.
- Xinping Zhao, Dongfang Li, Yan Zhong, Boren Hu, Yibin Chen, Baotian Hu, and Min Zhang. 2024a. [SEER: self-aligned evidence extraction for retrieval-augmented generation](#). In *EMNLP*, pages 3027–3041. Association for Computational Linguistics.
- Xinping Zhao, Jindi Yu, Zhenyu Liu, Jifang Wang, Dongfang Li, Yibin Chen, Baotian Hu, and Min Zhang. 2024b. [Medico: Towards hallucination detection and correction with multi-source evidence fusion](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 34–45.
- Xinping Zhao, Yan Zhong, Zetian Sun, Xinshuo Hu, Zhenyu Liu, Dongfang Li, Baotian Hu, and Min Zhang. 2025. [Funnelrag: A coarse-to-fine progressive retrieval paradigm for RAG](#). In *NAACL (Findings)*, pages 3029–3046. Association for Computational Linguistics.
- Kun Zhu, Xiaocheng Feng, Xiyuan Du, Yuxuan Gu, Weijiang Yu, Haotian Wang, Qianglong Chen, Zheng Chu, Jingchang Chen, and Bing Qin. 2024. [An information bottleneck perspective for effective noise filtering on retrieval-augmented generation](#). In *ACL (1)*, pages 1044–1069. Association for Computational Linguistics.

A Experimental Setup

A.1 Datasets and Metrics

We experiment on five knowledge-intensive benchmark QA datasets, including Natural Questions (NQ) (Kwiatkowski et al., 2019), TriviaQA (TQA) (Joshi et al., 2017), HotpotQA (Yang et al., 2018), 2WikiQA (Ho et al., 2020), and MuSiQue (Trivedi et al., 2022), where the first two are open-domain QA, and the last three are multi-hop QA. The detailed statistics of these five datasets are provided in Table 3, where the test set for HotpotQA is unavailable; therefore, we use its development set as a substitute for the test

set. Additionally, since the test set of 2WikiQA or MuSiQue is not annotated, we use their development sets for evaluation.

For evaluation, we adopt the Exact Match (EM) and unigram F_1 score to evaluate QA performance. EM examines exact correctness while F_1 calculates the degree of lexical overlap, offering a more fine-grained view of how well the prediction aligns with the golden answer. To measure the improvement of the computational efficiency, we employ the Compression Ratio (CR) following the previous works (Hwang et al., 2024a; Pan et al., 2024), where CR computes the ratio of the total length of the retrieval passages to the length of the extracted evidence:

$$\text{CR} = \frac{\sum_{p \in P} L_p}{L_e}, \quad (9)$$

where $P = \{p_1, p_2, \dots, p_k\}$ is a set of top- k retrieved passages; e is the extracted evidence; L_p and L_e denote the length of the passage p and evidence e , respectively.

A.2 Implementation Details

We employ Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct (Yang et al., 2025) as the initial models. We optimize these two models to learn to extract rational evidence through full parameter fine-tuning for 1 epoch on the NQ dataset. The learning rate is $1e^{-6}$, and we select the best-performing model on the dev set. For modeling verifiable rewards, we tune τ and γ within the ranges of $\{0.1, 0.2, 0.5, 1.0\}$ and $\{0.1, 0.3, 0.5, 0.8, 1.0\}$, respectively. For the length threshold ω and the weighting coefficients (*i.e.*, λ_1 , λ_2 , and λ_3), we empirically set ω , λ_1 , λ_2 , and λ_3 as 0.9, 0.8, 0.1, and 0.1, respectively, which commonly leads to satisfactory performance. For policy optimization, we set ϵ_{std} as 0.1 to stabilize training. During training, we set the PPO clipping hyperparameter ϵ and the coefficient β controlling Kullback–Leibler (KL)-divergence as 0.2 and $1e^{-2}$, respectively. For QA generators, we employ LoRA (Hu et al., 2022) to fine-tune Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct to predict the golden answer a based on the query and retrieved passages (q, P) or only the query (used for the baseline ‘Zero’), where LoRA rank is set as 8. Furthermore, each experiment is run five times, and the average results are reported.

A.3 Passage Retrieval

Following previous works (Wang et al., 2023; Zhu et al., 2024; Zhao et al., 2024a; Zhang et al.,

Dataset	Recall			NDCG		
	Train	Dev	Test	Train	Dev	Test
NQ	78.74	73.07	74.07	68.30	61.95	63.08
TQA	82.35	77.97	77.77	76.06	70.32	70.35
HotoptQA	34.02	28.45	-	27.36	22.39	-
2WikiQA	-	-	-	-	-	-
Musique	-	-	-	-	-	-

Table 6: Recall and NDCG of the top-5 retrieval passages in terms of training, development, and test sets.

2025a), we retrieve Wikipedia (Wiki) passages for all datasets. We use Dense Passage Retriever (DPR) (Karpukhin et al., 2020) to retrieve the top-5 passages from all Wiki passages, where we use the December 2018 Wiki dump (Karpukhin et al., 2020) as the retrieval corpus, and set the chunk size as 100 words. We use the retrieval toolkit, Tevatron (Gao et al., 2023), to perform corpus encoding, query encoding, and passage retrieval. Table 6 shows the retrieval performance *w.r.t.* Recall@5 and NDCG@5, where the retrieval passage that contains the golden answer is treated as the positive passage. Besides, for HotpotQA, we compute Recall and NDCG only for the ‘‘bridge’’ questions, while ignoring yes/no and comparison questions, following previous works (Jiang et al., 2024; Khalifa et al., 2023). The retrieval performance for HotpotQA is relatively low because it is a multi-hop QA dataset, and the answers are usually not spans within retrieval passages. Additionally, for 2WikiQA and MuSiQue, we adopt the *Distractor* setting, where each question is typically equipped with two gold paragraphs and 8-9 distractor paragraphs. As passage retrieval is not required under this setting, we do not report retrieval performance for these two datasets.

A.4 Comparison Baselines

To verify the effectiveness of EviOmni, we compare it with the following three groups of competitive baselines:

(1) **Without Refinement (WR)** includes (i) Zero-shot (Zero) generates output relying on LLMs’ parametric knowledge; (ii) Full Passage (Full) feeds all retrieval passages into LLMs;

(2) **Vanilla Refinement (VAR)** directly extracts evidence without explicit thinking, which includes (i) Select Context (SeleCtx) (Li et al., 2023) identifies and prunes redundancy in the input context based on perplexity; (ii) LLMlingua-2 (Pan et al., 2024) distills compression knowledge from GPT-4 to reduce crucial information losing; (iii) RE-

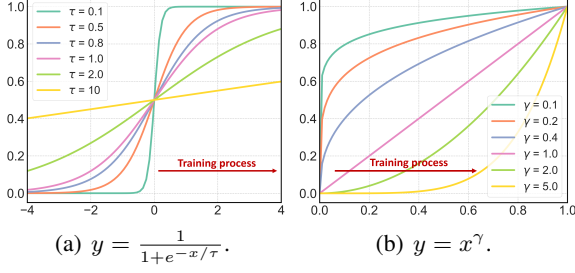


Figure 6: Illustration of length rewards in terms of different skewness or smoothness controlled by τ and γ , where we directly plot the basic functions for clarity.

COMP (Xu et al., 2024) selects useful sentences from retrieved documents; (iv) FILCO (Wang et al., 2023) trains a context filtering model to identify key clues; (v) SEER (Zhao et al., 2024a) learns to extract desired evidence via self-aligned learning;

(3) **Rational Refinement (RAR)** includes (i) Chain-of-Thought (CoT) (Wei et al., 2022) generates query-related information from retrieval passages with explicit thinking.

B Length Reward Formulations

We define length rewards for both the rationale r and the rational evidence e to encourage the rationale to be comprehensive while keeping the rational evidence concise.

Length Reward of Rationale. The rationale r should be sufficient to identify all key clues in the retrieved passages, serving as guidance for rational evidence extraction, which can be formulated as:

$$R_r^{len} = \begin{cases} \sigma(\frac{L_r}{L_e} - 1)/\tau, & \frac{L_r}{L_e} \geq 1, \\ \sigma(1 - \frac{L_e}{L_r})/\tau, & \text{otherwise.} \end{cases} \quad (10)$$

where $\sigma(\cdot)$ denotes the sigmoid function; $R_r^{len} \in (0, 1]$; L_r and L_e represent the length of rationale and rational evidence, respectively.

Length Reward of Rational Evidence. The rational evidence e should be concise relative to the retrieved passages P , while preserving key clues:

$$R_e^{len} = \begin{cases} 1.0, & (1 - \frac{L_e}{L_P}) \geq \omega, \\ (1 - \frac{L_e}{L_P})^\gamma, & \text{otherwise.} \end{cases} \quad (11)$$

where $R_e^{len} \in (0, 1]$; L_P is the length of retrieved passages; τ and γ are used to control the skewness or smoothness of length rewards, thereby avoiding excessively strict preferences for longer rationale or shorter rational evidence; ω is a threshold to avoid falling into a trivial solution.

Dataset		2WikiQA [†]		
Metric		EM	F1	CR
WR	Zero	18.69	49.99	-
	Full	32.39	60.68	1.0×
VAR	LLMLingua-2	25.49	54.77	4.50×
	FilCo	27.50	56.58	28.32×
RAR	CoT	37.20	64.44	6.03×
	EviOmni	38.76	66.48	41.89×

Table 7: Experimental Results on 2WikiQA, where [†] denotes OOD evaluation for the proposed EviOmni.

Dataset		MuSiQue [†]		
Metric		EM	F1	CR
WR	Zero	1.82	41.92	-
	Full	11.75	48.89	1.0×
VAR	LLMLingua-2	7.70	45.46	4.55×
	FilCo	8.36	45.74	67.50×
RAR	CoT	16.01	53.05	11.25×
	EviOmni	18.00	55.50	96.60×

Table 8: Experimental Results on MuSiQue, where [†] denotes OOD evaluation for the proposed EviOmni.

C Length Reward Analysis

Figure 6 illustrates the length rewards in terms of different skewness or smoothness controlled by τ and γ , respectively. From Figure 6(a), we can see that the relatively long rationale ($x = 2$) will be assigned large rewards, whereas a shorter rationale ($x = -2$) corresponds to small rewards. On the contrary, as shown in Figure 6(b), shorter rational evidence ($x = 0.8$) will be assigned large rewards. By adjusting τ and γ , we can control the skewness or smoothness of length rewards, as shown in the line charts of different colors in Figure 6.

D Additional Experiments on Multi-hop QA Benchmarks (RQ6)

To further verify the effectiveness of EviOmni, we conducted additional experiments on more challenging multi-hop QA benchmarks, *i.e.*, 2WikiQA (Ho et al., 2020) and MuSiQue (Trivedi et al., 2022), using the Qwen2.5-1.5B-Instruct. All experiments are conducted under the *Distractor* setting, which introduces extra irrelevant documents into the context, thereby posing more challenging evaluation scenarios. From the results shown in Table 7 and Table 8, EviOmni achieves the best EM/F1 scores and extremely high compression ratio on both 2WikiQA and MuSiQue.

In particular, compared with ‘Full’ and VAR methods, RAR methods consistently yield significantly better performance across multi-hop QA

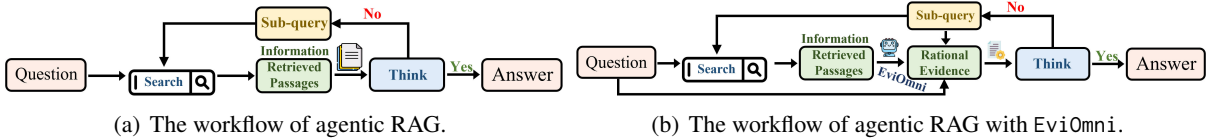


Figure 7: The workflow of agentic RAG (Left) and that with EviOmni (Right), where “Yes” and “No” indicate whether the model determines that the accumulated retrieved information or rational evidence is sufficient to answer the query, or insufficient and thus requires further search, respectively.

benchmarks, indicating that rational refinement is more effective at filtering redundant or noisy information while preserving key evidence necessary for multi-hop QA. On the other hand, compared with CoT, EviOmni incentivizes the rational evidence extraction capability into the model itself, rather than relying on coarse chains of thought, which leads to consistently higher EM and F1 scores together with an extremely high compression ratio. Notably, these gains are achieved under OOD evaluation settings, which further demonstrates the strong generalization of EviOmni. These results highlight the clear advantage of EviOmni in handling complex multi-hop QA scenarios, demonstrating its superior capacity to balance generation accuracy and inference efficiency.

E Potential of EviOmni on Agentic RAG (RQ7)

Driven by reinforcement learning, agentic RAG models (Jin et al., 2025; Li et al., 2025; Song et al., 2025; Sun et al., 2025) have become an effective approach for complex QA tasks. They follow a `think` → `search` → `information` → `think` → `search` → `information` → ... → `answer` loop, and as retrieval depth increases, accumulated context can grow rapidly, highlighting the need for a lightweight and effective denoising model to ensure both inference efficiency and generation accuracy.

To this end, we incorporate EviOmni into the agentic RAG loop as a lightweight rational evidence extractor. The general workflow of agentic RAG is illustrated in Figure 7(a), while the workflow of agentic RAG with EviOmni integration is shown in Figure 7(b). As shown in Figure 7(b), EviOmni plays a crucial role in the agentic RAG loop. Specifically, its input query of EviOmni comprises (i) the original question, (ii) the reasoning information enclosed within `<think> ... </think>`, and (iii) the sub-query enclosed within `<search> ... </search>`. We adopt this query augmentation strategy since the sub-query alone is usually am-

biguous to guide precise evidence extraction. At each step, the retrieved passages, together with the augmented query, are fed into EviOmni to extract rational evidence, thereby improving both the efficiency and accuracy of agentic RAG.

To evaluate the effectiveness of EviOmni in agentic RAG, we conduct experiments on a representative agentic RAG framework, *i.e.*, Search-R1 (Jin et al., 2025). We consider both the 7B² and 32B³ variants of Search-R1 to verify the robustness of EviOmni across different model scales. For retrieval, we follow the original Searcher-R1 settings, using the 2018 Wikipedia dump (Karpukhin et al., 2020) as the knowledge source and E5 (Wang et al., 2022) as the dense retriever. The maximum number of actions is set to 4. For EviOmni, we adopt the 7B variant to serve as a lightweight rational evidence extractor within the agentic RAG loop. We evaluate performance using Exact Match (EM) for QA accuracy and Compression Ratio (CR) to quantify the reduction of retrieved contents after applying EviOmni. For evaluation, we randomly sample 500 instances from the test or development set of NQ, TQA, HotpotQA, 2WikiQA, and MuSiQue, respectively.

The experimental results are summarized in Table 9. We compare Search-R1 against its variant augmented with EviOmni. The results show that integrating EviOmni consistently improves QA accuracy across both the 7B and 32B settings, yielding performance gains in 6/10 cases and comparable results in 4/10 cases, demonstrating strong generalization across both open-domain and multi-hop QA tasks. Overall, EviOmni improves the average QA performance across five QA tasks. For example, under the 7B setting, the average EM increases from 36.28 to 37.36. On the other hand, across both

²https://huggingface.co/PeterJinGo/SearchR1-nq_hotpotqa_train-qwen2.5-7b-it-em-grpo-v0.3

³https://huggingface.co/PeterJinGo/SearchR1-nq_hotpotqa_train-qwen2.5-32b-em-grpo-v0.3

Dataset	Open-domain QA				Multi-hop QA						Average	
	NQ		TQA		HotpotQA		2WikiQA		MuSiQue			
Metric	EM	CR	EM	CR	EM	CR	EM	CR	EM	CR	EM	CR
Search-R1-7B	37.20	1.0x	60.40	1.0x	36.60	1.0x	30.80	1.0x	16.40	1.0x	36.28	1.0x
Search-R1-7B w/ EviOmni	38.20	7.88x	63.20	8.29x	35.00	9.07x	34.80	10.47x	15.60	9.69x	37.36	9.08x
Search-R1-32B	46.60	1.0x	63.80	1.0x	37.20	1.0x	36.60	1.0x	14.40	1.0x	39.72	1.0x
Search-R1-32B w/ EviOmni	45.40	8.72x	65.60	8.73x	37.80	9.29x	35.80	11.76x	14.80	9.59x	39.88	9.62x

Table 9: Performance comparison between Search-R1 (Jin et al., 2025) with and without the proposed EviOmni.

Metric	Average number of searches↓				
Dataset	NQ	TQA	HoppotQA	2WikiQA	MuSiQue
Search-R1-7B					
w/o EviOmni	1.16	1.20	1.63	2.19	2.17
w/ EviOmni	1.27	1.22	1.67	2.12	1.98
Search-R1-32B					
w/o EviOmni	1.16	1.21	1.73	2.30	2.22
w/ EviOmni	1.15	1.15	1.66	2.06	2.02

Table 10: Study of the average number of searches, where lower values indicate better inference efficiency.

types of tasks, the average CR is approximately 9 times, substantially improving agentic RAG inference efficiency. Compared with 32B or larger agentic RAG models, the overhead of EviOmni is negligible, making it a lightweight and practical solution for large-scale, industrial agentic RAG systems. Additionally, we observe that EviOmni achieves higher compression ratios for multi-hop QA than for open-domain QA. This is mainly because multi-hop QA tasks involve more search turns and thus more irrelevant retrieved contents to compress. In conclusion, these results demonstrate that EviOmni plays a crucial role in agentic RAG systems of various scales, improving both the inference efficiency and QA accuracy across open-domain and multi-hop QA tasks.

F Study on Average Number of Searches (RQ7)

Table 10 presents the average number of searches under different settings. Overall, integrating EviOmni results in comparable or fewer search turns, indicating that it typically alleviates search overhead. For the 7B setting, EviOmni leads to comparable search counts on open-domain QA datasets (*i.e.*, NQ and TQA), while considerably reducing #Avg.Searches on multi-hop QA datasets, particularly on 2WikiQA and MuSiQue. In contrast, in the 32B setting, EviOmni consistently reduces the average number of search turns across

all datasets, demonstrating more stable efficiency improvements at larger model scales. Comparing open-domain and multi-hop QA, multi-hop tasks inherently require more search turns due to their compositional reasoning nature. Consequently, EviOmni is more effective on multi-hop QA, where longer search chains leave greater room for eliminating redundant searches.

G Prompts

We provide the prompts used for retrieval-augmented QA, closed-book QA, CoT generation, and rational evidence extraction in Table 11, Table 12, Table 13, and Table 14, respectively.

H Case Studies on Traditional RAG

Through extensive case studies, we found the reasons for the failure of vanilla evidence mainly lie in threefold, *i.e.*, **Incompleteness**, **Irrelevance**, and **Inaccuracy**, termed as “**3I**”.

(1) **Incompleteness**, where the evidence provides some relevant information but lacks key clues, an example on the left of Figure 8, vanilla evidence provides information about the award category (Best Actor) but fails to provide the specific name of the award winner.

(2) **Irrelevance**, where the evidence provides information about the wrong entity, event, or topic, an example in the middle of Figure 8, vanilla evidence provides irrelevant information about a different work and year.

(3) **Inaccuracy**, where the evidence contains incorrect information, such as wrong dates and names, an example in the right of Figure 8, vanilla evidence contains a factual error, attributing the opera “The Midsummer Marriage” to the wrong composer (Benjamin Britten instead of Michael Tippett).

In contrast, rational evidence shows a deep contextual understanding of the query’s intent, enables verification of factual accuracy, and supports rea-

soning to derive the correct and relevant evidence. It moves beyond vanilla evidence extraction towards rational evidence extraction.

I Case Studies on Agentic RAG

To verify the effectiveness of agentic RAG when integrated with EviOmni, we incorporate EviOmni into the agentic RAG loop as a lightweight rational evidence extractor. We use the representative agentic RAG model Search-R1 (Jin et al., 2025) and adopt the 7B variant of EviOmni for rational evidence extraction. The retrieved passages are enclosed within `<information>``</information>`, while rational evidence is enclosed within `<information w/ EviOmni>` ... `</information w/ EviOmni>`. Four cases are drawn from Search-R1, including two cases where Search-R1 succeeds and two cases where it fails. These cases are used to analyze how EviOmni improves agentic RAG:

Case I (Table 15). Integrating EviOmni ensures that only essential evidence is fed into Search-R1 at each step, substantially reducing context length and improving efficiency. Moreover, EviOmni triggers early stopping when sufficient evidence has been gathered in the second step, whereas Search-R1 continues with redundant searches. This shows that integrating EviOmni not only facilitates agentic RAG but also enables early stopping.

Case II (Table 16). EviOmni precisely extracts key evidence at each step, reducing context length and improving overall efficiency. Using the rational evidence, Search-R1 generates the correct answer, demonstrating the effectiveness of rational evidence in guiding agentic RAG.

Case III (Table 17). Search-R1 fails in this case, being misled by irrelevant retrieved information and producing an incorrect answer. In contrast, EviOmni explicitly detects that *“The passage does not contain any information related to The Rap Game or the contestants mentioned in the question”* during the second step. This demonstrates that EviOmni both compresses context to improve efficiency and prevents agentic RAG from being misled by irrelevant documents.

Case IV (Table 18). Search-R1 fails to generate appropriate sub-queries to decompose this complex problem, attempting to answer without sufficient evidence. In contrast, EviOmni consistently identifies that the retrieved passages lack the necessary

information to answer the question. This allows the model to refuse to answer rather than produce an incorrect answer based on insufficient evidence.

In conclusion, rational evidence benefits agentic RAG through the “4R” properties: (1) **Reduction** of redundant context, feeding only essential evidence at each step; (2) **Refusal** to answer when the available evidence is insufficient; (3) **Robustness** to noisy or irrelevant contents, guiding the model away from misleading information; and (4) **Right timing**, triggering early stopping once sufficient evidence is collected. When agentic RAG models succeed, integrating EviOmni leverages these properties to shorten context, accelerate inference, and avoid unnecessary search. Conversely, when they fail, EviOmni helps identify missing/irrelevant information, preventing the model from being misled and reducing the risk of producing wrong answers.

Prompt for Retrieval-Augmented QA

[Instruction]

You are a helpful assistant. Your task is:

1. Read the given question and use the documents provided to answer the question.
2. If the documents don't work, please answer the question based on your own knowledge.

Question: {question}

Document: {document}

Answer:

Table 11: The prompt for retrieval-augmented QA.

Prompt for Closed-book QA

[Instruction]

You are a helpful assistant. Your task is:

1. Read the given question and then answer the question directly.
2. Give a short answer to the question based on your own knowledge.

Question: {question}

Answer:

Table 12: The prompt for closed-book QA.

Prompt for CoT Generation

[Instruction]

You are a helpful assistant. Your task is:

Read the given documents, and answer the question below.

Question: {question}

Document: {document}

Let's think step by step.

Table 13: The prompt for Chain-of-Thought generation.

Prompt for Rational Evidence Extraction

[Instruction]

You are a highly skilled knowledge reasoner and extractor.

Your task is to carefully read the given question and passages to reason how the passages lead to the answer and extract relevant information that may be used to answer the question.

Follow these steps:

1. In the `<reason></reason>` tag, perform the following steps. Question Analysis: Analyze the question to understand the specific information they are seeking. Identify the key concepts, entities, and relationships involved. Passage Analysis: For each passage, carefully read and identify sentences or phrases that are useful for answering the given question.
2. In the `<extract></extract>` tag, synthesize useful information from the passages into a coherent narrative. Organize the information logically and concisely.
3. In `<answer></answer>` tags, give a short answer to the given question, based on the passages, reasoning information, and extracted knowledge. If none of them work, please answer the question based on your knowledge.

Question: {question}

Passages: {passages}

Table 14: The prompt for rational evidence extraction, where generation is terminated when encountering token '`</extract>`'.

Incompleteness	Irrelevance	Inaccuracy
<p>Question: who starred in the English-language Danish experimental horror film which Lars von Triers received award for</p> <p>Answer: ["Willem Dafoe"]</p> <p>Retrieval Passages: ... Lars von Triers "Antichrist" was the big winner, receiving both the awards for Best Danish Film, Best Actor (Willem Dafoe), Best Actress (Charlotte Gainsbourg), Best Cinematographer (Anthony Dod Mantle) ...</p> <p>Vanilla Evidence: Lars von Trier" is mentioned in Passage 1 as the director of "Antichrist," which won awards including Best Danish Film, Best Actor [Incomplete Information], Best Actress,</p> <p>Rational Evidence: Lars von Triers film "Antichrist" won awards for Best Danish Film, Best Actor (Willem Dafoe), Best Actress (Charlotte Gainsbourg),</p> <p>Failure Analysis: Vanilla evidence only mentions the awards but not the leading actors, i.e., Willem Dafoe.</p>	<p>Question: In what year was Shin Hyun-joon in the successful South Korean comey sequel?</p> <p>Answer: ["2005"]</p> <p>Retrieval Passages: It became a huge hit not only in Korea, but throughout Asia. Their follow-up "My Girl" (2005), about a cheeky con artist who pretends to be the long-lost granddaughter of a rich man, was equally successful.</p> <p>Vanilla Evidence: Shin Hyun-joon portrayed Lee Seon Joon in "Sungkyunkwan Scandal." This show was broadcasted in 2010. [Irrelevant Information]</p> <p>Rationale: ... Passage 2 states that "My Girl" (2005) was a successful follow-up, which matches the question's context ...</p> <p>Rational Evidence: "My Girl" (2005) was a successful follow-up to Shin Hyun-joon's original work.</p> <p>Failure Analysis: Vanilla Evidence provides irrelevant information about a different work and year.</p>	<p>Question: Ross Pople worked with which English composer known for his opera "The Midsummer Marriage"?</p> <p>Answer: ["Michael Tippett"]</p> <p>Retrieval Passages: Michael Tippett Sir Michael Kemp Tippett (2 January 1905 – 8 January 1998) was an English composer who In his lifetime he was sometimes ranked with his contemporary Benjamin Britten as one of the leading British composers of the 20th century.</p> <p>Vanilla Evidence: Ross Pople worked with Benjamin Britten, the English composer known for his opera "The Midsummer Marriage. [Inaccurate Information]</p> <p>Rational Evidence: Michael Tippett is an English composer known for his opera "The Midsummer Marriage.</p> <p>Failure Analysis: Vanilla evidence incorrectly attributes "The Midsummer Marriage" to Benjamin Britten instead of Michael Tippett.</p>

Figure 8: Three main failure issues, termed as **3I**: (1) **Incompleteness**, where the evidence provides some relevant information, but lacks the key clues needed to answer the questions; (2) **Irrelevance**, where the evidence provides information about the wrong entity, event, or topic; and (3) **Inaccuracy**, where the evidence contains incorrect information, such as wrong dates, names, or relationships.

Question: Curious is a women's fragrance by a singer born in what city and state?

Ground Truth: McComb, Mississippi

Step 1:

<think> I need to find out which city and state a singer, by whom the women's fragrance "Curious" was created, was born in. </think>

<search> Curious fragrance information </search>

<information> Doc 1(Title: "Curious (fragrance)") Curious (fragrance) Curious is a women's fragrance by Britney Spears for Elizabeth Arden, and is the first perfume to be endorsed by Britney Spears, preceding "'Fantasy.'" Spears reportedly earned \$52 million from the endorsement. "'Curious'" was released in September 2004, and was very successful internationally. It was the number one fragrance of 2004 in department stores, and in 2005, "'Curious'" was honored by the Fragrance Foundation as Best Women's Fragrance. The fragrance is a white floral scent. Its notes are Louisiana magnolia, golden Anjou pear, lotus flower, tuberose, star jasmine, pink cyclamen, vanilla-infused musk, sandalwood and blonde woods. Elizabeth Arden ... </information>

<information w/ EviOmni> **Britney Spears created the women's fragrance "Curious" for Elizabeth Arden.** However, the passage does not provide information about Britney Spears' birthplace. </information w/ EviOmni>

Step 2:

<think> I need to find out which city and state Britney Spears, the singer behind "Curious" fragrance, was born in. </think>

<search> Britney Spears birthplace </search>

<information> Doc 1(Title: "Britney Spears") the best-selling female albums artist of the 2000s. In 2004, she launched a perfume brand with Elizabeth Arden, Inc., from which sales exceeded US\$1.5 billion. . Spears was born in McComb, Mississippi, the second child of Lynne Irene Bridges and James Parnell Spears. Her maternal grandmother, Lillian Portell, was English (born in London), and one of Spears's maternal great-great-grandfathers was Maltese. Her siblings are Bryan James and Jamie Lynn. Britney was born in the Bible Belt, where socially conservative evangelical Protestantism is a particularly strong religious influence. Spears was baptized into the Southern Baptist Convention, but later in life studied ... </information>

<information w/ EviOmni> **Britney Spears was born in McComb, Mississippi.** This information is explicitly stated in the passage, providing the city and state of her birth. </information w/ EviOmni>

Terminate early because EviOmni believes that sufficient evidence has been obtained

Step 3:

<think> Since Britney Spears was born in McComb, Mississippi, I need to find out which city and state that is. </think>

<search> McComb, Mississippi location </search>

<information> Doc 1(Title: "McComb, Mississippi") McComb, Mississippi McComb is a city in Pike County, Mississippi, United States, approximately south of Jackson. As of the 2010 census, the city had a total population of 12,790. It is the principal city of the McComb, Mississippi Micropolitan Statistical Area. McComb was founded in 1872 after Henry Simpson McComb of the New Orleans, Jackson and Great Northern Railroad, a predecessor of the Illinois Central Railroad (now part of the Canadian National Railway), decided to move the railroad's maintenance shops away from New Orleans, Louisiana, to avoid the attractions of that city's saloons. The railroad purchased land in Pike County, ... </information>

<think> I have found that Britney Spears was born in McComb, Mississippi. Since the question asks for the city and state, I just need to confirm the state, which I already know from the previous search. </think>

<answer> McComb, Mississippi </answer>

Table 15: Agentic RAG with EviOmni case study I, where EviOmni triggers early stopping when it judges the evidence sufficient.

Question: Where are Teide National Park and Garajonay National Park located?

Ground Truth: Canary Islands, Spain

Step 1:

<think> I need to find out where Teide National Park and Garajonay National Park are located. I'll search for it. </think>

<search> Teide National Park and Garajonay National Park location </search>

<information> Doc 1(Title: "Garajonay National Park") decided to take their own lives. Garajonay National Park Garajonay National Park (,) is located in the center and north of the island of La Gomera, one of the Canary Islands (Spain). It was declared a national park in 1981 and a World Heritage Site by UNESCO in 1986. It occupies 40 km (15 sq mi) and it extends into each of the six municipalities on the island. Nearly 450,000 tourists visit the park every year. There is accommodation for tourists just outside the park's boundary. The park is named after the rock formation of Garajonay, the highest point ... </information>

<information w/ EviOmni> Garajonay National Park is located in the center and north of the island of La Gomera, **one of the Canary Islands (Spain).** </information w/ EviOmni>

Step 2:

<think> I found out that Garajonay National Park is located in the center and north of the island of La Gomera, one of the Canary Islands (Spain). Now I need to find the location of Teide National Park. </think>

<search> Teide National Park location </search>

<information> Doc 1(Title: "Teide National Park") Teide National Park Teide National Park (,) is a national park located in Tenerife (Canary Islands, Spain). The national park is centered on Mount Teide, the highest mountain of Spain (3,718 meters high) in addition, the highest volcano in the Atlantic Ocean islands and the third highest volcano in the world from its base in the ocean floor (7,500 meters high). Its national park status was passed on 22 January 1954, making it the third oldest national park in Spain. Pico Viejo, also included within the national park limits, is the second highest volcano in the Canary Islands with ... </information>

<information w/ EviOmni> Teide National Park is located in Tenerife, **one of the Canary Islands, Spain.** The park is centered on Mount Teide, the highest mountain in Spain. </information w/ EviOmni>

<think> I found out that Teide National Park is located in Tenerife (Canary Islands, Spain). Now I have the locations of both Teide National Park and Garajonay National Park. </think>

<answer> Canary Islands, Spain </answer>

Table 16: Agentic RAG with EviOmni case study II, where EviOmni efficiently extracts key evidence at each step, reducing redundant context and improving generation efficiency.

Question: Who was the winner of the season of The Rap Game where a contestant named Lady Leshurr as one of her influences?

Ground Truth: Mani

Step 1:

<think> I need to find the winner of the season of The Rap Game where a contestant named Lady Leshurr was one of her influences. I'll search for it. </think>

<search> who was the winner of the season of The Rap Game where a contestant named Lady Leshurr as one of her influences </search>

<information> Doc 1(Title: "Lady Leshurr") Act. Leshurr came out as pansexual in September 2018. Leshurr has named Ashley Walters, Lisa Maffia, Missy Elliott, and Lil Wayne as her primary influences. Lady Leshurr Melesha Katrina O'Garro (born 15 December 1989) known professionally as Lady Leshurr (,), is a British grime and hip hop rapper, singer, and producer from Kingshurst, Solihull, UK. She is best known for her ""Queen's Speech"" series of freestyles, the fourth of which went viral in 2016. ""The Rap Game"" season 2 competitor Nia Kay stated that Leshurr was one of her influences due to her uniqueness and rap skill. Her subsequent freestyle, ... </information>

<information w/ EviOmni> **Lady Leshurr was an influence on Nia Kay,** who competed in season 2 of The Rap Game. However, the winner of that season is not directly mentioned in the provided passages. </information w/ EviOmni>

Step 2:

<think> I found out that Lady Leshurr was one of the influences for Nia Kay in The Rap Game season 2. Now I need to find the winner of that season. </think>

<search> winner of The Rap Game season 2 </search>

<information> Doc 1(Title: "Chandra Davis") Chandra Davis Chandra Davis (born January 2, 1978 in Detroit, Michigan), a.k.a. London Charles and Deelishis, is an American model, and former reality show contestant. She is best known as the winner of VH1's hit reality TV show ""Flavor of Love 2"". Davis was a contestant on the second season of VH1's Flavor of Love (aged 29). She went on to win the competition when Flavor Flav chose her over Tiffany 'New York' Pollard, who returned after previously named the runner-up on the show's first season. After FOL, Davis announced she is currently not dating Flav, via a VH1 special, ... </information>

<information w/ EviOmni> **The passage does not contain any information related to The Rap Game or the contestants mentioned in the question.** It only discusses Chandra Davis, who won "Flavor of Love 2. </information w/ EviOmni>

<think> I found out that the winner of The Rap Game season 2 is London Charles (Chandra Davis). Now I can provide the answer. </think>

<answer> London Charles </answer>

Table 17: Agentic RAG with EviOmni case study III, where EviOmni explicitly points out that "The passage does not contain any information related to The Rap Game or the contestants mentioned in the question." in the second step.

Question: This expansion of the 2008 magazine article "Is Google Making Us Stoopid?" was a finalist for what award?

Ground Truth: Pulitzer Prize

Step 1:

<think> I need to find out which expansion of the 2008 magazine article "Is Google Making Us Stoopid?" was a finalist for what award. I'll search for it. </think>

<search> This expansion of the 2008 magazine article "Is Google Making Us Stoopid?" was a finalist for what award </search>

<information> Doc 1(Title: "Is Google Making Us Stupid?") Is Google Making Us Stupid? Is Google Making Us Stupid? What the Internet Is Doing to Our Brains (alternatively Is Google Making Us Stoopid?) is a magazine article by technology writer Nicholas G. Carr, and is highly critical of the Internet's effect on cognition. It was published in the July/August 2008 edition of ""The Atlantic"" magazine as a six-page cover story. Carr's main argument is that the Internet might have detrimental effects on cognition that diminish the capacity for concentration and contemplation. Despite the title, the article is not specifically targeted at Google, but more at the cognitive impact of ... </information>

<information w/ Evi0mni> The article "Is Google Making Us Stoopid?" by Nicholas G. Carr was published in the July/August 2008 edition of "The Atlantic" magazine. **However, the passage does not mention any awards the article was a finalist for.** </information w/ Evi0mni>

Step 2:

<think> I found out that the expansion of the 2008 magazine article "Is Google Making Us Stoopid?" was a finalist for an award. I need to find out which award it was finalist for. </think>

<search> which award was the expansion of the 2008 magazine article "Is Google Making Us Stoopid?" finalist for </search>

<information> Doc 2(Title: "Is Google Making Us Stupid?") inconclusive and stated that ""it will take time before it's clear whether we should mourn the old ways, celebrate the new, or learn to stop worrying and love the Net"". Is Google Making Us Stupid? Is Google Making Us Stupid? What the Internet Is Doing to Our Brains (alternatively Is Google Making Us Stoopid?) is a magazine article by technology writer Nicholas G. Carr, and is highly critical of the Internet's effect on cognition. It was published in the July/August 2008 edition of ""The Atlantic"" magazine as a six-page cover story. Carr's main argument is that the Internet might have ... </information>

<information w/ Evi0mni> The passages **do not provide information about the expansion of the 2008 article or the award it was a finalist for.** However, it is common for such articles to be nominated for literary awards like the National Magazine Award. </information w/ Evi0mni>

Step 3:

<think> I found out that the expansion of the 2008 magazine article "Is Google Making Us Stoopid?" was a finalist for an award. I need to find out which award it was finalist for. </think>

<search> which award was the expansion of the 2008 magazine article "Is Google Making Us Stoopid?" finalist for </search>

<information> Doc 1(Title: "Is Google Making Us Stupid?") Is Google Making Us Stupid? Is Google Making Us Stupid? What the Internet Is Doing to Our Brains (alternatively Is Google Making Us Stoopid?) is a magazine article by technology writer Nicholas G. Carr, and is highly critical of the Internet's effect on cognition. It was published in the July/August 2008 edition of ""The Atlantic"" magazine as a six-page cover story. Carr's main argument is that the Internet might have detrimental effects on cognition that diminish the capacity for concentration and contemplation. Despite the title, the article is not specifically targeted at Google, but more at the cognitive impact of ... </information>

<information w/ Evi0mni> **The passage does not provide any information about the award the expansion of the 2008 magazine article "Is Google Making Us Stoopid?" was a finalist for.** The article discusses the original publication details but does not mention any awards. </information w/ Evi0mni>

<answer> National Magazine Award </answer>

Table 18: Agentic RAG with Evi0mni case study IV, where Evi0mni consistently highlights in its extracted rational evidence that the passages do not provide enough information to answer the question.

Algorithm 1 Traditional RAG with EviOmni

Input: Input query q , retriever \mathcal{R} , generator \mathcal{M}_G , EviOmni model \mathcal{M}_E

Output: Final answer o

- 1: Retrieve relevant passages $P \leftarrow \mathcal{R}(q)$
 - 2: Generate rationale $r \leftarrow \mathcal{M}_E(\cdot|q, P)$
 - 3: Generate rational evidence $e \leftarrow \mathcal{M}_E(\cdot|q, P, r)$
 - 4: Generate final answer $o \leftarrow \mathcal{M}_G(\cdot|q, e)$
 - 5: **return** o
-

Algorithm 2 Agentic RAG with EviOmni

Input: Input query q , max iterations T , retriever \mathcal{R} , agentic RAG model \mathcal{M}_G , EviOmni model \mathcal{M}_E

Output: Final answer o

- 1: Initialize history buffer $y \leftarrow \emptyset$
 - 2: Initialize action count $t \leftarrow 0$
 - 3: Initialize final answer $o \leftarrow \text{none}$
 - 4: **while** $t < T$ **do**
 - 5: Conduct step-by-step reasoning $y_t \leftarrow \mathcal{M}_G(\cdot|q, y)$
 - 6: Append y_t to the history buffer $y \leftarrow y + y_t$
 - 7: Generate next action $y_a \leftarrow \mathcal{M}_G(\cdot|q, y)$
 - 8: **if** $\langle \text{search} \rangle$ tag in y_a **then**
 - 9: Extract sub-query q_s from y_a
 - 10: Retrieve relevant passages $P \leftarrow \mathcal{R}(q_s)$
 - 11: Compose context-aware query $q_c \leftarrow q \oplus y_t \oplus q_s$ \triangleright Integrate the original query, reasoning information, and sub-query to form context-aware query with high contextual integrity
 - 12: Generate rationale using EviOmni $r \leftarrow \mathcal{M}_E(\cdot|q_c, P)$
 - 13: Generate rational evidence using EviOmni $y_e \leftarrow \mathcal{M}_E(\cdot|q_c, P, r)$
 - 14: Append y_a and y_e to the history buffer $y \leftarrow y + y_a + y_e$
 - 15: **end if**
 - 16: **if** $\langle \text{answer} \rangle$ tag in y_a **then**
 - 17: Extract answer o_t from y_a
 - 18: Assign o_t to the final answer $o \leftarrow o_t$
 - 19: **break**
 - 20: **end if**
 - 21: Increment action count $t \leftarrow t + 1$
 - 22: **end while**
 - 23: **return** o
-