CTRL-RAG: CONTRASTIVE LIKELIHOOD REWARD-BASED REINFORCEMENT LEARNING FOR CONTEXT-FAITHFUL RAG MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

With the growing use of Retrieval-Augmented Generation (RAG), improving contextual faithfulness and retrieval-grounded reasoning in large language models (LLMs) has become increasingly important. Existing RAG-oriented reinforcement learning (RL) methods primarily rely on external reward signals, such as correctness or LLM-based judges, which often provide indirect or noisy supervision for document faithfulness. Meanwhile, internal self-reward signals based on model confidence lack explicit grounding in retrieved evidence and may reinforce spurious behaviors during training. To address these limitations, we propose CTRL-RAG, an internal–external hybrid reward framework centered on a Contrastive Likelihood Reward (CLR). CLR measures the log-likelihood difference between responses generated with and without supporting evidence, directly estimating the contribution of retrieved documents to generation. This encourages the model to identify and utilize relevant evidence while suppressing reliance on noisy context. Combined with a correctness reward through a gating mechanism, CTRL-RAG encourages the model to use relevant evidence while treating correctness as a prerequisite. Experiments on single-hop, multi-hop, biomedical, and faithfulness benchmarks show consistent improvements in retrieval-grounded reasoning and contextual faithfulness across both dense and Mixture-of-Experts architectures.

1 INTRODUCTION

As Retrieval-Augmented Generation (RAG) becomes more widely used, improving contextual faithfulness and evidence-grounded reasoning has become increasingly important for Large Language Models (LLMs). This also matters in agentic search: decisions about when to search and which documents to retrieve occur upstream, while the generator must still use the evidence it receives. We study this latter problem with fixed retrieved contexts. Existing approaches typically rely on Supervised Fine-Tuning (SFT) Zhang et al. (2024); Jiao et al. (2025) or Reinforcement Learning (RL) with external rewards, such as answer accuracy, citation quality, or formatting constraints Huang et al. (2025); Wu et al. (2025). More recently, some approaches further incorporate LLM-based judge models trained for sentence-level faithfulness assessment Team et al. (2026), reflecting a broader shift toward incorporating contextual grounding signals into reward design.

Limitations of External Reward Signals. In domains with verifiable outcomes, external rewards can be relatively direct: a mathematical answer can be checked against a known result, and code can be executed against tests. In source-grounded generation, an answer can be correct while deriving from the model's prior knowledge rather than the supplied documents. Thus answer accuracy alone does not establish evidence use, and citation markers alone do not establish that cited text supports a claim. An LLM judge can compare a response with retrieved passages, but cannot observe how strongly the generator relied on those passages when producing it. The judge must infer grounding from surface cues such as wording and citations, which may be misleading, especially for multi-hop or context-sensitive evidence. Even strong evaluators (e.g., GPT-5.4 in our reproduced judge baselines) therefore provide an imperfect, potentially noisy reward for document reliance.

Limitations of Internal Reward Signals. Internal reward signals based on model uncertainty, such as entropy or perplexity, avoid dependence on external supervision but introduce a different set of

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

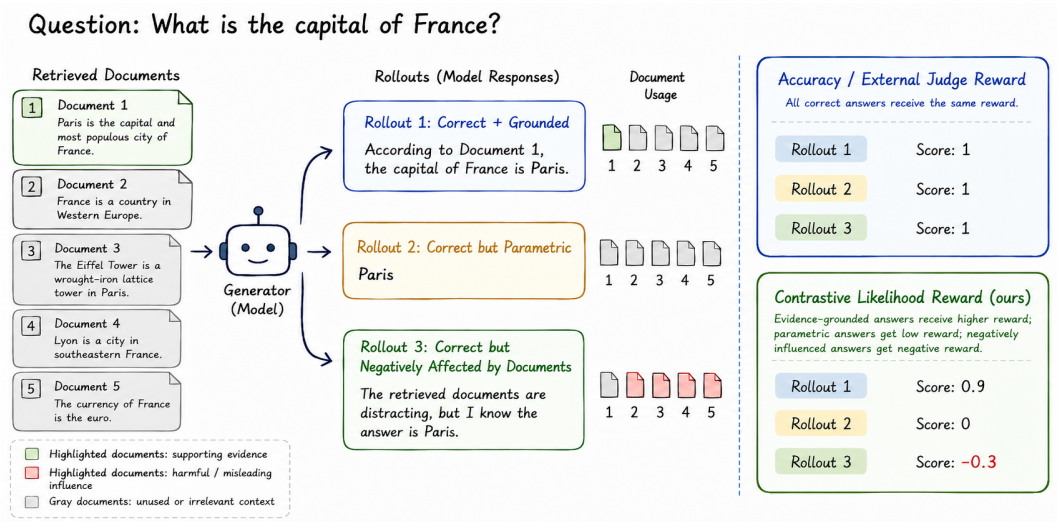


Figure 1: The comparison between the traditional RAG RL methods (external judge signals) and our Contrastive Likelihood Rewards (CLR). All rollouts are generated using the same input; however, outcomes may vary depending on the extent to which the model utilizes the retrieved documents. A higher positive score indicates a greater degree of document utilization by the model, whereas a larger negative score indicates that the documents pose a greater burden on the model.

challenges. Without grounding in external evidence, these signals may reinforce spurious model confidence and lead to reward drift during training. In addition, they are not designed to capture whether the model's confidence is actually grounded in the retrieved evidence, which is central to RAG.

These limitations highlight a central challenge in RAG reward design: the reward should be sensitive to whether retrieved evidence contributes to answer generation, while remaining useful for stable optimization. Neither outcome correctness nor model confidence alone establishes this dependence on the supplied documents.

To address this challenge, we introduce a reward formulation based on contrastive likelihood estimation. Our approach combines two complementary sources of supervision: an intrinsic signal derived from the model's likelihood over generated answers, and an extrinsic grounding signal based on pre-identified supporting documents. Specifically, we compute the difference between the likelihood of a generated answer conditioned on the full retrieved context and that conditioned on an evidence-ablated context, where the supporting documents are removed, and only the remaining documents are retained. This contrastive formulation directly measures the dependence of generation on supporting evidence, enabling the model to distinguish useful evidence from distractors while maintaining stable optimization signals.

The resulting reward can be used independently or combined with correctness-based supervision. In the latter case, correctness serves as a prerequisite, while the contrastive reward encourages the model to identify supporting evidence within noisy retrieval results and ground its answers in that evidence.

Our contributions are summarized as follows:

- **Contrastive Likelihood Reward Formulation for RAG.** We propose **CTRL-RAG**, a contrastive reward formulation for training faithfulness and contextual reasoning abilities of LLMs. Our method leverages likelihood differences under full and evidence-ablated contexts to explicitly estimate the contribution of retrieved documents to answer generation. By combining intrinsic model confidence with externally identified supporting evidence, CTRL-RAG provides a direct and model-aware signal for optimizing contextual faithfulness.
- **Effective Across Architectures and Inference Modes.** We demonstrate consistent improvements across both dense and Mixture-of-Experts (MoE) models, as well as under reasoning

and non-reasoning inference modes. These results show that the proposed reward design generalizes across model architectures and decoding settings.

2 Related Work

Post-training for RAG Generators. RAG and related systems improve factuality by augmenting LLMs with external knowledge Zhang et al. (2023); Guu et al. (2020); Lewis et al. (2021), placing increasing emphasis on context-faithful generation. SFT-based methods such as Self-RAG Asai et al. (2023) and HIRAG Jiao et al. (2025) enhance evidence utilization via structured supervision. RL-based approaches, including RAG-RL Huang et al. (2025) and PA-RAG Wu et al. (2025), further optimize behaviors such as citation accuracy and document preference, while recent systems such as Kimi K2 Team et al. (2026) incorporate LLM-based judges for sentence-level faithfulness. Complementary benchmarks (e.g., MuSiQue Trivedi et al. (2022), PRGB Tan et al. (2025), FaithfulEval Ming et al. (2025)) evaluate reasoning and faithfulness under controlled or counterfactual settings. Despite these advances, reward design for aligning generation with retrieved evidence remains an open challenge.

External and Internal Reward Signals. External reward signals, including rule-based metrics and LLM-based judges, are widely used in RL for RAG but often suffer from reward hacking, high cost, or noisy supervision Guo et al. (2025); Zhang et al. (2025a); Liu et al. (2025a). While such methods can incorporate faithfulness objectives, their signals are typically indirect or unreliable at evidence grounding levels. In contrast, internal signals—such as entropy, self-certainty, or self-rewarding mechanisms Zhang et al. (2025b); Liu et al. (2025b); Zhao et al. (2025); Yuan et al. (2025)—provide stable and low-cost alternatives but lack explicit grounding in external evidence. This disconnect motivates reward designs that can jointly capture model confidence and evidence utilization, which our contrastive likelihood formulation explicitly addresses.

3 Preliminaries

We introduce the RAG generation setup, commonly used reward signals, and the optimization algorithm.

3.1 Rewards for RAG Generators

Given a query q and retrieved documents $\mathcal{D} = \{d_1, \ldots, d_k\}$, a RAG generator parameterized by θ defines a policy π_θ:

$$P(y|q, \mathcal{D}) = \prod_{t=1}^{T} P(y_t \mid y_{<t}, q, \mathcal{D}), \tag{1}$$

where $y = (y_1, \ldots, y_T)$ is the generated response of length T.

In practice, reward design often relies on heuristic signals for grounding and correctness. We consider two commonly used rewards: a **citation reward** R_{cite} that encourages referencing supporting documents $\mathcal{D}^+ \subseteq \mathcal{D}$ via citation markers c_i, and a **correctness reward** R_{acc} that measures agreement with a ground-truth answer y^*:

$$R_{cite} = \frac{1}{|\mathcal{D}^+|} \sum_{d_i \in \mathcal{D}^+} \mathbb{I}(c_i \in y), \quad R_{acc} = \mathbb{I}(y^* \subseteq y), \tag{2}$$

where $\mathbb{I}(\cdot)$ is the indicator function.

The total reward is typically defined as:

$$R_{\text{total}} = \alpha R_{cite} + \beta R_{acc} - \eta \cdot \text{Cost}(y), \tag{3}$$

where α, β are scaling factors and $\text{Cost}(y)$ penalizes sequence length or repetition. While effective in practice, such rewards provide only indirect supervision for contextual grounding.

3.2 GROUP RELATIVE POLICY OPTIMIZATION (GRPO)

We adopt Group Relative Policy Optimization (GRPO) to optimize the RAG generator. For each query q, the model samples a group of responses $\{o_1, o_2, \ldots, o_G\}$.

The objective is formulated as:

$$\mathbb{J}(\theta) = \mathbb{E}_{q,\{o_i\}} \left[\frac{1}{G} \sum_{i=1}^{G} \left(\frac{1}{|o_i|} \sum_{t=1}^{|o_i|} r_{i,t}^{clip}(\theta) - \beta \mathbb{D}_{\text{KL}} \right) \right], \tag{4}$$

$$r_{i,t}^{clip}(\theta) = \min \left(r_{i,t}(\theta)\hat{A}_{i,t}, \text{clip}(r_{i,t}(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_{i,t} \right), \tag{5}$$

where $\mathbb{D}_{\text{KL}} = KL(\pi_\theta \| \pi_{ref})$ and

$$r_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t} \mid q, o_{i,<t})}, \quad \hat{A}_{i,t} = \frac{R_{i,t} - \mu_R}{\sigma_R + \epsilon}, \tag{6}$$

where μ_R and σ_R are the mean and standard deviation of rewards within the group.

GRPO performs stable optimization via relative reward comparisons and is compatible with the reward formulation introduced in the next section.

4 METHODOLOGY

Existing reward designs for retrieval-augmented generation (RAG) often rely on external supervision signals, such as answer accuracy or LLM-based judges, which can be indirect or noisy for contextual faithfulness. In contrast, internal reward signals based on model confidence lack explicit grounding in retrieved evidence. To address these limitations, we propose a hybrid reward mechanism that combines internal likelihood-based confidence with external evidence supervision. Specifically, we use contrastive likelihood estimation to measure how much supporting documents contribute to the likelihood of a generated response, encouraging more faithful and evidence-grounded generation. Overall Methodology Figure is shown in Figure 4.

4.1 EVIDENTIAL CONTRIBUTION

To quantify how much a generated response is grounded in the provided documents, we introduce a metric termed **Evidential Contribution**. This metric assesses the contribution of supporting documents to the model's confidence in generating a response. We use the term Evidential Contribution to avoid confusion with classical Information Gain in information theory and decision tree learning Quinlan (1986).

Let q denote the question and $\mathcal{D} = \{d_1, d_2, \ldots, d_n\}$ be a set of n retrieved documents. We partition $\mathcal{D}$ into a subset of supporting documents, $\mathcal{D}^+$, and a subset of noisy or irrelevant documents, $\mathcal{D}^- = \mathcal{D} \setminus \mathcal{D}^+$. In practice, we obtain $\mathcal{D}^+$ using a hybrid annotation procedure that combines rule-based (Answer in Documents) annotation with LLM-based judgment. Given a generated sequence $y = (y_1, \ldots, y_T)$ of length T, we first define the sequence-level log-likelihood under the full context as:

$$S(y \mid \mathcal{D}) = \sum_{t=1}^{T} \log P(y_t \mid y_{<t}, q, \mathcal{D}) \tag{7}$$

To measure the total dependence of the response on the provided evidence, we contrast the full-context likelihood with a counterfactual baseline where all supporting documents are removed. We define this counterfactual score, $S^-(y \mid \mathcal{D}^-)$, as:

$$S^-(y \mid \mathcal{D}^-) = \sum_{t=1}^{T} \log P(y_t \mid y_{<t}, q, \mathcal{D} \setminus \mathcal{D}^+) \tag{8}$$

Based on these scores, we define the Evidential Contribution $\mathcal{E}(y)$ as the reduction in log-likelihood when these supporting documents are removed:

$$\mathcal{E}(y) = S(y \mid \mathcal{D}) - S^-(y \mid \mathcal{D}^-) \tag{9}$$

A higher $\mathcal{E}(y)$ indicates that the response y is strongly reliant on the supporting documents, thereby reflecting stronger document grounding and faithfulness.

4.2 CONTRASTIVE LIKELIHOOD REWARD

Our sequence-level metric, Evidential Contribution $\mathcal{E}(y)$, is an aggregation of token-level effects. To construct a robust reward signal, we first decompose $\mathcal{E}(y)$ into its token-level components. We define the token-level Evidential Contribution $\epsilon(y_t)$ as:

$$\epsilon(y_t) = \log P(y_t \mid y_{<t}, q, \mathcal{D}) - \log P(y_t \mid y_{<t}, q, \mathcal{D} \setminus \mathcal{D}^+) \tag{10}$$

By construction, the sequence-level contribution is the summation of these token-level scores: $\mathcal{E}(y) = \sum_{t=1}^{T} \epsilon(y_t)$. However, utilizing $\mathcal{E}(y)$ directly as a reward signal poses two practical challenges:

- **Length Bias:** As an accumulation of token-level scores, $\mathcal{E}(y)$ is inherently biased towards longer sequences. A model could exploit this by generating verbose or repetitive content from the supporting documents to artificially inflate the reward, leading to degeneration and training instability.
- **Negative Contributions:** In practice, token-level contributions $\epsilon(y_t)$ are rarely exactly zero, and sequence-level scores $\mathcal{E}(y)$ can also become non-positive. Such negative values can arise from various sources, including document noise, context-length shifts after document removal, generation randomness, and conflicts with the model's parametric knowledge. Treating these cases as strong negative supervision may be unreliable and can distract the model from learning from high-quality positive examples.

To mitigate these issues, we propose the **Contrastive Likelihood Reward** (R_{CLR}), which incorporates length normalization and a significance threshold τ:

$$R_{CLR}(y) = \frac{\mathcal{E}(y) \cdot \mathbb{I}(\mathcal{E}(y) > \tau)}{\sqrt{T}} \tag{11}$$

First, normalization by $\sqrt{T}$ mitigates length bias. We use $\sqrt{T}$ as a sub-linear normalization term to balance two competing effects: without normalization, the reward is biased toward overly long responses and may destabilize training, whereas normalization by T penalizes length too aggressively and weakens the reward for genuinely informative long answers. Empirically, $\sqrt{T}$ yields the best trade-off, and Section 5.3 shows that it outperforms linear normalization by T.

Second, the indicator function $\mathbb{I}(\mathcal{E}(y) > \tau)$ acts as a significance filter over sequence-level contributions. Since non-positive values of $\mathcal{E}(y)$ often correspond to unreliable or hard-to-interpret cases, we set $\tau = 0$ so that such responses receive zero reward during training. This prevents the model from being optimized on ambiguous negative signals and concentrates learning on responses with clear positive evidential support. Empirically, this simple thresholding strategy yields the best performance in our experiments. We provide a detailed analysis of this choice in Appendix A.6.

4.3 HYBRID REWARD INTEGRATION

While the Contrastive Likelihood Reward (R_{CLR}) encourages document-grounded generation, it does not guarantee factual correctness. This may lead to *faithfully wrong* responses, i.e., answers that are grounded in the retrieved documents but still incorrect. To encourage responses that are both faithful and correct, we combine R_{CLR} with an accuracy-based reward, R_{acc}.

Since the two rewards have different scales, we first normalize R_{CLR} within each candidate group using min-max scaling:

$$R'_{CLR}(y_i) = \frac{R_{CLR}(y_i) - \min_{y \in \mathcal{Y}} R_{CLR}(y)}{\max_{y \in \mathcal{Y}} R_{CLR}(y) - \min_{y \in \mathcal{Y}} R_{CLR}(y) + \epsilon}, \tag{12}$$

where $\mathcal{Y}$ is the set of candidate responses for the same prompt.

A simple weighted sum does not always produce the desired behavior, as the model may prefer optimizing the easier reward signal. Since correctness should be a prerequisite in RAG, we instead adopt a gating formulation inspired by GDPO Liu et al. (2026):

$$R_{\text{hybrid}} = R'_{CLR} \cdot R_{\text{acc}}, \tag{13}$$

where $R_{\text{acc}} \in \{0, 1\}$. Under this design, incorrect responses receive zero reward, while evidential grounding determines the reward magnitude among correct answers. We use this formulation in our main experiments and compare it with additive variants in Section 5.3.

We provide token-level visualization and interpretability analysis of CLR in Appendix B.

5 EXPERIMENT

Datasets. To evaluate the document utilization and contextual faithfulness of our proposed reward, we conduct experiments on RAGQALeaderboard Zhehao Tan, Yihan Jiao, Dan Yang (2026) and PRGB Tan et al. (2025). RAGQALeaderboard contains both multi-hop QA tasks, including 2Wiki Ho et al. (2020), HotpotQA Yang et al. (2018), and MuSiQue Trivedi et al. (2022), as well as single-hop QA tasks, including TriviaQA Joshi et al. (2017), PopQA Mallen et al. (2023), and PubMedQA Jin et al. (2019). PRGB isolates parametric knowledge through placeholder substitution, enabling a more direct evaluation of retrieval faithfulness. We report answer accuracy as the primary evaluation metric for all benchmarks.

Training and Evaluation. We train all RL-based models using GRPO within the MS-Swift framework. The source data pool comprises HotpotQA, 2Wiki, and MuSiQue. Before RL training, we construct SFT data from HotpotQA and 2Wiki, selecting samples according to their $pass@8$ accuracy to obtain a shared cold-start model. All RL methods start from this checkpoint and draw candidates from the same remaining raw pool across the three datasets. Each method then applies reward-specific sample selection to obtain groups with informative reward variation; the selected subsets need not be identical. Appendix A.1 describes this distinction. During evaluation, we follow the official protocols of RAGQALeaderboard and PRGB. We use 30 retrieved documents for RAGQALeaderboard and 10 documents for PRGB. Additional implementation details are provided in Appendix A and Appendix C.

Baselines. We use Qwen3-30B-A3B-Base as the primary backbone model and Qwen3-8B as a secondary backbone for supplementary experiments (results are provided in Appendix D). Our approach is compared against two groups of baselines. The first group consists of RL baselines initialized from the same SFT checkpoint and sharing the raw candidate pool, with reward-specific selection before training. Thus these comparisons evaluate the practical reward-and-selection pipelines, rather than isolating the reward formula under an identical final training subset. The second group includes publicly available instruction-tuned models as reference points for overall performance.

5.1 MAIN EXPERIMENTS

Table 1 summarizes the main experimental results. For RL comparisons using the same initialization and raw candidate pool, we evaluate several reward baselines. The accuracy reward R_{acc} directly optimizes answer correctness, while the citation reward R_{cite} encourages explicit document citation. We also include R_{total}, a weighted reward formulation following prior RAG-RL-style training, where we set $\alpha = 0.5$, $\beta = 0.5$, and $\eta = 0$ according to Eq. 3. In addition, we compare against LLM-as-a-Judge rewards: $R_{\text{J-Acc}}$ uses a strong LLM(GPT-5.4) to judge answer correctness, and $R_{\text{J-Faith}}$ uses the same LLM to approximate the sentence-level faithfulness supervision adopted by K2-style training. The final RL subsets are selected separately for each reward as described in Appendix A.1.

Across multi-hop reasoning benchmarks, both R_{CLR} and R_{hybrid} outperform conventional reward designs such as R_{acc} and R_{cite}, and achieve stronger or more consistent performance than R_{total} and the LLM-as-a-Judge baselines on most retrieval-intensive tasks. These results suggest that likelihood-based contrastive rewards provide a more direct and model-aware signal for training RAG generators, especially when the model needs to identify and rely on supporting evidence from retrieved documents. Moreover, integrating R_{CLR} with the accuracy reward further improves overall performance, particularly in the no-think setting and across multi-hop QA benchmarks. This indicates that contextual grounding and answer correctness are complementary objectives, and that the benefit of CLR is not limited to explicit reasoning-mode generation.

On the PRGB benchmark, which isolates parametric knowledge and directly evaluates contextual faithfulness, R_{CLR} achieves the strongest performance among all reward variants. This result indicates that the proposed reward encourages the model to rely more heavily on retrieved evidence

Table 1: **Main results on Qwen3 models.** The best and second-best results within each evaluation setting are marked in **bold** and underline, respectively. The overall best result in each column is highlighted in green. $R_{\text{J-Acc}}$ denotes the LLM-as-a-Judge accuracy reward, $R_{\text{J-Faith}}$ denotes the LLM-as-a-Judge faithfulness reward, and R_{total} denotes the weighted combination of accuracy and citation as defined in Eq. 3. AVG denotes the arithmetic mean across all benchmarks. Results on Qwen3-8B are deferred to Appendix D.

Method	AVG	Multi-hop			Single-hop		Faithfulness	Biomedical
		2Wiki	Hotpot	MuSiQue	TQA	PQA	PRGB	PubMed
Qwen3-30B-A3B (think)								
SFT	80.3	81.1	79.3	64.9	92.0	92.2	84.5	68.2
SFT+RL(R_{acc})	83.1	85.4	82.3	70.6	93.2	93.8	85.9	70.4
SFT+RL(R_{cite})	82.4	83.4	80.4	69.5	93.3	94.0	85.2	70.8
SFT+RL(R_{total})	84.2	86.5	82.5	73.2	93.7	**95.4**	86.4	71.8
SFT+RL($R_{\text{J-Acc}}$)	81.9	84.2	80.7	67.9	93.2	92.9	84.1	70.1
SFT+RL($R_{\text{J-Faith}}$)	81.5	83.3	80.3	67.6	92.0	92.4	85.2	70.0
SFT+RL($R_{\text{J-Faith+Acc}}$)	83.5	85.2	82.2	72.5	93.2	94.0	85.9	71.8
SFT+RL(R_{CLR})	**85.0**	86.9	82.5	72.7	93.8	94.7	**89.5**	**74.6**
SFT+RL(R_{hybrid})	84.9	**87.1**	**83.2**	**73.9**	**94.0**	94.9	88.9	72.6
Qwen3-30B-A3B (no think)								
SFT	77.6	78.7	78.3	63.4	92.2	92.3	73.8	64.2
SFT+RL(R_{acc})	80.0	80.5	80.8	66.2	92.5	93.2	76.7	70.4
SFT+RL(R_{cite})	78.9	79.0	79.2	65.3	93.0	91.8	75.5	68.4
SFT+RL(R_{total})	81.2	81.1	82.2	69.1	93.4	92.5	78.3	71.6
SFT+RL($R_{\text{J-Acc}}$)	79.2	79.7	79.7	63.8	92.8	93.7	74.4	70.1
SFT+RL($R_{\text{J-Faith}}$)	78.6	78.7	78.9	62.4	91.2	92.5	75.4	71.2
SFT+RL($R_{\text{J-Faith+Acc}}$)	80.4	80.4	81.3	67.0	92.6	92.8	75.8	73.0
SFT+RL(R_{CLR})	82.6	**83.0**	83.7	69.6	92.9	93.7	78.5	76.8
SFT+RL(R_{hybrid})	**83.5**	82.9	**84.1**	**70.6**	**94.8**	**94.6**	**79.5**	**77.9**
Open-source Post-trained Models								
Qwen3-8B	61.9	71.4	71.0	44.5	90.2	74.0	64.7	17.6
Qwen3-14B	63.2	68.8	69.4	37.6	86.8	72.7	65.1	42.0
Qwen3-32B	69.3	77.9	73.2	48.6	89.4	76.4	68.3	51.6
Kimi-K2-Instruct	75.8	82.3	76.3	**66.5**	92.5	86.3	71.5	**55.2**
Qwen3-30B-A3B-Instruct-2507	75.5	81.4	81.9	55.8	**94.2**	86.7	75.9	52.4
Qwen3-235B-A22B-Instruct-2507	**78.0**	**84.9**	**82.8**	65.3	93.8	**87.6**	**78.2**	53.2

rather than latent parametric memorization. Notably, the gain over LLM-as-a-Judge faithfulness rewards further suggests that token-level contrastive likelihood can capture grounding signals that are difficult to recover from sentence-level judge feedback alone. Finally, our method achieves performance competitive with strong open-source post-trained models, particularly on multi-hop reasoning and faithfulness-oriented benchmarks. This is notable because our models are trained from a base checkpoint with a controlled post-training setup, while the open-source instruction-tuned models may benefit from substantially broader and less transparent post-training data.

5.2 Experiment Analysis

Faithfulness Analysis. Beyond PRGB, we further evaluate document reliance on RAGQALeaderboard using the following metric:

$$RR_\theta = \text{Acc}_\theta(Q, D) - \text{Acc}_\theta(Q),$$

where $\text{Acc}_\theta(Q, D)$ denotes the accuracy with retrieved documents and $\text{Acc}_\theta(Q)$ denotes the accuracy without retrieval context. Intuitively, RR_θ measures the performance gain obtained from external documents.

As shown in Figure 2, $\text{Acc}_\theta(Q)$ remains relatively stable throughout training, while RR_θ increases steadily. This trend suggests that the performance improvement primarily originates from improved document utilization rather than changes in the model's retrieval-free behavior. Correspondingly, $\text{Acc}_\theta(Q, D)$ also improves monotonically during training.

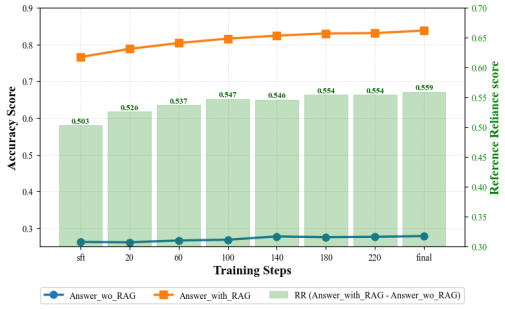


Figure 2: Reference Reliance score and retrieval-based accuracy during training.

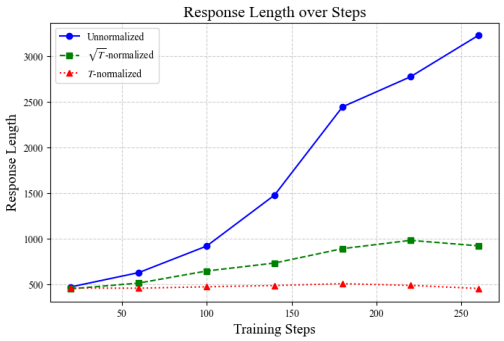


Figure 3: Response length under different normalization strategies.

For the final model, we obtain

$$\mathrm{Acc}_\theta(Q, D) = 0.838, \quad \mathrm{Acc}_\theta(Q) = 0.279, \quad RR_\theta = 0.559,$$

compared with Qwen3-30B-A3B-Instruct-2507:

$$\mathrm{Acc}_\theta(Q, D) = 0.720, \quad \mathrm{Acc}_\theta(Q) = 0.291, \quad RR_\theta = 0.429.$$

These results indicate that our method substantially improves retrieval-conditioned reasoning capability and document utilization. Additional analysis on likelihood dynamics and perplexity evolution is provided in Appendix A.3.

5.3 Ablation Experiments

Impact of Length Normalization. Since $\mathcal{E}(y)$ accumulates token-level contributions, the reward naturally increases with response length. To mitigate this bias, we investigate different normalization strategies.

As shown in Figure 3, without normalization, the model rapidly learns to generate excessively verbose and repetitive outputs, frequently reaching the maximum generation limit. In contrast, directly normalizing by T overly suppresses reward growth and prevents effective learning. Normalization by $\sqrt{T}$ achieves a more favorable balance: the response length initially increases and then gradually stabilizes. This observation empirically supports our choice of sub-linear normalization.

Impact of Evidence Attribution Strategies. To analyze the design choices of CLR, we compare different evidence removal and aggregation strategies.

We compare coarse-grained evidence removal with leave-one-out variants. In contrast to the intuition that fine-grained document-level attribution may provide a more precise training signal, removing all supporting documents simultaneously achieves stronger overall performance in our experiments. This suggests that the global likelihood contrast between full evidence and evidence-ablated contexts already provides a sufficiently informative and stable grounding signal for RL optimization.

We further compare different aggregation and filtering strategies for leave-one-out variants. Minimum-based aggregation and average pooling achieve comparable performance, with only moderate differences across benchmarks. Introducing the threshold operation generally improves performance, indicating that filtering weak or noisy contribution signals helps stabilize training. Overall, these results suggest that while fine-grained attribution can provide useful diagnostic signals, the simpler remove-all formulation is more effective as the main reward design.

Impact of Integration with Accuracy. Building on the evidence attribution mechanism above, we further integrate an answer correctness reward. We compare additive fusion and multiplicative gating between the CLR reward and the accuracy reward.

As shown in the lower part of Table 2, the multiplicative formulation achieves stronger overall performance on RAGQA, while additive fusion achieves competitive performance on PRGB. Overall,

Table 2: **Detailed ablation study results.** "Remove-All" denotes removing all supporting documents simultaneously when computing the contrastive score. LOO-Min and LOO-Avg denote minimum and average aggregation over leave-one-out scores, respectively. τ denotes the significance threshold in Eq. 11. $+$Acc and $\times$Acc denote additive fusion and multiplicative gating with the accuracy reward.

Variant	AVG	MuSiQue	Hotpot	2Wiki	PQA	TQA	PubMed	PRGB
SFT	77.6	63.4	78.7	78.3	92.2	92.3	64.2	73.8
Evidence removal and aggregation strategies								
LOO-Min	81.8	68.2	81.5	81.9	93.2	93.6	75.8	78.3
LOO-Avg	81.7	68.2	81.4	81.9	93.2	93.5	75.6	78.4
Remove-All	82.2	68.8	82.4	83.0	92.9	93.7	76.2	78.4
LOO-Min + τ	82.3	69.0	81.5	82.7	93.5	93.7	76.8	78.6
LOO-Avg + τ	82.2	69.7	81.7	82.7	93.2	94.0	75.2	78.6
Remove-All + τ	82.6	69.6	83.0	83.7	92.9	93.7	76.8	78.5
Integration with accuracy reward								
LOO-Avg + τ + Acc	83.0	70.3	82.2	83.6	93.9	94.3	77.0	79.5
LOO-Avg + $\tau \times$ Acc	82.4	69.3	81.9	82.6	93.7	93.3	77.8	78.2
LOO-Min + τ + Acc	82.9	70.9	82.6	83.9	94.2	93.7	75.2	80.0
LOO-Min + $\tau \times$ Acc	83.1	70.5	81.9	83.0	94.4	94.9	78.0	79.3
Remove-All + τ + Acc	83.2	70.4	82.9	83.8	93.6	94.1	77.5	80.3
Remove-All + $\tau \times$ Acc	83.5	70.6	82.9	84.1	94.8	94.6	77.9	79.5

these results suggest that enforcing correctness as a gating condition can more effectively improve retrieval-based reasoning performance.

6 Limitations

Despite its effectiveness, our framework still has several limitations. First, CTRL-RAG introduces additional computational overhead because CLR requires extra logits computation for likelihood estimation during reward calculation. Compared with conventional scalar reward models, this increases training latency and memory consumption, particularly in long-context generation settings. However, as discussed in Appendix A.4, this overhead can be substantially reduced by using an optimized vLLM-based reward computation backend. Second, our reward design explicitly prioritizes contextual faithfulness to retrieved documents. As a result, when retrieved evidence is factually incorrect or conflicts with the model's parametric knowledge, the framework may discourage otherwise correct responses that are insufficiently grounded in the provided context. More broadly, the current formulation assumes that supporting documents are largely reliable and does not explicitly model retrieval quality or evidence trustworthiness. Future work will explore more efficient likelihood estimation strategies, as well as adaptive reward mechanisms that dynamically balance contextual grounding with parametric knowledge when retrieved evidence is noisy or unreliable.

7 Conclusion

In this work, we present CTRL-RAG, a reinforcement learning reward design for improving contextual faithfulness and retrieval-based reasoning in RAG systems. At its core is the Contrastive Likelihood Reward (CLR), which quantifies the contribution of supporting documents through contrastive likelihood estimation and explicitly encourages evidence-grounded generation. Extensive experiments across both Dense and Mixture-of-Experts architectures show that CLR consistently enhances multi-hop reasoning, document utilization, and contextual faithfulness. Integrating CLR with accuracy-based rewards further improves performance, highlighting the complementary relationship between correctness and grounding. Overall, our results suggest that contrastive likelihood-based rewards provide an effective and scalable supervision signal for RAG models, offering a practical and interpretable approach for training more reliable and context-aware generation.

AI USE STATEMENT

Generative AI tools assisted with editing the English text of this manuscript, including grammar correction and revisions for clarity and academic style, as well as LaTeX formatting. They also assisted with minor edits to the supplementary code and its packaging to remove potentially author-identifying local paths and metadata for double-blind review. The authors take responsibility for the final manuscript and its scientific claims, analyses, and conclusions.

REFERENCES

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection, 2023. URL https://arxiv.org/abs/2310.11511.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645 (8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: Retrieval-augmented language model pre-training, 2020. URL https://arxiv.org/abs/2002.08909.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps, 2020. URL https://arxiv.org/abs/2011.01060.

Jerry Huang, Siddarth Madala, Risham Sidhu, Cheng Niu, Hao Peng, Julia Hockenmaier, and Tong Zhang. Rag-rl: Advancing retrieval-augmented generation via rl and curriculum learning, 2025. URL https://arxiv.org/abs/2503.12759.

YiHan Jiao, ZheHao Tan, Dan Yang, DuoLin Sun, Jie Feng, Yue Shen, Jian Wang, and Peng Wei. Hirag: Hierarchical-thought instruction-tuning retrieval-augmented generation, 2025. URL https://arxiv.org/abs/2507.05714.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W. Cohen, and Xinghua Lu. Pubmedqa: A dataset for biomedical research question answering, 2019. URL https://arxiv.org/abs/1909.06146.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension, 2017. URL https://arxiv.org/abs/1705.03551.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021. URL https://arxiv.org/abs/2005.11401.

Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, Yejin Choi, Jan Kautz, and Pavlo Molchanov. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization, 2026. URL https://arxiv.org/abs/2601.05242.

Shuliang Liu, Xinze Li, Zhenghao Liu, Yukun Yan, Cheng Yang, Zheni Zeng, Zhiyuan Liu, Maosong Sun, and Ge Yu. Judge as a judge: Improving the evaluation of retrieval-augmented generation through the judge-consistency of large language models, 2025a. URL https://arxiv.org/abs/2502.18817.

Wei Liu, Siya Qi, Xinyu Wang, Chen Qian, Yali Du, and Yulan He. Nover: Incentive training for language models via verifier-free reinforcement learning, 2025b. URL https://arxiv.org/abs/2505.16022.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories, 2023. URL https://arxiv.org/abs/2212.10511.

Yifei Ming, Senthil Purushwalkam, Shrey Pandit, Zixuan Ke, Xuan-Phi Nguyen, Caiming Xiong, and Shafiq Joty. Faitheval: Can your language model stay faithful to context, even if "the moon is made of marshmallows", 2025. URL https://arxiv.org/abs/2410.03727.

J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986. doi: 10.1007/BF00116251.

Zhehao Tan, Yihan Jiao, Dan Yang, Lei Liu, Jie Feng, Duolin Sun, Yue Shen, Jian Wang, Peng Wei, and Jinjie Gu. Prgb benchmark: A robust placeholder-assisted algorithm for benchmarking retrieval-augmented generation, 2025. URL https://arxiv.org/abs/2507.22927.

Kimi Team, Yifan Bai, Yiping Bao, Y. Charles, Cheng Chen, Guanduo Chen, Haiting Chen, Huarong Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, Zhuofu Chen, Jialei Cui, Hao Ding, Mengnan Dong, Angang Du, Chenzhuang Du, Dikang Du, Yulun Du, Yu Fan, Yichen Feng, Kelin Fu, Bofei Gao, Chenxiao Gao, Hongcheng Gao, Peizhong Gao, Tong Gao, Yuyao Ge, Shangyi Geng, Qizheng Gu, Xinran Gu, Longyu Guan, Haiqing Guo, Jianhang Guo, Xiaoru Hao, Tianhong He, Weiran He, Wenyang He, Yunjia He, Chao Hong, Hao Hu, Yangyang Hu, Zhenxing Hu, Weixiao Huang, Zhiqi Huang, Zihao Huang, Tao Jiang, Zhejun Jiang, Xinyi Jin, Yongsheng Kang, Guokun Lai, Cheng Li, Fang Li, Haoyang Li, Ming Li, Wentao Li, Yang Li, Yanhao Li, Yiwei Li, Zhaowei Li, Zheming Li, Hongzhan Lin, Xiaohan Lin, Zongyu Lin, Chengyin Liu, Chenyu Liu, Hongzhang Liu, Jingyuan Liu, Junqi Liu, Liang Liu, Shaowei Liu, T. Y. Liu, Tianwei Liu, Weizhou Liu, Yangyang Liu, Yibo Liu, Yiping Liu, Yue Liu, Zhengying Liu, Enzhe Lu, Haoyu Lu, Lijun Lu, Yashuo Luo, Shengling Ma, Xinyu Ma, Yingwei Ma, Shaoguang Mao, Jie Mei, Xin Men, Yibo Miao, Siyuan Pan, Yebo Peng, Ruoyu Qin, Zeyu Qin, Bowen Qu, Zeyu Shang, Lidong Shi, Shengyuan Shi, Feifan Song, Jianlin Su, Zhengyuan Su, Lin Sui, Xinjie Sun, Flood Sung, Yunpeng Tai, Heyi Tang, Jiawen Tao, Qifeng Teng, Chaoran Tian, Chensi Wang, Dinglu Wang, Feng Wang, Hailong Wang, Haiming Wang, Jianzhou Wang, Jiaxing Wang, Jinhong Wang, Shengjie Wang, Shuyi Wang, Si Wang, Xinyuan Wang, Yao Wang, Yejie Wang, Yiqin Wang, Yuxin Wang, Yuzhi Wang, Zhaoji Wang, Zhengtao Wang, Zhengtao Wang, Zhexu Wang, Chu Wei, Qianqian Wei, Haoning Wu, Wenhao Wu, Xingzhe Wu, Yuxin Wu, Chenjun Xiao, Jin Xie, Xiaotong Xie, Weimin Xiong, Boyu Xu, Jinjing Xu, L. H. Xu, Lin

Xu, Suting Xu, Weixin Xu, Xinran Xu, Yangchuan Xu, Ziyao Xu, Jing Xu, Jing Xu, Junjie Yan, Yuzi Yan, Hao Yang, Xiaofei Yang, Yi Yang, Ying Yang, Zhen Yang, Zhilin Yang, Zonghan Yang, Haotian Yao, Xingcheng Yao, Wenjie Ye, Zhuorui Ye, Bohong Yin, Longhui Yu, Enming Yuan, Hongbang Yuan, Mengjie Yuan, Siyu Yuan, Haobing Zhan, Dehao Zhang, Hao Zhang, Wanlu Zhang, Xiaobin Zhang, Yadong Zhang, Yangkun Zhang, Yichi Zhang, Yizhi Zhang, Yongting Zhang, Yu Zhang, Yutao Zhang, Yutong Zhang, Zheng Zhang, Haotian Zhao, Yikai Zhao, Zijia Zhao, Huabin Zheng, Shaojie Zheng, Longguang Zhong, Jianren Zhou, Xinyu Zhou, Zaida Zhou, Jinguo Zhu, Zhen Zhu, Weiyu Zhuang, and Xinxing Zu. Kimi k2: Open agentic intelligence, 2026. URL https://arxiv.org/abs/2507.20534.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop questions via single-hop question composition, 2022. URL https://arxiv.org/abs/2108.00573.

Jiayi Wu, Hengyi Cai, Lingyong Yan, Hao Sun, Xiang Li, Shuaiqiang Wang, Dawei Yin, and Ming Gao. PA-RAG: RAG alignment via multi-perspective preference optimization. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 9091–9112, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.459. URL https://aclanthology.org/2025.naacl-long.459/.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering, 2018. URL https://arxiv.org/abs/1809.09600.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models, 2025. URL https://arxiv.org/abs/2401.10020.

Kaiyan Zhang, Yuxin Zuo, Bingxiang He, Youbang Sun, Runze Liu, Che Jiang, Yuchen Fan, Kai Tian, Guoli Jia, Pengfei Li, Yu Fu, Xingtai Lv, Yuchen Zhang, Sihang Zeng, Shang Qu, Haozhan Li, Shijie Wang, Yuru Wang, Xinwei Long, Fangfu Liu, Xiang Xu, Jiaze Ma, Xuekai Zhu, Ermo Hua, Yihao Liu, Zonglin Li, Huayu Chen, Xiaoye Qu, Yafu Li, Weize Chen, Zhenzhao Yuan, Junqi Gao, Dong Li, Zhiyuan Ma, Ganqu Cui, Zhiyuan Liu, Biqing Qi, Ning Ding, and Bowen Zhou. A survey of reinforcement learning for large reasoning models, 2025a. URL https://arxiv.org/abs/2509.08827.

Qingyang Zhang, Haitao Wu, Changqing Zhang, Peilin Zhao, and Yatao Bian. Right question is already half the answer: Fully unsupervised llm reasoning incentivization, 2025b. URL https://arxiv.org/abs/2504.05812.

Tianjun Zhang, Shishir G. Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E. Gonzalez. Raft: Adapting language model to domain specific rag, 2024. URL https://arxiv.org/abs/2403.10131.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. Siren's song in the ai ocean: A survey on hallucination in large language models, 2023. URL https://arxiv.org/abs/2309.01219.

Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards, 2025. URL https://arxiv.org/abs/2505.19590.

Zhehao Tan, Yihan Jiao, Dan Yang. RagQALeaderboard: A Comprehensive Leaderboard for RAG-based Medical Question Answering, January 2026. URL https://github.com/AQ-MedAI/RagQALeaderboard.

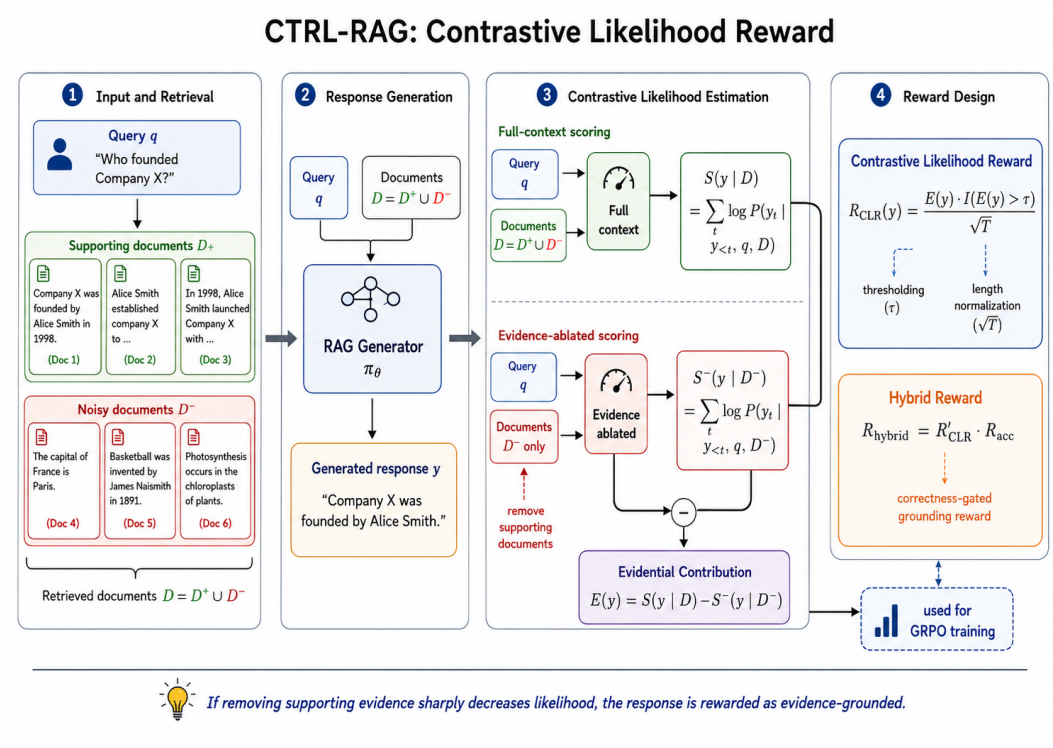


Figure 4: Methodology of CTRL-RAG

A Training Settings

To evaluate the effectiveness of the proposed reward mechanism, we conduct experiments using two medium-scale base policy models with distinct architectures: **Qwen3-30B-A3B-Base**, a Mixture-of-Experts (MoE) model, and **Qwen3-8B-Base**, a dense model. The training process follows a two-stage pipeline. We first perform supervised fine-tuning (SFT) to obtain a cold-start model, and then conduct reinforcement learning (RL) with GRPO. During RL, we evaluate both the standalone contrastive likelihood reward, denoted as **CLR-only**, and the hybrid reward that combines CLR with an accuracy-based reward. We evaluate the resulting models under both reasoning and non-reasoning inference modes.

A.1 Dataset Composition and Selection

Our source data pool comprises HotpotQA, 2Wiki, and MuSiQue. For SFT, we draw from HotpotQA and 2Wiki and select samples based on $pass@8$ accuracy. The resulting SFT dataset contains 74,109 samples, of which 80% have $pass@8 = 1.0$ and 20% have $pass@8 \in [0.5, 0.875]$. After excluding the SFT examples, we form an approximately 10,000-sample raw RL candidate pool from the remaining data across all three datasets: 90% have $pass@8 \in [0.1, 0.875]$ and 10% have $pass@8 = 1.0$. Every RL reward method starts from this same raw pool, but applies its own reward-based sample selection before training.

This selection matters for Group Relative Policy Optimization (GRPO), which learns from differences in reward among responses to the same prompt. In a preliminary experiment, applying one common filtered subset to all reward methods produced models whose performance remained close to the shared SFT backbone, with little gain from RL. A prompt with strongly differentiated CLR scores can still yield an accuracy reward of 1 for every sampled response; its within-group accuracy-reward variance is then zero, so it provides no relative accuracy advantage to GRPO. Conversely, a subset selected for accuracy-reward variation need not provide useful CLR variation. We therefore select informative prompts separately for each reward rather than requiring identical final subsets. For CLR training, we additionally retain instances whose calculated log-probabilities have standard deviation

greater than 10. This design keeps the raw data source and initialization shared, while allowing the final training samples to differ across reward methods.

A.2 Implementation Details

We implement our training framework using MS-Swift, with a customized reward service modified to support entropy regularization. The RL stage uses a group size of 8, a learning rate of 1×10^{-6}, and an effective batch size of 1024. Training is distributed across a heterogeneous cluster of $32 \times$ H800 and $8 \times$ H800 GPUs. We use Tensor Parallelism (TP=4) and Pipeline Parallelism (PP=4) to support rollout generation and optimization. The reward curve during training is shown in Figure 5.

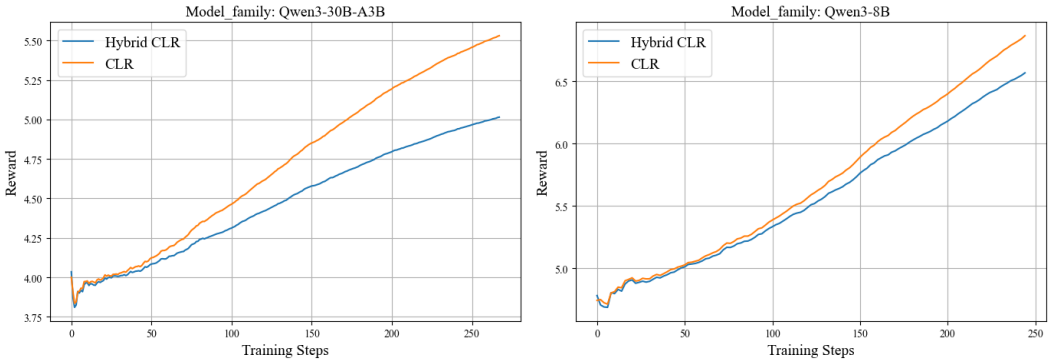


Figure 5: Reward curve during RL training.

A.3 Perplexity Analysis

The proposed R_{CLR} is constructed from two likelihood scores: the full-context score $S(y \mid \mathcal{D})$ and the evidence-ablated score $S^{-}(y \mid \mathcal{D}^{-})$, where all supporting documents are removed from the context. To better understand the training dynamics, we track their corresponding perplexities:

$$PPL(\cdot) = \exp\left(-\frac{S(\cdot)}{T}\right),$$

where T denotes the response length.

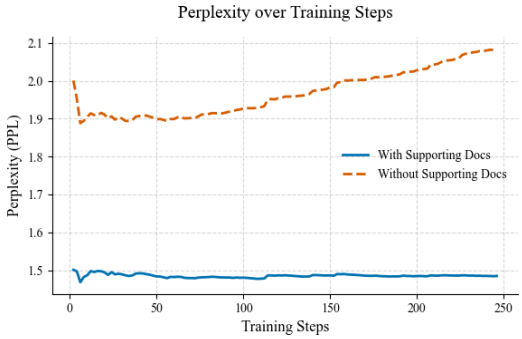


Figure 6: Evolution of $PPL(y \mid \mathcal{D})$ and $PPL^{-}(y \mid \mathcal{D})$ during training.

As illustrated in Figure 6, $PPL(y \mid \mathcal{D})$ stabilizes after an initial decrease, while $PPL^{-}(y \mid \mathcal{D})$ gradually increases throughout training. This indicates that the model becomes increasingly sensitive to the removal of supporting evidence, suggesting stronger dependence on retrieved documents during generation.

A.4 Training Overhead and Optimization

The proposed contrastive likelihood reward introduces additional computational overhead during RL training, mainly due to the need for multiple likelihood evaluations under different document contexts. In our current implementation based on the MS-Swift rollout server framework, reward computation takes approximately 8 seconds per rollout, making it one of the major bottlenecks in the training pipeline.

To assess the practical efficiency of this reward computation, we further implement an optimized backend using a dedicated vLLM engine with `-enforce-eager` disabled. This reduces the reward computation time from approximately 8 seconds to 0.8 seconds per rollout. In our training setup, this optimization can reduce the overall training time from about 10 minutes per step to around 6 minutes per step. For reference, an LLM-as-a-Judge reward based on a strong external model such as GPT-5.4 typically requires around 2 seconds per rollout in our setup(roughly 6–7 minutes per step). These results suggest that although CLR incurs extra likelihood-computation cost, efficient inference backends can substantially mitigate the overhead and make the training efficiency practically manageable.

A.5 Omission of KL Divergence

We omit the KL divergence penalty during RL training. In the standard GRPO objective, the KL term is defined as $\beta \log \frac{\pi_\theta}{\pi_{\text{ref}}}$. However, in our setting, this term creates an optimization conflict with the proposed CLR objective. Specifically, CLR encourages the policy model to increase the likelihood of responses that are strongly supported by the retrieved documents, whereas the KL penalty discourages deviation from the reference policy. This conflict can suppress the optimization of the contrastive reward and hinder effective learning.

In preliminary experiments, we observed that including the KL term led to unstable training and a continuous decline in reward values. We tested KL coefficients $\beta \in \{0.05, 0.04, 0.03, 0.01\}$ and found that training collapsed within 20 steps in all cases, manifested as reward degradation, garbled rollout outputs, or repetitive generation. Therefore, we remove the KL penalty in our final RL training setup to enable stable optimization of the contrastive reward.

A.6 Rationale for the Significance Threshold τ

In our final design, we set $\tau = 0$ in Equation 11, so that only responses with positive Evidential Contribution receive non-zero reward. This reflects our goal of suppressing non-positive evidential signals, which are often ambiguous and less reliable for training. We further track the proportion of samples with $\mathcal{E}(y) \leq 0$ during training. At the global batch level, this proportion is about 0.07 at the beginning, increases to 0.11 in the middle stage, and drops to 0.03 near convergence. This suggests that non-positive samples are present throughout training and can introduce noise into policy optimization, especially in earlier stages. We also evaluate larger thresholds, $\tau = 0.5$ and $\tau = 1$, but observe no performance gain. Moreover, the proportion of samples satisfying $\mathcal{E}(y) \leq \tau$ does not change substantially compared with the case of $\tau = 0$, indicating that most suppressed samples are already concentrated in the non-positive region. As a result, increasing τ mainly zeros out weak but potentially useful positive signals, rather than isolating a substantially different set of noisy cases. Overall, these results support $\tau = 0$ as a simple and effective choice: it masks ambiguous non-positive signals while preserving useful grounded supervision.

B Interpretability Analysis: What the Model Learns

In the previous sections, we defined our contrastive likelihood-based reward R_{CLR} and proposed two hybrid reward formulations, R_{hybrid}, which integrate R_{CLR} with the accuracy reward R_{acc}. In this section, we provide a more intuitive analysis of these rewards to better understand what the model has learned.

We visualize the token-level Evidential Contribution in Figure7 to examine which tokens are encouraged by our reward model, R_{CLR}. From the figure, we observe the following patterns:

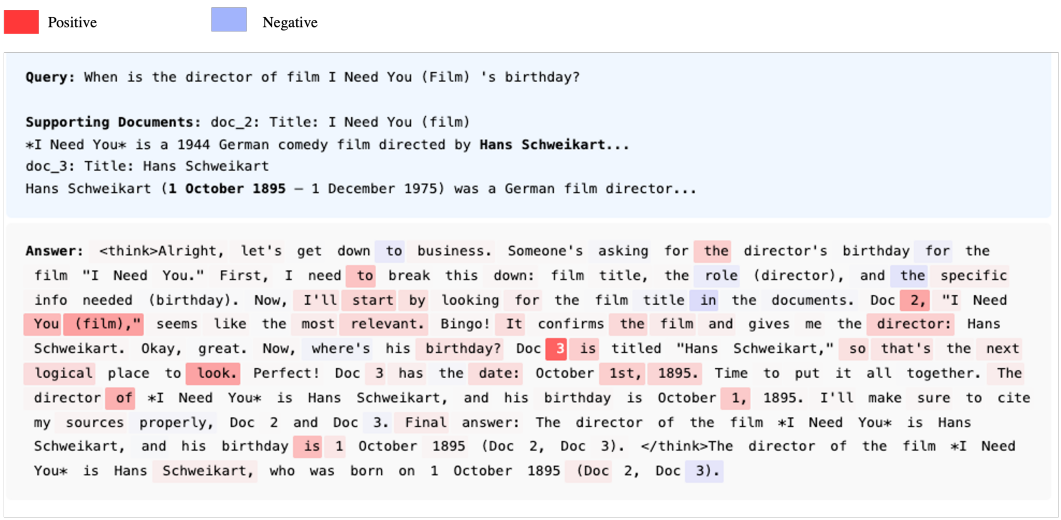


Figure 7: An example of token-level Evidential Contribution. The darker the color, the larger the absolute value of $\epsilon(y_t)$.

Grounding in Evidence: Tokens originating from supporting documents, such as document IDs (e.g., "2", "3") and their specific contents (e.g., "1st, 1895"), consistently receive high rewards. This confirms that our method effectively steers the model to locate and utilize relevant source material.

Discouraging Redundancy: Repeated content tends to receive lower token-level Evidential Contribution than newly introduced evidence-bearing tokens. Intuitively, once a piece of evidence has already been incorporated into the response, repeating it usually adds little additional support to the generation. As a result, later repetitions contribute less to the contrastive likelihood difference. At the sequence level, the $\frac{1}{\sqrt{T}}$ normalization further reduces the incentive to accumulate reward through verbosity or repetition. Together, these effects encourage the model to introduce new, evidence-supported information rather than restating the same content, leading to more concise and stable generations.

Collectively, these observations provide compelling evidence that R_{CLR} yields substantial improvements in document-based reasoning, faithfulness, and efficiency.

C Evaluation Settings

We use benchmark in RagQALeaderboardZhehao Tan, Yihan Jiao, Dan Yang (2026), which provided a fair environment of RAG-QA scenarios to evaluate models, and PRGBTan et al. (2025) to evaluate contextual faithfulness of LLMs. For inference, we leverage the vLLM framework deployed on a cluster of eight NVIDIA H800 GPUs. Regarding specific task configurations, we retrieve 30 documents for RagQALeaderboard. For the PRGB benchmark, we conduct three iterations with a noise distribution of 4, 4, and 1 for noise document levels 1, 2, and 3, respectively. For Evaluation prompts, we followed the default prompts of their benchmarks.

D Additional Results on Qwen3-8B

To further evaluate the robustness of the proposed reward design across model scales, we additionally conduct experiments on Qwen3-8B. The results demonstrate trends consistent with those observed on Qwen3-30B-A3B, indicating that the proposed contrastive likelihood reward generalizes effectively across both dense and MoE architectures.

Table 3: **Additional results on Qwen3-8B and open-source baselines.** The best and second-best results within each Qwen3-8B setting are marked in **bold** and underline, respectively. AVG denotes the arithmetic mean across all benchmarks.

Method	AVG	Multi-hop			Single-hop		Faithfulness	Biomedical
		2Wiki	Hotpot	MuSiQue	TQA	PQA	PRGB	PubMed
Qwen3-8B (think)								
SFT	81.6	82.0	79.6	66.8	91.4	89.6	84.7	77.2
SFT+RL(R_{acc})	83.6	84.9	81.5	70.5	92.6	91.1	86.7	78.0
SFT+RL(R_{cite})	82.4	82.6	79.7	68.1	91.8	91.6	85.8	77.2
SFT+RL(R_{total})	83.9	85.0	81.3	70.7	92.6	91.4	86.6	**80.0**
SFT+RL(R_{CLR})	83.8	84.4	81.6	69.3	92.3	92.5	88.1	78.6
SFT+RL(R_{hybrid})	**85.0**	**86.2**	**82.4**	**71.5**	**92.8**	**93.2**	**89.4**	79.8
Qwen3-8B (no think)								
SFT	72.0	74.6	76.1	60.2	87.7	88.6	74.1	43.0
SFT+RL(R_{acc})	77.0	79.8	79.6	64.3	91.1	89.7	76.6	58.2
SFT+RL(R_{cite})	77.2	78.3	78.8	62.9	91.8	90.4	74.9	63.2
SFT+RL(R_{total})	79.0	78.9	78.6	64.8	91.6	90.4	76.4	74.8
SFT+RL(R_{CLR})	79.7	80.4	79.9	64.4	91.6	89.6	76.6	**75.6**
SFT+RL(R_{hybrid})	**80.4**	**82.1**	**80.7**	**65.2**	**92.1**	**90.7**	**77.6**	74.8
Open-source Post-trained Models								
Qwen3-8B	61.9	71.4	71.0	44.5	90.2	74.0	64.7	17.6
Qwen3-14B	63.2	68.8	69.4	37.6	86.8	72.7	65.1	42.0
Qwen3-32B	69.3	77.9	73.2	48.6	89.4	76.4	68.3	51.6
Kimi-K2-Instruct	75.8	82.3	76.3	**66.5**	92.5	86.3	71.5	**55.2**
Qwen3-30B-A3B-Instruct-2507	75.5	81.4	81.9	55.8	**94.2**	86.7	75.9	52.4
Qwen3-235B-A22B-Instruct-2507	**78.0**	**84.9**	**82.8**	65.3	93.8	**87.6**	**78.2**	53.2

E LLM-AS-A-JUDGE PROMPTS

In this section, we present the full prompts used for our LLM-as-a-Judge evaluation framework. We design two distinct judge prompts targeting different aspects of response quality: (1) an **Accuracy Judge** (§E.1) that evaluates whether the model's answer is semantically consistent with the ground-truth answer, and (2) a **Faithfulness Judge** (§E.2) that evaluates whether each paragraph of the model's response is factually grounded in the retrieved context documents.

E.1 ACCURACY JUDGE PROMPT

The accuracy judge prompt instructs the LLM to compare the model's answer against the ground-truth answer at the semantic level, assessing whether the core information is consistent regardless of surface-level phrasing differences. The judge outputs a structured JSON object containing a binary verdict (`0` or `1`) and a textual justification. The full prompt is shown in Prompt E.1.

Prompt 1 — Accuracy Judge ($R_{\text{J-Acc}}$)

 # Role
You will play the role of a rigorous and detailed AI response evaluator. Your task is to strictly compare the *model answer* with the *standard answer* based on the given *question*, and determine whether the core information of the two is consistent.

 # Evaluation Criteria

1. **Semantic Equivalence:** The model answer and the standard answer do not have to be identical in wording. If they convey the same core facts, viewpoints and conclusions, it shall be judged as "consistent".
2. **Completeness:**
   - If the model answer contains **all key information** of the standard answer, it is still judged as "consistent" even if there is a small amount of additional information that does not affect the core meaning.

- If the model answer **lacks any key information point** in the standard answer, it shall be judged as "inconsistent".
- If the model answer contains facts **contradictory** to the standard answer, or provides completely wrong additional information, it shall be judged as "inconsistent".

3. **Final Goal:** Your judgment shall focus on whether users can obtain the same core knowledge or instructions after reading the model answer as from the standard answer.

 # Output Format
You must strictly follow the JSON format below. Do not add any extra text or explanations before and after the JSON.

```
"judge": 0 or 1, "reason": "Your judgment reason."
```

 # Input Placeholders

- `{question}`: The question posed to the model.
- `{answer}`: The model-generated answer to be evaluated.
- `{gt}`: The ground-truth standard answer.

E.2 Faithfulness Judge Prompt

The faithfulness judge prompt instructs the LLM to perform paragraph-level fact-checking of the model's response against the provided retrieved context. For each paragraph, the judge outputs a structured JSON object containing a verdict label (`supported`, `not_supported`, or `no_rad`), a rationale, and a supporting excerpt from the context. The full prompt is shown in Prompt E.2.

Prompt 2 — Faithfulness Judge ($R_{\text{J-Faith}}$)

 # Task
Evaluate the factual accuracy of the model response against the provided context document. Act as a strict fact-checker. Do **not** use any external knowledge or make assumptions beyond what is stated in the context.

 # Output Format
For each paragraph in the response (separated by line breaks), output a JSON object with the following fields:

- `paragraph_number`: (integer) Sequential index of the paragraph (starting from 1).
- `summary`: (string) A brief summary of the main claims (1–2 sentences).
- `label`: (string) One of the following verdicts:
  - `supported` — fully and accurately supported by the context.
  - `not_supported` — not accurately supported; includes partial inaccuracies, unsupported inferences, misattributions, or ignored contradictory context.
  - `no_rad` — introductory, concluding, or conversational content that makes no verifiable factual claims.
- `rationale`: (string) Detailed justification referencing the evaluation criteria.
- `excerpt`: (string or `null`) Direct quote(s) from the context most relevant to the verdict; multiple excerpts concatenated with "`;` ". Set to `null` if label is `no_rad` or if no relevant text exists.

 # Evaluation Criteria

1. **Explicit Support Only:** A paragraph is `supported` only if the information is **explicitly stated** or **strongly/trivially implied** by the context. Do not mark paragraphs as `supported` if they require non-trivial inference or assumption.

2. **Precision and Nuance:** Pay close attention to specific terminology, qualifiers (*e.g.*, "significantly", "most", "some"), quantities, and causal relationships. Simplifications are only acceptable if they maintain full factual accuracy.
3. **Completeness and Context:** A paragraph is `not_supported` if it cherry-picks information while ignoring contradictory or significantly qualifying information elsewhere in the same document.
4. **Accurate Attribution:** Claims must be attributed to the correct subject, object, or scope as defined in the context. Misattributions yield `not_supported`.
5. **No Hallucinations:** The response must not introduce information, terms, or concepts not present in the context.
6. **Paragraph-Level Evaluation:** If a paragraph contains multiple claims and even one is not supported, the entire paragraph is labeled `not_supported`.

 # Output Example

```
"paragraph_number": 1, "summary": "...", "label": "supported", "rationale": "...", "excerpt": "Relevant quote from context."
```

 # Note on Quotation Marks
Do **not** use raw quotation marks (`"`) inside field values. Replace any quotation marks with backticks (`` ` ``).

 # Input Placeholders

- `{question}`: The question posed to the model.
- `{answer}`: The model-generated response to be evaluated.
- `{context}`: The retrieved context documents.