LongR: Unleashing Long-Context Reasoning via Reinforcement Learning with Dense Utility Rewards

Anonymous ACL submission

Abstract

Reinforcement Learning (RL) has emerged as a key driver for LLM reasoning. This capability is equally pivotal in long-context scenarios—such as long-dialogue understanding and structured data analysis—where the challenge extends beyond consuming tokens to performing rigorous deduction. While existing efforts focus on data synthesis or architectural changes, recent work points out that relying solely on sparse, outcome-only rewards yields limited gains, as such coarse signals are often insufficient to effectively guide the complex long-context reasoning. To address this, we propose **LongR**, a unified framework that enhances long-context performance by integrating a dynamic "Think-and-Read" mechanism—which interleaves reasoning with document consultation—with a contextual density reward based on relative information gain to quantify the utility of the relevant documents. Empirically, LongR achieves a 9% gain on LongBench v2 and consistent improvements on RULER and InfiniteBench, demonstrating robust efficiency in navigating extensive contexts. Furthermore, LongR consistently enhances performance across diverse RL algorithms (e.g., DAPO, GSPO). Finally, we conduct in-depth analyses to investigate the impact of reasoning chain length on efficiency and the model's robustness against distractors.

1 Introduction

Reinforcement Learning (RL) has emerged as a potent mechanism for unlocking the reasoning potential of Large Language Models (LLMs) (Guo et al., 2025a; Team et al., 2025a). However, reasoning is not only critical in domains like mathematics but is equally pivotal in the long-context domain. In real-world scenarios—ranging from synthesizing insights across heterogeneous documents to debugging repository-scale codebases—the challenge lies not merely in consuming vast input tokens, but in performing rigorous deduction and comprehension over the content. Moreover, distinct long-context reasoning capabilities are foundational for autonomous agents to sustain coherent actions over extended trajectories.

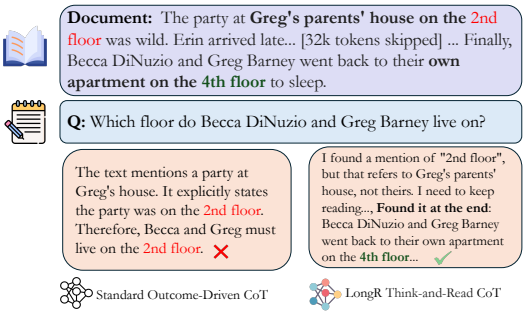

Figure 1: Qualitative comparison between Outcome-Driven CoT (Standard RL, left) and Think-and-Read CoT (LongR, right). Left: Without dense supervision, the baseline model tends to rely on superficial query matching while overlooking detailed context reading. Right: LongR incorporates a contextual dense reward. By incentivizing the model to actively consult the context during the reasoning process, it significantly enhances long-context understanding capabilities.

Recently, there has been a surge of interest in applying RL to long-context scenarios (Bai et al., 2025a; Zeng et al., 2025). Prior works have predominantly focused on synthesizing high-quality reasoning data (Wang et al., 2025) or modifying model architectures to natively handle extended sequences (Team et al., 2025b). More recently, LongRLVR (Anonymous, 2025) pointed out that relying solely on sparse rewards yields limited performance gains for long-context RL. They proposed partitioning the context into discrete segments and employing an interleaved mechanism of reasoning and chunk selection to enhance long-context understanding.

However, existing exploration like LongR-LVR (Anonymous, 2025) requires pre-segmenting documents into discrete chunks and training the policy to output specific chunk identifiers. While

effective in specific settings, this paradigm imposes structural constraints that limit broader applicability: (1) **Information Integrity Risks**: Such rigid segmentation introduces an inherent risk of disrupting data integrity, as guaranteeing that information remains completely uncompromised is non-trivial. Furthermore, this approach necessitates artificial data construction with additional overhead, while the resulting specialized format inherently diverges from mainstream post-training distributions and natural user queries. (2) **Interrupted Reasoning Flow**: Decoupling the action of "looking up" from the flow of "reasoning" forces the model to adhere to a specialized output format (e.g., discrete tag generation) rather than natural language reasoning. In contrast, standard LLMs benefit from a coherent chain of thought that mirrors natural language reasoning.

To bridge this gap, we propose **LongR**, a unified framework operating strictly within standard autoregressive generation. We highlight three key contributions: (1) **Dynamic "Think-and-Read" Mechanism**: This mechanism mirrors the human cognitive process of interleaving reasoning with document consultation by integrating evidence seeking directly into the chain of thought. By eliminating the need for discrete chunking, it preserves the integrity of the original information. (2) **Contextual Density Reward via Relative Information Gain**: Recognizing that not all contexts readily support the construction of explicitly verifiable rewards (e.g., raw documents lacking fine-grained annotations), we instead quantify the information-theoretic utility of retrieved documents to provide dense supervision. By introducing this intrinsic density reward, we incentivize the model to actively seek high-value information, thereby enhancing its efficiency in utilizing long contexts. (3) **Self-Learned Reasoning Pattern**: We show that the "Think-and-Read" pattern can be acquired from generic, readily available data, without ever constructing long-context-specific "consult-then-reason" demonstrations. Through a progressive curriculum, models learn to dynamically interleave consultation with reasoning purely via RL, eliminating the need for costly, long-context-specific data engineering.

We empirically validate LongR using LongBench v2 (Bai et al., 2025a), where it achieves a performance gain of approximately 9% shown in Table 1. LongR exhibits remarkable universality, enhancing generalization on generic long-context tasks (e.g., RULER and InfiniteBench (Hsieh et al., 2024; Zhang et al., 2024)) shown in Table 2 while consistently boosting performance across a diverse spectrum of RL algorithms, including DAPO, GSPO, CISPO (Yu et al., 2025b; Zheng et al., 2025; MiniMax et al., 2025) shown in Table 6. Furthermore, we validate the effectiveness of our proposed contextual reward in Table 4, where it significantly outperforms outcome-only baselines by approximately 4%. Additionally, Table 7 demonstrates that our curriculum learning strategy contributes substantially to performance gains. Finally, we conduct in-depth analyses on the length of reasoning chains in terms of computational efficiency, as well as LongR's robustness against distractors.

Our contributions can be summarized as follows:

- We propose **LongR**, which leverages an additional contextual reward to guide long-context reasoning.
- We conduct extensive experiments (using LongBench v2, RULER, etc.) to validate the effectiveness of LongR. For instance, LongR achieves an improvement of 9% on LongBench v2.
- We conduct in-depth analyses to investigate the efficiency and robustness of LongR.

2 Related Work

Reinforcement Learning in Long-context Scenario. While Reinforcement Learning (RL) has become a staple for enhancing mathematical reasoning (Yu et al., 2025b; Zheng et al., 2025), its adaptation to long-context domains is a burgeoning field. Prevailing strategies predominantly focus on architectural innovations—such as linear or sparse attention mechanisms (Team et al., 2025b,c; Gao et al., 2025)—which often necessitate extensive pre-training. Others, Wang et al. (2025) and Shen et al. (2025) prioritize the synthesis of superior reasoning data. LongRLVR (Anonymous, 2025) demonstrates that relying solely on outcome-based rewards yields limited gains for long-context reasoning and supplements sparse outcome signals using pre-segmented chunks. In contrast, our method avoids rigid segmentation, thereby preserving data integrity and ensuring broader applicability. A concurrent work, EAPO (Guan et al., 2026), addresses reward sparsity by training a learned black-box proxy to score evidence quality. However, such

learned rewards are prone to instability, necessitating a computationally expensive reward-policy co-evolution loop to iteratively refine the reward model during training. In contrast, LongR employs Relative Information Gain, a white-box, information-theoretic metric derived directly from a frozen verifier.

Reward Design in Reinforcement Learning. The design of reward functions is a pivotal factor (Ouyang et al., 2022; Kwon et al., 2023). Previous methods use reward models in RL and approaches like RewardAnything (Yu et al., 2025c), GenPRM (Zhao et al., 2025), and others (Guo et al., 2025b; Chen et al., 2025; Li et al., 2025) leverage the semantic understanding of LLMs to evaluate intermediate reasoning steps or generalizable objectives. Similarly, in the long-context domain, LongReward (Zhang et al., 2025) utilizes LLM feedback to score multidimensional quality. However, relying on learned RMs presents a challenge, as reward hacking remains difficult to completely mitigate. Consequently, recent breakthroughs such as DeepSeek-R1 (Guo et al., 2025a) have pivoted back to rule-based, outcome-only rewards, relying on verifiable signals to drive robust self-evolution without hacking risks. Yet, in long-context scenarios, the extended reasoning horizon renders outcome signals excessively sparse (Anonymous, 2025); we mitigate this by introducing additional relative information gain as an intrinsic dense reward, offering better robustness.

3 Method

3.1 Problem Formulation

RL for Long-Context Reasoning. We formulate the long-context question answering task as a Markov Decision Process (MDP). Unlike standard language modeling which conditions solely on a query x, our policy π_θ must process a lengthy document c to derive the answer y. The training objective is to maximize the expected reward while maintaining a trust region constraint relative to a reference model π_{ref}:

$$\max_{\pi_\theta} \mathbb{E}_{\substack{x,c\sim\mathcal{D} \\ y\sim\pi_\theta(\cdot|x,c)}} \Big[r_\phi(x,c,y) - \beta\, \mathbb{D}_{\text{KL}}\big(\pi_\theta(\cdot|x,c) \| \pi_{\text{ref}}(\cdot|x,c)\big) \Big] \tag{1}$$

where r_ϕ denotes the reward function that evaluates the correctness and quality of the reasoning chain y given the context c.

3.2 Supervised Fine-tuning (Cold Start)

To mitigate the instability of training reinforcement learning from scratch, we employ a warm-up stage using Supervised Fine-Tuning (SFT), effectively treating it as behavioral cloning. In this phase, the policy π_θ is initialized to mimic expert reasoning behaviors by maximizing the likelihood of high-quality demonstration trajectories. The optimization objective is defined as:

$$\mathcal{L}_{\text{SFT}}(\theta) = - \mathbb{E}_{(x,c,y)\sim\mathcal{D}_{\text{SFT}}} \left[\sum_{t=1}^{|y|} \log \pi_\theta(y_t|x,c,y_{<t}) \right] \tag{2}$$

where c represents the long document, x is the query, and y denotes the ground-truth chain-of-thought sequence. This process serves two critical roles: injecting domain knowledge into the model and strictly enforcing the structural requirements (e.g., the `<think>` encapsulation) necessary for the subsequent group-based exploration.

3.3 LongR Training Framework

We introduce LongR, a framework that integrates contextual dense rewards with curriculum learning to enhance long-context understanding capabilities.

3.3.1 Intrinsic Grounding Reward via Information Gain

In long-context reasoning tasks, optimizing policies solely via sparse outcome rewards poses significant optimization challenges. As identified by LongRLVR (Anonymous, 2025), relying exclusively on a final scalar feedback can lead to vanishing gradients for the contextual grounding process. Consequently, the model fails to learn effective contextual grounding, as the supervision signal for identifying relevant evidence vanishes due to the vast search space of the long context.

Conventionally, the reward function relies on a combination of format constraints and final answer correctness. Let y be the generated trajectory and y^* be the ground truth. The standard sparse reward r_{base} is formulated as:

$$r_{\text{base}}(y, y^*) = r_{\text{fmt}}(y) + r_{\text{acc}}(y, y^*), \tag{3}$$

where $r_{\text{fmt}}(\cdot)$ is an indicator function that ensures the output adheres to the required structure (e.g., enclosing reasoning within `<think>` tags), and $r_{\text{acc}}(\cdot)$ verifies whether the final answer matches y^*.

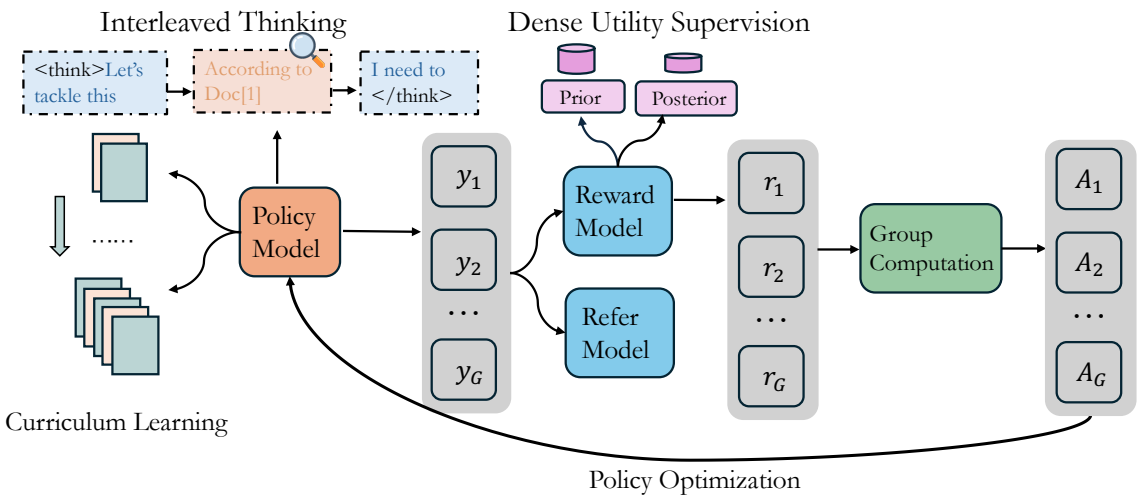

Figure 2: Overview of the LongR Framework. The system operates within a standard reinforcement learning pipeline enhanced by three key mechanisms: (1) Curriculum Learning: through a progressive curriculum, models learn to dynamically interleave consultation with reasoning purely via RL, eliminating the need for costly, long-context-specific SFT data engineering. (2) Interleaved Think-and-Read Policy: generating actions as a dynamic stream where reasoning steps (blue) alternate with grounded evidence extraction (orange), enabling flexible information seeking without rigid chunking. (3) Dense Utility Supervision: providing additional fine-grained feedback by quantifying the utility of each extracted document via relative information gain.

We augment the standard objective with a dense verification signal. We formulate the total reward r_{total} as:

$$r_{\text{total}}(x, c, y) = r_{\text{base}}(y, y^*) + r_{\text{ctx}}(y, c), \quad (4)$$

where $r_{\text{ctx}}(y, c)$ denotes contextual reward, which quantifies contextual utility via information gain.

Probabilistic Grounding Verification. To facilitate an interleaved process of reasoning and document consultation, we employ a specialized prompt template (see Figure 4). This constraint compels the policy π_θ to explicitly distinguish between internal reasoning and external citations. Consequently, based on the generated trajectory y, we identify the set of relevant context segments $\mathcal{S} = \{s_1, s_2, \ldots, s_K\}$ that the model explicitly consulted. These segments represent the concrete information the model processed to facilitate long-context understanding.

Quantifying Contextual Utility via Mutual Information. We measure the contextual utility of each consulted segment $s_k \in \mathcal{S}$. We approach this through an Information-Theoretic lens, positing that if the model genuinely "reads" and relies on the document, the context c must provide a significant reduction in uncertainty regarding s_k.

Formally, we define this utility as the pointwise information gain (Cover, 1999), calculated as the log-ratio between the posterior and prior probabilities:

$$\begin{aligned} \text{Score}(s_k, c) &\approx \log \frac{P(s_k|c)}{P(s_k)} \\ &= \log P(s_k|c) - \log P(s_k). \end{aligned} \quad (5)$$

To compute the utility score objectively, we employ a frozen verifier model π_{ver}, distinct from the optimizing policy. We utilize Qwen3-4B[1] as π_{ver} for two primary reasons: (1) **Computational efficiency**: using a larger model to score every generated segment during RL training would impose prohibitive computational overhead; (2) **Representational adequacy**: we assume that a model pretrained on an extensive corpus which is 36T tokens as reported in (Yang et al., 2025) provides a sufficiently accurate estimate of the true conditional and marginal probabilities for our purpose of quantifying information gain. This choice represents a pragmatic trade-off.

The utility is quantified using token-level negative log-likelihood (NLL) derived from this verifier. For a segment s_k consisting of T tokens $(x_1, \ldots, x_T)$, the NLL is defined as:

$$\mathcal{L}(s_k|\text{ctx}) = -\frac{1}{T} \sum_{t=1}^{T} \log \pi_{\text{ver}}(x_t|x_{<t}, \text{ctx}). \quad (6)$$

[1] https://huggingface.co/Qwen/Qwen3-4B

Consequently, the final score is computed as the uncertainty reduction captured by the verifier:

$$\text{Score}(s_k, c) = \mathcal{L}(s_k) - \mathcal{L}(s_k|c). \tag{7}$$

The choice of NLL as the underlying metric is that a reduction in NLL serves as a robust proxy for the presence of valid long-range dependencies (Bai et al., 2025b). By maximizing this NLL-based mutual information, we explicitly incentivize the model to use context effectively.

Reward Normalization. While the raw score captures the absolute information gain, its magnitude is sensitive to the intrinsic complexity of the text segment, which can introduce variance instability during reinforcement learning. To derive a scale-invariant signal, we normalize the score by the model's prior uncertainty. This transforms the metric into the relative information gain (Theil, 1967), representing the fraction of uncertainty explained by the context:

$$\begin{aligned}\frac{\text{Score}(s_k, c)}{\mathcal{L}(s_k)} &= \frac{\mathcal{L}(s_k) - \mathcal{L}(s_k|c)}{\mathcal{L}(s_k)} \\ &= 1 - \frac{\mathcal{L}(s_k|c)}{\mathcal{L}(s_k)}.\end{aligned} \tag{8}$$

We obtain the final dense reward:

$$r(s_k, c) = 1 - \frac{\mathcal{L}(s_k|c)}{\mathcal{L}(s_k)}. \tag{9}$$

This bounded reward $r(s_k, c)$ encourages the policy to maximize the *relative* explanatory power of the retrieved context. Finally, the trajectory-level contextual reward is the average score of all unique quotes:

$$r_{\text{ctx}}(y, c) = \frac{1}{K} \sum_{k=1}^{K} r(s_k, c). \tag{10}$$

If no quotes are generated ($K = 0$), the $r_{\text{ctx}}(y, c)$ is set to 0.

3.3.2 Progressive Context Curriculum

To eliminate the need for costly, long-context-specific data engineering, we adopt a curriculum learning strategy where the model is progressively exposed to documents of increasing length.

Specifically, we partition the training corpus into M subsets based on document length. In stage m ($0 \leq m < M$), the model is trained exclusively on documents whose length does not exceed L_m, where $L_m = \min(L_{\text{max}}, L_0 \times 2^m)$. For instance, the initial stage ($m = 0$) might use documents up to 16K tokens, while subsequent stages gradually increase this limit to 32K, and finally the full target length L_{max}.

4 Experiments

4.1 Set Up

We conduct experiments using the `Qwen3` family at both 4B and 8B scales. Our training consists of two stages: supervised fine-tuning (SFT) followed by reinforcement learning (RL) with DAPO (Yu et al., 2025b). We employ a two-stage curriculum learning strategy, progressively increasing context length from 16K to 32K tokens.

We evaluate LongR on three long-context benchmarks: **LongBench v2** (Bai et al., 2025a), **RULER** (Hsieh et al., 2024), and **InfiniteBench** (Zhang et al., 2024). Baselines include: (1) officially released `Qwen3-Base` models; (2) our reproduced `Qwen3-SFT` models; and (3) `Qwen3-SFT w/ DAPO` trained with outcome-only rewards. All evaluations use official scripts with a fixed random seed.

Detailed training configurations, hyperparameters, and dataset specifications are provided in Appendix B.

4.2 Main Results

Results on LongBench v2. Table 1 presents results on LongBench v2. LongR yields substantial gains over both the SFT baseline and outcome-only RL (DAPO). For `Qwen3-8B`, LongR achieves an overall score of 36.23, improving over DAPO by +3.39 and over SFT by +9.59. The `Qwen3-4B` model shows a similar trend, with LongR reaching 35.44. Overall, outperforming DAPO by +5.03 and SFT by +9.79.

Notably, LongR delivers the strongest improvements on challenging examples: Hard questions improve from 25.65 to 33.20 (8B) and from 28.10 to 35.61 (4B), while the Long split rises from 23.35 to 39.12 (8B). These results indicate that our method enables more effective use of document context over shallow heuristics.

4.3 Ablation on More Long-context Benchmarks

Table 2 shows that LongR consistently improves performance on the RULER benchmark at both 128K and 64K context lengths. For the 8B model,

		Difficulty		Length		
Model	**Overall**	**Easy**	**Hard**	**Short**	**Medium**	**Long**
Qwen3-8B*	33.38	40.62	28.93	35.65	28.76	34.65
Qwen3-8B-Base	20.58	19.67	21.14	27.93	17.86	14.08
Qwen3-8B-SFT	26.64	28.25	25.65	32.02	24.45	23.35
Qwen3-8B-SFT w/ DAPO	32.84	39.90	28.49	37.93	28.57	32.21
Qwen3-8B-SFT w/ LongR	**36.23**	**41.15**	**33.20**	**38.75**	**32.67**	**39.12**
Qwen3-4B*	31.46	34.70	29.46	35.16	28.10	**31.46**
Qwen3-4B-Base	17.13	19.59	15.62	20.71	13.69	17.78
Qwen3-4B-SFT	25.65	21.54	28.10	32.93	22.53	19.64
Qwen3-4B-SFT w/ DAPO	30.41	28.32	31.63	33.33	31.36	23.87
Qwen3-4B-SFT w/ LongR	**35.44**	**35.16**	**35.61**	**38.75**	**37.21**	26.39

Table 1: Evaluation results (%) on LongBench v2. Qwen3-8B* and Qwen3-4B* are officially released models available on HuggingFace. Short, Medium, and Long refer to length ranges of <32k, 32-128k, and >128k, respectively. Qwen3-8B* is evaluated using the official thinking mode, and we have set the random seed to 42 to ensure reproducibility.

Model	**RULER-128k**				**RULER-64k**			
	NIAH	**VT**	**QA**	**Avg**	**NIAH**	**VT**	**QA**	**Avg**
Qwen3-8B-Base	51.88	47.68	17.60	39.05	61.76	28.08	39.70	43.18
Qwen3-8B-SFT	61.64	51.96	30.50	48.03	62.96	31.00	38.80	44.25
Qwen3-8B-SFT w/ DAPO	57.64	76.56	30.40	54.87	66.13	58.52	40.40	55.02
Qwen3-8B-SFT w/ LongR	**58.63**	**80.20**	**45.59**	**61.47**	**69.88**	**86.64**	**59.41**	**71.98**
Qwen3-4B-Base	46.74	45.64	14.00	35.46	60.55	56.40	19.30	45.42
Qwen3-4B-SFT	53.02	**91.44**	27.20	57.22	66.20	90.92	20.20	59.11
Qwen3-4B-SFT w/ DAPO	54.57	90.48	28.70	57.92	67.74	91.96	30.90	63.53
Qwen3-4B-SFT w/ LongR	**56.28**	90.84	**35.00**	**60.71**	**71.36**	**93.32**	**43.60**	**69.43**

Table 2: Evaluation results (%) on RULER-128K and RULER-64K. NIAH denotes Needle-In-A-Haystack retrieval, VT denotes Variable Tracking, and QA denotes multi-hop question answering.

LongR achieves average scores of 61.47 (128K) and 71.98 (64K), outperforming the outcome-only DAPO baseline by +6.60 and +16.96 respectively. The 4B model shows a similar trend, with LongR attaining 60.71 (128K) and 69.43 (64K), leading in all settings. Notable gains appear in VT and QA for the 8B model, and primarily in QA for the 4B model.

Table 3 presents results on InfiniteBench, where LongR again achieves the highest average scores: 50.18 for 8B and 47.60 for 4B. It delivers substantial gains on key subtasks, most notably improving En.MC from 54.15 to 58.45 (8B) and boosting En.QA from 12.68 to 21.85 (4B).

4.4 Impact of Reward Design Choices

To validate the effectiveness of our proposed reward design, we conduct an ablation study comparing our method against three distinct reward configurations.

As presented in Table 4, the Relative Information Gain yields the superior performance. The results reveal critical insights into reward design:

Outcome-only Reward. The sparse outcome signal is insufficient to guide the policy through the vast search space of long-context reasoning.

w/ Absolute Density. This variant underperforms because the raw information gain is unbounded and sensitive to intrinsic text complexity. The policy tends to hack the reward by generating unnecessarily verbose or esoteric segments to maximize prior uncertainty ($\mathcal{L}(s_k)$), rather than focusing on true relevance.

w/ Threshold Density. It indiscriminately rewards all segments above the cutoff equally, failing

Methods	Re.Pa	Re.Nu	En.Sum	En.QA	Zh.QA	En.MC	Avg.
Qwen3-8B-SFT	85.76	86.44	14.18	25.09	14.47	54.15	46.68
Qwen3-8B-SFT w/ DAPO	85.93	83.56	11.86	**27.45**	**23.54**	56.33	48.11
Qwen3-8B-SFT w/ LongR	**87.44**	**89.76**	**15.72**	27.40	22.31	**58.45**	**50.18**
Qwen3-4B-SFT	86.78	83.44	12.76	12.68	9.65	53.28	43.10
Qwen3-4B-SFT w/ DAPO	**87.63**	84.75	**17.78**	14.83	10.39	54.59	44.99
Qwen3-4B-SFT w/ LongR	86.93	**87.05**	16.07	**21.85**	**17.39**	**56.29**	**47.60**

Table 3: Evaluation results (%) on InfiniteBench. Re.Pa and Re.Nu denote Retrieve.PassKey and Retrieve.Number, En.Sum denotes English Summarization, En.QA and Zh.QA denote English and Chinese question answering, and En.MC denotes English multiple-choice reasoning.

		Difficulty		Length		
Model	**Overall**	**Easy**	**Hard**	**Short**	**Medium**	**Long**
Qwen3-8B-SFT	26.64	28.25	25.65	32.02	24.45	23.35
Reinforcement Learning						
Outcome-only Reward	32.80	40.10	28.30	38.33	28.37	32.41
w/ Absolute Density	32.41	**41.41**	26.85	35.97	**33.14**	25.00
w/ Threshold Density	34.10	35.16	**33.44**	**41.25**	26.74	36.81
w/ Relative Density (Ours)	**36.23**	41.15	33.20	38.75	32.67	**39.12**
Qwen3-4B-SFT	25.65	21.54	28.10	32.93	22.53	19.64
Reinforcement Learning						
Outcome-only Reward	30.42	28.12	31.83	33.33	31.16	24.07
w/ Absolute Density	31.61	30.21	32.48	36.67	31.63	23.15
w/ Threshold Density	33.95	**38.02**	31.43	**40.14**	29.77	**31.94**
w/ Relative Density (Ours)	**35.44**	35.16	**35.61**	38.75	**37.21**	26.39

Table 4: Ablation study on Reward Design. We compare our proposed method against three baselines: (1) Outcome-only, which relies **exclusively** on sparse final signals (r_{acc} and r_{fmt}) without dense supervision; (2) w/ Absolute Density, which substitutes the relative metric with the absolute information gain (Eq. 7) as the dense signal; and (3) w/ Threshold Density, which applies a binary reward scheme, assigning a discrete value of 1 only if the relative gain (Eq. 9) exceeds 0.5, and 0 otherwise. Evaluation results (%) on LongBench v2.

to distinguish between moderately relevant text and precise evidence.

w/ Relative Density (Ours). Our method effectively acts as a scale-invariant filter. For irrelevant or parametric content, the context offers no new predictive power, resulting in similar prior and posterior likelihoods ($\mathcal{L}(s_k|c) \approx \mathcal{L}(s_k)$), which drives the reward to near zero. Conversely, for specific, context-dependent evidence (e.g., unique entities or dates), the document significantly reduces the high initial uncertainty ($\mathcal{L}(s_k|c) \ll \mathcal{L}(s_k)$), resulting in a high reward. This mechanism naturally encourages the model to extract information that is maximally informative yet grounded, regardless of the segment's length. A detailed case study is provided in Figure 10 to illustrate how our method works.

4.5 Compare with Different Sizes of Verifier Models

Our contextual reward uses a frozen verifier π_{ver} to estimate information gain (Eqs. 7–9). We thus study how verifier scale affects performance to find an efficient balance.

Results in Table 5 show that while larger verifiers generally improve performance, gains saturate quickly. Moving from a 1.7B to 4B verifier yields clear improvements for both Qwen3-8B (Overall: 34.59→36.23) and Qwen3-4B (Overall: 35.09→35.44), particularly on harder and longer tasks. Further scaling to 32B brings only marginal

		Difficulty		Length		
Model	**Overall**	**Easy**	**Hard**	**Short**	**Medium**	**Long**
Qwen3-8B-SFT	26.64	28.25	25.65	32.02	24.45	23.35
+LongR w/ 1.7B verifier	34.59	39.06	31.83	43.33	26.98	35.19
+LongR w/ 4B verifier	36.23	41.15	33.20	38.75	32.67	39.12
+LongR w/ 32B verifier	36.83	41.67	33.84	38.75	33.14	40.97
Qwen3-4B-SFT	25.65	21.54	28.10	32.93	22.53	19.64
+LongR w/ 1.7B verifier	35.09	32.03	36.98	37.08	33.37	35.19
+LongR w/ 4B verifier	35.44	35.16	35.61	38.75	37.21	26.39
+LongR w/ 32B verifier	36.08	36.20	36.01	38.61	36.40	31.25

Table 5: Ablation study on Verifier Size. We evaluate LongR performance using varying verifier sizes (1.7B, 4B, and 32B) from the Qwen3 family. For a controlled comparison, all other components remain fixed. Evaluation results (%) on LongBench v2.

		Difficulty		Length		
Model	**Overall**	**Easy**	**Hard**	**Short**	**Medium**	**Long**
Qwen3-8B-SFT	26.64	28.25	25.65	32.02	24.45	23.35
w/ DAPO	**36.23**	**41.15**	33.20	38.75	32.67	**39.12**
w/ CISPO	34.99	35.94	34.41	34.41	**38.61**	33.84
w/ GSPO	35.39	33.92	**39.17**	**39.17**	35.12	29.63
Qwen3-4B-SFT	25.65	21.54	28.10	32.93	22.53	19.64
w/ DAPO	**35.44**	**35.16**	35.61	38.75	**37.21**	26.39
w/ CISPO	33.95	31.51	35.45	**40.14**	31.98	27.55
w/ GSPO	33.90	30.60	**35.93**	39.03	32.91	**33.90**

Table 6: Ablation study on different RL algorithms. Evaluation results (%) on LongBench v2. Best results within each backbone are highlighted in **bold**.

gains (8B: 36.23→36.83; 4B: 35.44→36.08), with uneven improvements across task splits. A detailed analysis of training efficiency is presented in Figure 11. These results indicate that a 4B verifier provides an optimal balance: it captures most performance gains while remaining computationally efficient for training-time scoring. We therefore adopt Qwen3-4B as our default verifier.

4.6 Generalizable Improvements Across RL Algorithms

We further benchmark two additional variants, CISPO and GSPO, under the same training recipe. Both methods deliver substantial improvements over the *SFT* baselines on LongBench v2 for both 8B and 4B (e.g., +8.35 and +8.75 overall for 8B and +8.30 and +8.25 overall for 4B), achieving a performance comparable to DAPO. Notably, CISPO and GSPO distribute their gains differently across evaluation splits. CISPO shows a clear advantage on medium-length instances (38.61% on the 8B Medium split, the best among all methods), while GSPO is particularly strong on more challenging regimes, achieving the best 8B performance on the Hard split (39.17%) and the best 4B performance on the Long split (33.90%). These results suggest that our recipe remains effective across different RL algorithms, and that the choice of RL objective mainly affects where the improvements concentrate along the length and difficulty spectrum rather than whether improvements emerge. Unless otherwise specified, we use DAPO by default due to its strongest and most balanced overall performance.

5 Conclusion

We present LongR for enhancing long-context reasoning. LongR introduces a "Think-and-Read" pattern that interleaves internal reasoning with explicit document consultation, and proposes a contextual dense reward based on relative information gain to mitigate reward sparsity.

Limitations

Due to computational constraints, we were only able to conduct RL training at context lengths up to 32K tokens, as scaling to longer sequences incurs a significantly higher computational budget. We plan to explore scaling to longer contexts in future work.

References

Anonymous. 2025. Longrlvr: Overcoming the long-context bottleneck in reinforcement learning with verifiable rewards. OpenReview. Under review as a conference paper at ICLR 2026.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, and 1 others. 2025a. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3639–3664.

Yushi Bai, Jiajie Zhang, Xin Lv, Linzhi Zheng, Siqi Zhu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2025b. Longwriter: Unleashing 10, 000+ word generation from long context llms. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang, Denghui Zhang, Tong Zhang, and 1 others. 2025. Rm-r1: Reward modeling as reasoning. *arXiv preprint arXiv:2505.02387*.

Thomas M Cover. 1999. *Elements of information theory*. John Wiley & Sons.

Yizhao Gao, Shuming Guo, Shijie Cao, Yuqing Xia, Yu Cheng, Lei Wang, Lingxiao Ma, Yutao Sun, Tianzhu Ye, Li Dong, and 1 others. 2025. Seerattention-r: Sparse attention adaptation for long reasoning. *arXiv preprint arXiv:2506.08889*.

Xin Guan, Zijian Li, Shen Huang, Pengjun Xie, Jingren Zhou, and Jiuxin Cao. 2026. Evidence-augmented policy optimization with reward co-evolution for long-context reasoning. *arXiv preprint arXiv:2601.10306*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025a. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Jiaxin Guo, Zewen Chi, Li Dong, Qingxiu Dong, Xun Wu, Shaohan Huang, and Furu Wei. 2025b. Reward reasoning model. *arXiv preprint arXiv:2505.14674*.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. Ruler: What's the real context size of your long-context language models? *arXiv preprint arXiv:2404.06654*.

Sam Ade Jacobs, Masahiro Tanaka, Chengming Zhang, Minjia Zhang, Shuaiwen Leon Song, Samyam Rajbhandari, and Yuxiong He. 2023. Deepspeed ulysses: System optimizations for enabling training of extreme long sequence transformer models. *Preprint*, arXiv:2309.14509.

Minae Kwon, Sang Michael Xie, Kalesha Bullard, and Dorsa Sadigh. 2023. Reward design with language models. *arXiv preprint arXiv:2303.00001*.

Yi-Chen Li, Tian Xu, Yang Yu, Xuqin Zhang, Xiong-Hui Chen, Zhongxiang Ling, Ningjing Chao, Lei Yuan, and Zhi-Hua Zhou. 2025. Generalist reward models: Found inside large language models. *arXiv preprint arXiv:2506.23235*.

MiniMax, :, Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, Chengjun Xiao, Chengyu Du, Chi Zhang, Chu Qiao, Chunhao Zhang, Chunhui Du, Congchao Guo, and 109 others. 2025. Minimax-m1: Scaling test-time compute efficiently with lightning attention. *Preprint*, arXiv:2506.13585.

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

Weizhou Shen, Ziyi Yang, Chenliang Li, Zhiyuan Lu, Miao Peng, Huashan Sun, Yingcheng Shi, Shengyi Liao, Shaopeng Lai, Bo Zhang, Dayiheng Liu, Fei Huang, Jingren Zhou, and Ming Yan. 2025. Qwenlong-l1.5: Post-training recipe for long-context reasoning and memory management. *Preprint*, arXiv:2512.12967.

Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, and 1 others. 2025a. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*.

Kimi Team, Yu Zhang, Zongyu Lin, Xingcheng Yao, Jiaxi Hu, Fanqing Meng, Chengyin Liu, Xin Men, Songlin Yang, Zhiyuan Li, and 1 others. 2025b. Kimi linear: An expressive, efficient attention architecture. *arXiv preprint arXiv:2510.26692*.

MiniCPM Team, Chaojun Xiao, Yuxuan Li, Xu Han, Yuzhuo Bai, Jie Cai, Haotian Chen, Wentong Chen, Xin Cong, Ganqu Cui, and 1 others. 2025c. Minicpm4: Ultra-efficient llms on end devices. *arXiv preprint arXiv:2506.07900*.

Henri Theil. 1967. Economics and information theory.

Siyuan Wang, Gaokai Zhang, Li Lyna Zhang, Ning Shang, Fan Yang, Dongyao Chen, and Mao Yang. 2025. Loongrl: Reinforcement learning for advanced reasoning over long contexts. *arXiv preprint arXiv:2510.19363*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, and 1 others. 2025a. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent. *arXiv preprint arXiv:2507.02259*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, and 1 others. 2025b. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Zhuohao Yu, Jiali Zeng, Weizheng Gu, Yidong Wang, Jindong Wang, Fandong Meng, Jie Zhou, Yue Zhang, Shikun Zhang, and Wei Ye. 2025c. Rewardanything: Generalizable principle-following reward models. *arXiv preprint arXiv:2506.03637*.

Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, and 1 others. 2025. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models. *arXiv preprint arXiv:2508.06471*.

Jiajie Zhang, Zhongni Hou, Xin Lv, Shulin Cao, Zhenyu Hou, Yilin Niu, Lei Hou, Yuxiao Dong, Ling Feng, and Juanzi Li. 2025. LongReward: Improving long-context large language models with AI feedback. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3718–3739, Vienna, Austria. Association for Computational Linguistics.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Hao, Xu Han, Zhen Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. 2024. ∞Bench: Extending long context evaluation beyond 100K tokens. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15262–15277, Bangkok, Thailand. Association for Computational Linguistics.

Jian Zhao, Runze Liu, Kaiyan Zhang, Zhimu Zhou, Junqi Gao, Dong Li, Jiafei Lyu, Zhouyi Qian, Biqing Qi, Xiu Li, and 1 others. 2025. Genprm: Scaling test-time compute of process reward models via generative reasoning. *arXiv preprint arXiv:2504.00891*.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, and 1 others. 2025. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*.

A Templates

System Prompt Template

You are a rigorous, evidence-driven reasoning assistant. You will be presented with a question based on a provided lengthy document; your task is to answer the question based on the document.
First, carefully read the query and search for relevant information within the long document. Then, reason based on the quoted evidence to formulate your answer.
The reasoning process and answer must be enclosed within `<think>` and `<answer>` tags, respectively.
Example Format:
`<think>` reasoning process `</think>` `<answer>` answer here `</answer>`.

Figure 3: The system prompt used for outcome-only RL.

System Prompt Template

You are a rigorous, evidence-driven reasoning assistant. You will be presented with a question based on a provided lengthy document; your task is to answer the question based on the document.
First, carefully read the query and search for relevant information within the long document. When you find a relevant document, you can quote the exact text snippets verbatim within your reasoning. Then, reason based on the quoted evidence to formulate your answer.
The reasoning process and answer must be enclosed within `<think>` and `<answer>` tags, respectively.
Example Format:
`<think>` reasoning process ...[Relevant text from the document] ...`</think>` `<answer>` answer here `</answer>`.

Figure 4: The system prompt template used for interleaved reasoning and document consultation. It explicitly instructs the model to generate structured reasoning paths with quoted evidence.

B Training and Evaluation Details

Training Setting. We conduct experiments using the `Qwen3` family at both 4B and 8B scales. All training processes are executed on 16 NVIDIA H20 141GB GPUs.

- **Supervised Fine-tuning.** We conduct our training using LLaMA-Factory[2]. For the cold-start phase, we utilize 20k open-source instruction-following samples from

[2] https://github.com/hiyouga/LLaMA-Factory

AM-DeepSeek-R1-0528-Distilled[3] and we employ a cosine learning rate scheduler with a peak rate of 2e-5 and a warmup ratio of 0.03. We set the maximum sequence length to 16,384, with sequence packing and training for 900 steps.

- **Reinforcement Learning.** We use verl[4] for RL and use DAPO (Yu et al., 2025b).

  Curriculum Learning: We adopt a two-stage curriculum strategy, with each stage comprising 100 training steps and a constant learning rate of 1e-6. The context length is limited to 16K in the initial stage ($m = 0$) and extended to 32K in the subsequent stage ($m = 1$).

  Data: Following MemAgent (Yu et al., 2025a), we synthesize 1,000 samples for each length configuration (16K and 32K). To further bolster reasoning capabilities, we incorporate the DocQA-RL-1.6K dataset[5] during the second stage ($m = 1$).

  Reward Configuration: The format reward r_{fmt} is set to 1.0 for valid `<think>...</think>` and `<answer>...</answer>` structure, and the accuracy reward r_{acc} is set to 2.0 for correct answers. The contextual density reward r_{ctx} is computed using a frozen `Qwen3-4B` verifier.

  Hyperparameters: We set the learning rate to 1e-6, the batch size 8, and the rollout number to 16. The maximum sequence length is 32,768, and we allow a maximum output length of 4096 tokens and Ulysses sequence parallel 2 (Jacobs et al., 2023).

Evaluation Setting. We employ a comprehensive suite of long-context benchmarks to rigorously evaluate model performance:

- **LongBench v2** (Bai et al., 2025a): A challenging benchmark focused on realistic long-context deeper understanding. It employs a multiple-choice question format to facilitate rigorous and objective evaluation.

- **RULER** (Hsieh et al., 2024): A benchmark designed to evaluate the effective context window size through diverse synthetic tasks.

- **InfiniteBench** (Zhang et al., 2024): A dataset targeting diverse long-context capabilities, covering heterogeneous tasks such as retrieval (Re.Pa, Re.Nu), summarization (En.Sum), QA (En.QA, Zh.QA), and multi-choice reasoning (En.MC).

For inference, we utilize vLLM[6] with a decoding temperature of 1.0. We set maximum generated tokens to 16384. All evaluations are conducted using official scripts with a fixed random seed of 42 to ensure reproducibility.

Baselines. We conduct comparisons using the `Qwen3` family at both 4B and 8B scales. The baselines include: (1) the officially released `Qwen3-Base` models; (2) our reproduced `Qwen3-SFT` models obtained from the Cold Start phase; and (3) the `Qwen3-SFT w/ DAPO` variants, which undergo standard full-parameter RL training (Outcome-only) using the same data and steps as LongR with template Figure 3 for comparison.

C Stage-wise Performance of Curriculum Learning

Results and analysis. As shown in Table 7, curriculum learning yields effective and stable improvements across stages. For `Qwen3-8B`, the Overall score increases from 26.64 (`SFT`) to 32.41 after the 16K warm-up (Stage m=0), and further to 36.23 after extending to 32K (Stage m=1). The largest gains are observed on the **Hard** (33.20) and **Long** (39.12) subsets, indicating that the model progressively learns to ground reasoning in longer contexts.

The same trend holds for `Qwen3-4B`: Overall performance rises from 25.65 to 31.56 (Stage m=0) and then to 35.44 (Stage m=1), with balanced gains across Difficulty and Length splits. These results validate that progressive context scaling enables the model to first acquire core reasoning skills in shorter contexts and then generalize them effectively to longer documents.

[3] https://huggingface.co/datasets/a-m-team/AM-DeepSeek-R1-0528-Distilled
[4] https://github.com/volcengine/verl
[5] https://huggingface.co/datasets/Tongyi-Zhiwen/DocQA-RL-1.6K

		Difficulty		Length		
Model	**Overall**	**Easy**	**Hard**	**Short**	**Medium**	**Long**
Qwen3-8B-SFT	26.64	28.25	25.65	32.02	24.45	23.35
W/ Curriculum Learning						
w/ Stage m = 0	32.41	38.54	28.62	35.56	27.91	36.11
w/ Stage m = 1	36.23	41.15	33.20	38.75	32.67	39.12
Qwen3-4B-SFT	25.65	21.54	28.10	32.93	22.53	19.64
W/ Curriculum Learning						
w/ Stage m = 0	31.56	31.25	31.75	36.81	30.35	25.23
w/ Stage m = 1	35.44	35.16	35.61	38.75	37.21	26.39

Table 7: Detailed contribution of each stage for curriculum learning.

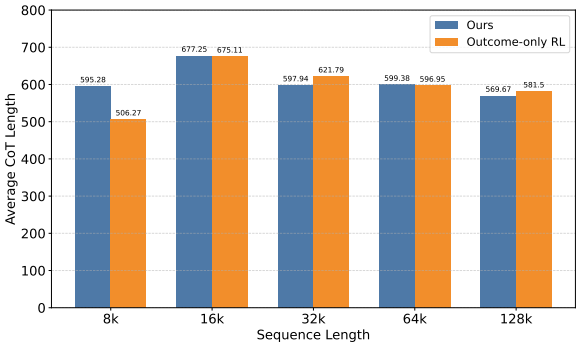

Figure 5: Reasoning chain length comparison for 8B models.

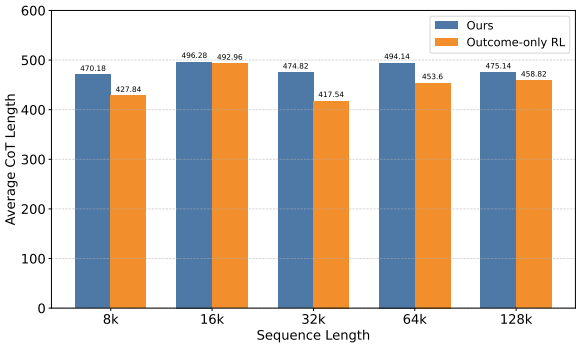

Figure 6: Reasoning chain length comparison for 4B models.

D Analysis

D.1 Length of Reasoning Chain

We investigate whether the introduction of the "Think-and-Read" mechanism imposes a significant computational burden or encourages reward hacking behavior (e.g., indiscriminately copying long document segments). Figure 5 and 6 illustrate the average length of reasoning chains generated by LongR compared to the Outcome-only baseline on the RULER (NIAH) benchmark across varying context lengths ranging from 8k to 128k. The task's design—embedding a single specific fact within a vast corpus of unrelated text—creates a sharp demarcation between relevant evidence (the needle) and irrelevant noise (the haystack). The corresponding accuracy are shown in Figure 8 and Figure 9.

Surgical Extraction vs. Length Inflation. A primary concern with density-based rewards is that the policy might learn to "hack" the metric by blindly copying extensive, irrelevant text chunks to accumulate dense rewards. As observed, LongR maintains an average token count strictly comparable to—and in distinct cases (e.g., 32k, 128k) even marginally lower than—the Outcome-only baseline.

This stability provides compelling evidence for the robustness of our **Relative Information Gain** design. Since the relative metric assigns negligible scores to irrelevant or parametric content (where the posterior likelihood $\mathcal{L}(s_k|c)$ remains similar to the prior $\mathcal{L}(s_k)$), the optimization landscape naturally penalizes the inclusion of "noise." Consequently, instead of inflating the output with hallucinations or irrelevant copies, the model converges to a **surgical extraction strategy**: it learns to pinpoint and quote only the specific "needles" necessary for answering the query. This confirms that LongR enhances long-context grounding without incurring additional inference overhead or succumbing to length-based reward hacking.

D.2 Distractors Resistance

We further evaluate robustness using the **Distractor Count** metric on the NIAH task, where any citation other than the specific "needle" is considered noise. As shown in Figure 7, despite being explicitly in-

[6] https://github.com/vllm-project/vllm

centivized to consult the document via r_{ctx}, LongR displays a distractor usage profile nearly identical to the Outcome-only baseline across all context lengths. This indicates that the model does not resort to "spray-and-pray" tactics (indiscriminately quoting text) to accumulate dense rewards.

E Additional Figures

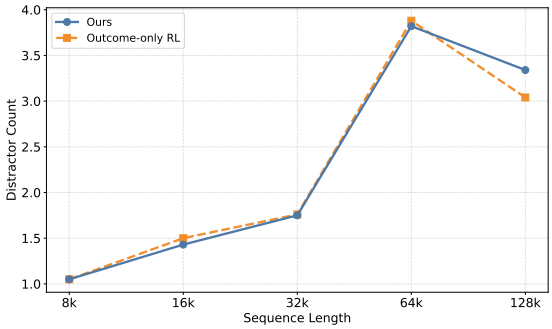

Figure 7: Distractor count for 8B models.

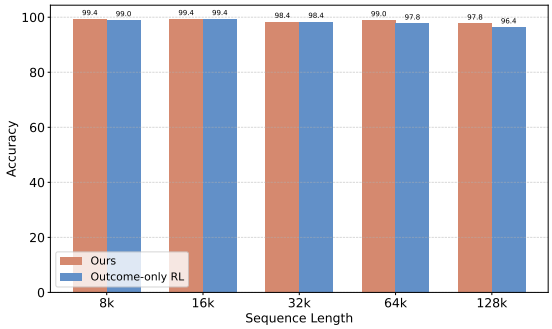

Figure 8: The NIAH accuracy for 8B models.

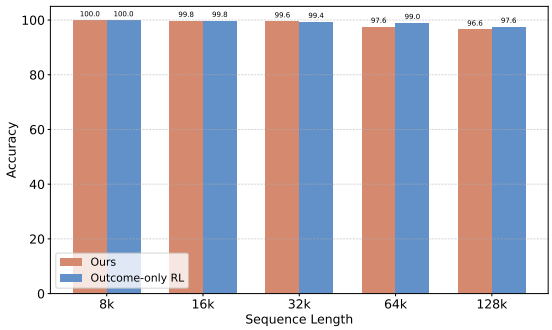

Figure 9: The NIAH accuracy for 4B models.

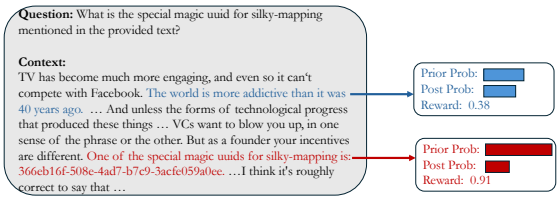

Figure 10: Mechanism Decomposition of Relative Information Gain. We visualize the internal calculation of r_{ctx} on two distinct segments, which are synthesized from RULER. For the Semantic Distractor (blue), the prior loss $\mathcal{L}(s)$ is already low due to the text's high predictability (common sense), resulting in a limited relative gain (0.38) despite context consistency. In contrast, the Target Evidence (red) exhibits a high prior loss (unpredictable UUID) which is drastically reduced by the context, yielding a dominant reward (0.91). This decomposition confirms that LongR prioritizes information utility (surprise reduction) over mere textual fluency.

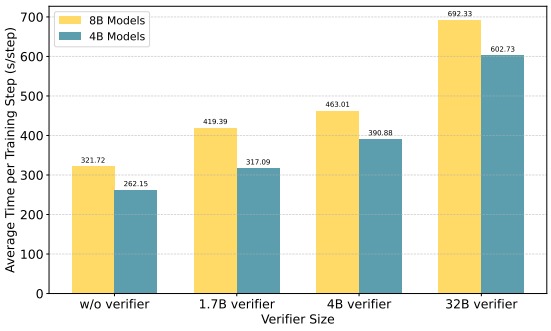

Figure 11: Comparison of average training step time across different verifier sizes. The y-axis "Average Time per Training Step (s/step)" measures the mean duration of a single forward pass during training. The average is computed over the first 100 training steps under the experimental setup described in Section B.