Exploring Fine-Tuning for In-Context Retrieval and Efficient KV-Caching in Long-Context Language Models

Anonymous ACL submission

Abstract

With context windows of millions of tokens, Long-Context Language Models (LCLMs) can encode entire document collections, offering a strong alternative to conventional retrieval-augmented generation (RAG). However, it remains unclear whether fine-tuning strategies can improve long-context performance and translate to greater robustness under KV-cache compression techniques. In this work, we investigate which training strategies most effectively enhance LCLMs' ability to identify and use relevant information, as well as enhancing their robustness under KV-cache compression. Our experiments show substantial in-domain improvements: achieving up to +20-point gains over the base model. However, out-of-domain generalization remains task dependent with large variance – LCLMs excels on finance questions (+9 points), while RAG shows stronger performance on multiple-choice questions (+6 points) over the baseline models. Finally, we show that our fine-tuning approaches bring moderate improvements in robustness under KV-cache compression, with gains varying across tasks.

1 Introduction

Long-Context Language Models (LCLMs) have demonstrated remarkable performance across diverse benchmarks through continual pre-training and context extension techniques (Peng et al., 2023; Su et al., 2023; Xiong et al., 2024; Gao et al., 2025). The main goal is to enable the encoding of increasingly longer documents, potentially encompassing entire knowledge bases and eliminating the need for retrieval pipelines (Lee et al., 2024). This motivation stems from the fact that retrieval-augmented generation (RAG), while effective as the de-facto standard for knowledge-intensive tasks, requires storing and managing comprehensive knowledge indexes while maintaining robust retrieval capabilities. However, despite the significant progress in long-context modeling, current literature demonstrates that standard RAG systems continue to outperform LCLMs across various tasks (Li et al., 2024a,b; Lee et al., 2024; Bai et al., 2024a; Xu et al., 2024; Jin et al., 2024). Recent works have explored the effects of supervised fine-tuning (SFT) on LCLMs through question answering over long-context inputs (Qiu et al., 2025; Zhang et al., 2024a; Jin et al., 2024), yet these studies typically train and evaluate on in-domain data, falling short in assessing their out-of-domain generalization and the factors contributing to the model performance. Moreover, little attention has been devoted to understanding which training techniques yield the best performance under KV-cache compression, a critical consideration given the cost of storing complete key-value caches.

We argue that to effectively replace RAG systems, models must learn to behave like in-context learning RAG systems: attending primarily to relevant information within the input context while disregarding irrelevant content. This motivates our research question: *Does fine-tuning LCLMs to selectively attend to relevant information improve in-context retrieval performance and enable more efficient inference?*

Given the shortcomings of SFT in the literature (Qiu et al., 2025), we explore GRPO-based (Shao et al., 2024) approaches with verifiable reward functions, from simple answer-only objectives to complex reasoning with LLM judge evaluation, comparing their performance against RAG pipelines in both in- and out-of-domain scenarios. Our findings show that certain strategies surpass RAG on in-domain benchmarks, though out-of-domain generalization remains inconsistent. Finally, we show that fine-tuning brings moderate improvements in robustness under KV-cache compression, while our analysis reveals minimal gains in attention-based document ranking, exhibiting negligible correlation with task performance.

2 Methodology

Training Data. To improve the in-context retrieval capabilities of the LCLMs, we generate training data containing a sparse set of relevant documents within the input context, with the vast majority of information consisting of hard negatives that encourage the model to attend selectively to relevant content. We construct our training data from two open-domain question answering benchmarks: HotpotQA (Yang et al., 2018) and 2WikiMultihopQA (Ho et al., 2020). Each instance in these benchmarks includes an annotated list of (up to 2) relevant passages from Wikipedia required to answer the question.

We employ a strong retriever to fetch additional Wikipedia passages related to each question and concatenate them with the gold relevant passages. This creates a challenging setting where the majority of retrieved information may be topically related to the question but not necessary for answering it (Qiu et al., 2025). Following Karpukhin et al. (2020), we parse Wikipedia by dividing each article into 100-word passages, ensuring no length bias during training (the gold passages provided in these datasets also average approximately 100 words). To prevent positional bias during training, we randomly shuffle both the relevant passages and hard negatives within each context. Then, for each document d_i in the context, we associate it with a unique ID using a special tag (i.e., "[DOC i]"), in order to distinguish clear document boundaries.

Additional details and examples about training data can be found in Appendix B.

Training Strategies. Given that current SFT techniques for eliciting in-context learning capabilities of LCLMs have shown limited benefits (Qiu et al., 2025; Jin et al., 2024), we hypothesize that LCLMs require exploration during training to discover which information is truly relevant to answering input questions. To enable this exploration, we employ reinforcement learning with GRPO, which allows the model to sample multiple outputs and learn from group-relative advantages.

Given a question q and context c containing passages $\{d_1, \ldots, d_p\}$, we train a policy model π_θ to generate a response y that identifies relevant documents and produces the correct answer a^*. Let $\mathcal{D}^* \subseteq \{1, \ldots, p\}$ denote the set of gold-standard relevant document indices. We design five verifiable reward functions $R(y, q, c)$ to guide model training, each encouraging different aspects of relevant information extraction and reasoning.

Reward Functions. To encourage models to develop selective attention capabilities, we design five reward functions that progressively increase in complexity and constraint. We hypothesize that simpler objectives (i.e., answer correctness alone) may allow models greater flexibility in learning attention patterns, while more structured objectives (i.e., requiring document citations or content reproduction) may provide stronger supervision but risk overfitting to specific output formats. We define these reward functions for model output y as follows, where $\text{ans}(y)$ extracts the answer, $\text{ids}(y)$ extracts cited document indices, $\text{cont}(y)$ extracts reproduced content, $\text{quotes}(y)$ extracts quoted spans, and $\approx$ denotes sub-exact match:

$$R_{\text{AO}}(y) = \mathbb{1}[\text{ans}(y) \approx a^*] \quad (1)$$

$$R_{\text{ID}}(y) = \mathbb{1}[\text{ids}(y) = \mathcal{D}^*] + R_{\text{AO}}(y) \quad (2)$$

$$R_{\text{ID+C}}(y) = \mathbb{1}[\text{cont}(y) \approx \{d_i : i \in \mathcal{D}^*\}] + R_{\text{ID}}(y) \quad (3)$$

$$R_{\text{ID+Q}}(y) = \mathbb{1}[\text{quotes}(y) \subseteq \{d_i : i \in \mathcal{D}^*\}] + R_{\text{ID}}(y) \quad (4)$$

$$R_{\text{R+Judge}}(y) = J(\text{reason}(y), \mathcal{D}_{\text{cited}}, \mathcal{D}^*, q, c) + R_{\text{AO}}(y) \quad (5)$$

$R_{\text{AO}}(y)$ rewards solely correct answers. $R_{\text{ID}}(y)$ additionally encourages identifying all relevant document IDs before answering. $R_{\text{ID+C}}(y)$ further rewards the model if it reproduces the full text of relevant documents. $R_{\text{ID+Q}}(y)$ rewards extracting pertinent quotes from relevant documents rather than reproducing entire passages. $R_{\text{R+Judge}}(y)$ prompts the model to provide step-by-step reasoning, which an LLM judge J evaluates for coherence and appropriate use of cited documents $\mathcal{D}_{\text{cited}}$. Details about the reward functions can be found in Appendix C.

3 Experimental Setup

Training. For training data construction, we employ Qwen3-Embedding-4B (Zhang et al., 2025) as the retriever and set $k = 500$ to retrieve the top 500 passages for each question. We then filter instances to retain only those with context lengths of at most 32k tokens. This choice ensures two key advantages: *i)* training remains computationally efficient; *ii)* it allows us to evaluate whether models trained on these moderate-length contexts can generalize to much longer contexts, including 1M tokens. Details about the prompts and hyperparameters used to train our models can be found in Appendix D.

Model	HELMET (SubEM ↑)			
	HotpotQA	NQ	TriviaQA	Avg.
Base	61.0	61.1	86.6	69.6
w/ gold docs	72.2	70.7	85.8	76.2
+ $R_{\text{AO}}(y)$	**81.1**	67.7	92.1	**80.3**
+ $R_{\text{ID}}(y)$	75.2	65.5	91.0	77.2
+ $R_{\text{ID+C}}(y)$	67.9	56.9	85.8	70.2
+ $R_{\text{ID+Q}}(y)$	74.1	59.1	85.5	72.9
+ $R_{\text{R+Judge}}(y)$	70.6	**72.8**	**92.3**	78.6

Table 1: Average performance on HELMET's "RAG" subset across context lengths 4k–128k.

Benchmarks. We evaluate our models on a diverse set of benchmarks, domains and tasks. For in-domain analysis, we select the HotpotQA (Yang et al., 2018), NQ (Kwiatkowski et al., 2019), and TriviaQA (Joshi et al., 2017) subsets from the RAG split of HELMET (Yen et al., 2025), where each instance contains a question with p passages ($p - k$ distractors and k relevant documents). In these subsets, is it assumed that RAG has been applied to construct the input context; therefore, we do not perform additional retrieval. For out-of-domain evaluation, we assess performance on ∞Bench (Zhang et al., 2024b) (MC, QA, and SUM), LongBench-v2 (Bai et al., 2025, LB-v2), and the English financial split of Loong (Wang et al., 2024), which requires reasoning over tables and numbers. The prompts used in our experiments can be found in Appendix E, while details on the evaluation metrics can be found in Appendix F.

Models and Baselines. Based on preliminary experiments comparing multiple LCLMs with context windows from 128k to 1M against their RAG counterparts (Appendix G), we select Qwen2.5-7B-Instruct-1M (Qwen et al., 2025) as our base model. This model showed a clear performance gap between full-context and RAG, making it an ideal candidate for investigating whether fine-tuning can close this gap. We use RAG@32k as our baseline, as it achieves the highest average performance.

4 Results

In-Domain Results. Table 1 presents average performance on the HELMET's RAG subset across context lengths from 4k to 128k tokens (see Appendix H for per-length results). The $R_{\text{AO}}(y)$ reward function achieves the strongest improvements on HotpotQA, with gains of 20.1 points (81.1 vs. 61.0) over the base model, while $R_{\text{R+Judge}}(y)$ achieves the best performance on NQ (72.8 vs. 61.1) and TriviaQA (92.3 vs. 86.6). The $R_{\text{ID}}(y)$ reward function also yields substantial gains, averaging 75.2 on HotpotQA, 65.5 on NQ, and 91.0 on TriviaQA. However, more constrained strategies, specifically $R_{\text{ID+C}}(y)$ and $R_{\text{ID+Q}}(y)$, show performance degradation, suggesting that overly restrictive training objectives may impede the model's ability to process extended contexts effectively. Notably, both $R_{\text{AO}}(y)$ and $R_{\text{R+Judge}}(y)$ surpass the performance obtained when the base model is informed about which documents are relevant: $R_{\text{AO}}(y)$ exceeds gold docs on HotpotQA (81.1 vs. 72.2) and TriviaQA (92.1 vs. 85.8), while $R_{\text{R+Judge}}(y)$ surpasses gold docs on NQ (72.8 vs. 70.7) and TriviaQA (92.3 vs. 85.8). This indicates that fine-tuning successfully enables models to identify relevant information without sacrificing the model's ability to leverage its full context.

Model	∞Bench			LB-v2	Loong	
	MC (Acc. ↑)	QA (SubEM ↑)	Sum (R-L ↑)	ALL (Acc. ↑)	Fin. (Judge ↑)	Avg.
Full Context (up to 1M tokens)						
Base	72.0	30.0	31.2	32.0	49.9	43.0
+ $R_{\text{AO}}(y)$	70.3	**38.2**	31.0	31.2	55.3	45.2
+ $R_{\text{ID}}(y)$	71.6	34.7	29.7	31.0	51.2	43.6
+ $R_{\text{ID+C}}(y)$	70.7	31.3	30.2	31.4	47.5	42.2
+ $R_{\text{ID+Q}}(y)$	70.7	30.8	30.5	32.0	51.5	43.1
+ $R_{\text{R+Judge}}(y)$	72.5	30.7	**31.7**	32.6	**58.8**	**45.3**
KV-Cache with RetrievalAttn						
Base	62.0	27.6	27.3	30.2	39.5	37.3
+ $R_{\text{AO}}(y)$	65.5	33.6	28.0	29.6	41.7	39.7
+ $R_{\text{ID}}(y)$	64.2	29.3	26.0	30.0	43.0	38.5
+ $R_{\text{ID+C}}(y)$	63.3	27.1	27.7	29.6	39.8	37.5
+ $R_{\text{ID+Q}}(y)$	62.9	27.1	28.1	32.0	39.7	38.0
+ $R_{\text{R+Judge}}(y)$	63.8	28.2	28.8	31.2	41.5	38.7
RAG (Base Model)						
RAG@32k	**78.6**	29.3	25.1	**38.0**	47.8	43.8

Table 2: Out-of-domain performance comparison.

Model	Performance Drop (%)					
	MC	QA	Sum	ALL	Fin	Avg
Base	-13.9	-27.6	-12.5	-5.6	-20.8	-16.1
+ $R_{\text{AO}}(y)$	**-6.8**	-28.3	-9.7	-5.1	-24.6	-14.9
+ $R_{\text{ID}}(y)$	-10.3	-28.3	-12.5	-3.2	**-16.0**	-14.1
+ $R_{\text{ID+C}}(y)$	-10.5	-28.7	-8.3	-5.7	-16.2	**-13.9**
+ $R_{\text{ID+Q}}(y)$	-11.0	-32.7	**-7.9**	**0.0**	-22.9	-14.9
+ $R_{\text{R+Judge}}(y)$	-12.0	-30.2	-10.5	-4.5	-18.7	-15.2

Table 3: Performance drop (%) with RA vs. full context. MC/QA/Sum: ∞Bench, ALL: LB-v2, Fin: Loong.

Out-of-Domain Results. Table 2 evaluates generalization to out-of-domain benchmarks where

Model	Documents Ranking (NDCG@10)			
	HotpotQA	NQ	TriviaQA	Avg.
Base	82.0	84.7	83.6	83.4
+ $R_{\text{AO}}(y)$	82.3	85.0	83.8	83.7
+ $R_{\text{ID}}(y)$	82.3	85.0	83.7	83.7
+ $R_{\text{ID+C}}(y)$	82.4	85.1	84.1	83.9
+ $R_{\text{ID+Q}}(y)$	**82.5**	**85.4**	**84.3**	**84.1**
+ $R_{\text{R+Judge}}(y)$	82.2	84.9	83.7	83.6

Table 4: NDCG@10 computed on reranked documents on HELMET at 32k context length.

inputs consist of long, cohesive documents rather than retrieved passages. Results reveal mixed patterns across tasks and models. $R_{\text{R+Judge}}(y)$ demonstrates the most substantial improvement on Loong Financial, achieving 58.8 (+8.9 over base) and outperforming RAG@32k by 11 points (58.8 vs. 47.8). Similarly, $R_{\text{AO}}(y)$ yields the largest gain on ∞Bench QA (38.2, +8.2 points), also surpassing RAG@32k (38.2 vs. 29.3). However, on other benchmarks (LB-v2, ∞Bench MC and SUM) we do not observe substantial improvement: multiple-choice accuracy remains largely unchanged (70.3-72.5 vs. 72.0 base), and summarization shows minimal variation (30.2-31.7 vs. 31.2 base). Conversely, RAG at 32k tokens outperforms all fine-tuned models on ∞Bench MC (78.6% vs. 72.5% best fine-tuned) and LB-v2 (38.0% vs. 32.6% best fine-tuned). These mixed results indicate that no single approach consistently dominates across diverse out-of-domain tasks, with both fine-tuning and RAG offering complementary strengths.

KV-Cache Compression Analysis. To assess whether fine-tuning translates to stable performance in efficient inference settings, we evaluate models under RetrievalAttention (Liu et al., 2024) (RA), a KV-cache compression technique that retains only the top-k most-attended key-value pairs. We select RA because it directly aligns with our training objective: models that allocate more attention to relevant content should theoretically retain performance when compression preserves only highly-attended tokens.

Tables 2 and 3 quantify performance under RA, showing that all models experience substantial degradation, with the base model dropping 16.1% on average. Fine-tuned models reduce this degradation moderately: $R_{\text{ID+C}}(y)$ achieves the smallest drop (13.9%), representing 2.2 percentage point improvements over the base model. However, even the best compressed fine-tuned models fall short of RAG@32k's performance, indicating that compression still incurs performance costs that fine-tuning only partially mitigates. Robustness patterns vary considerably by task: $R_{\text{AO}}(y)$ shows the best resilience on ∞Bench MC (6.8% drop vs. 13.9% base) but the worst on Loong Financial (24.6% drop vs. 20.8% base), while $R_{\text{ID+Q}}(y)$ maintains perfect performance on LB-v2 (0.0% drop) yet degrades substantially on ∞Bench QA (32.7% drop). Our findings demonstrate that while fine-tuning can improve task performance and reduce degradation under compression, current approaches yield moderate gains that remain highly task-specific, highlighting the need for more targeted optimization strategies for efficient long-context inference.

Document Ranking Analysis. To investigate whether performance improvements stem from enhanced selective attention, we measure document ranking quality using NDCG@10 at 32k context length (our training setup), ranking documents by cumulative attention scores and comparing against HELMET's relevance-based ordering.

Table 4 shows that the base model already achieves high NDCG@10 scores (82.0–84.7). Fine-tuning yields minimal improvements: $R_{\text{ID+Q}}(y)$ achieves the highest scores with only 0.5–0.7 point gains over the base. However, the Pearson correlation between attention ranking and task performance is negligible (r=-0.09, p=0.86): $R_{\text{ID+Q}}(y)$ achieves the best NDCG@10 but poor accuracy (72.9), while $R_{\text{AO}}(y)$ and $R_{\text{R+Judge}}(y)$ demonstrate strong performance (80.3, 78.6) despite comparable NDCG@10 scores. These findings suggest that improved attention-based document ranking does not explain the observed performance gains.

5 Conclusions

We investigated whether fine-tuning LCLMs to attend to relevant information improves in-context retrieval and enables efficient inference. While certain approaches achieve substantial in-domain improvements, out-of-domain generalization remains inconsistent, with no approach consistently outperforming RAG. We showed that fine-tuning reduces degradation under KV-cache compression, though gains remain moderate and highly task-dependent. Finally, our analysis reveals that attention-based document ranking improves minimally, and correlates weakly with task performance, suggesting that observed gains stem from mechanisms beyond simple attention reallocation to relevant content.

Limitations

We train our models on contexts up to 32k tokens both for computational efficiency and for evaluating generalization on longer inputs, up to 1M tokens. This design choice allows us to assess generalization to longer contexts but limits our ability to investigate whether training on extremely long sequences would yield different outcomes. Given sufficient computational resources, it would be valuable to extend training to 1M context lengths, though our observed generalization patterns suggest findings would remain largely consistent.

Moreover, our evaluation focuses on English-language benchmarks in question answering, multiple choice, summarization, and financial analysis domains. The effectiveness of these fine-tuning strategies for other languages, modalities, or task types remains an open question.

Finally, our training data is constructed from Wikipedia passages due to the availability of gold-standard annotations in HotpotQA and 2WikiMultihopQA. While this enables controlled experimentation with verifiable rewards, the lack of domain-specific annotated data (e.g., legal documents, scientific papers, technical manuals) limits our ability to assess whether domain-adapted training would improve out-of-domain generalization. Given sufficient annotated data containing gold documents for other domains, our approach could be extended to enable more robust cross-domain performance.

References

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

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024b. Longbench: A bilingual, multitask benchmark for long context understanding. *Preprint*, arXiv:2308.14508.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2025. 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, Vienna, Austria. Association for Computational Linguistics.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, Xiaoyi Dong, Haodong Duan, Qi Fan, Zhaoye Fei, Yang Gao, Jiaye Ge, Chenya Gu, Yuzhe Gu, Tao Gui, and 81 others. 2024. Internlm2 technical report. *Preprint*, arXiv:2403.17297.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 26 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Tianyu Gao, Alexander Wettig, Howard Yen, and Danqi Chen. 2025. How to train long-context language models (effectively). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7376–7399, Vienna, Austria. Association for Computational Linguistics.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, and 37 others. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *Preprint*, arXiv:2406.12793.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *Preprint*, arXiv:2011.01060.

Sam Jacobs, Masahiro Tanaka, Chengming Zhang, Minjia Zhang, Reza Aminabadi, Shuaiwen Song, Samyam Rajbhandari, and Yuxiong He. 2024. System optimizations for enabling training of extreme long sequence transformer models. pages 1206–1208.

Ziyan Jiang, Xueguang Ma, and Wenhu Chen. 2024. Longrag: Enhancing retrieval-augmented generation with long-context llms. *Preprint*, arXiv:2406.15319.

Bowen Jin, Jinsung Yoon, Jiawei Han, and Sercan O. Arik. 2024. Long-context llms meet rag: Overcoming challenges for long inputs in rag. *Preprint*, arXiv:2410.05983.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Jinhyuk Lee, Anthony Chen, Zhuyun Dai, Dheeru Dua, Devendra Singh Sachan, Michael Boratko, Yi Luan, Sébastien M. R. Arnold, Vincent Perot, Siddharth Dalmia, Hexiang Hu, Xudong Lin, Panupong Pasupat, Aida Amini, Jeremy R. Cole, Sebastian Riedel, Iftekhar Naim, Ming-Wei Chang, and Kelvin Guu. 2024. Can long-context language models subsume retrieval, rag, sql, and more? *Preprint*, arXiv:2406.13121.

Xinze Li, Yixin Cao, Yubo Ma, and Aixin Sun. 2024a. Long context vs. rag for llms: An evaluation and revisits. *Preprint*, arXiv:2501.01880.

Zhuowan Li, Cheng Li, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. 2024b. Retrieval augmented generation or long-context LLMs? a comprehensive study and hybrid approach. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 881–893, Miami, Florida, US. Association for Computational Linguistics.

Di Liu, Meng Chen, Baotong Lu, Huiqiang Jiang, Zhenhua Han, Qianxi Zhang, Qi Chen, Chengruidong Zhang, Bailu Ding, Kai Zhang, Chen Chen, Fan Yang, Yuqing Yang, and Lili Qiu. 2024. Retrievalattention: Accelerating long-context llm inference via vector retrieval. *Preprint*, arXiv:2409.10516.

Francesco Maria Molfese, Luca Moroni, Luca Gioffré, Alessandro Scirè, Simone Conia, and Roberto Navigli. 2025. Right answer, wrong score: Uncovering the inconsistencies of llm evaluation in multiple-choice question answering. *Preprint*, arXiv:2503.14996.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2023. Yarn: Efficient context window extension of large language models. *Preprint*, arXiv:2309.00071.

Yifu Qiu, Varun R. Embar, Yizhe Zhang, Navdeep Jaitly, Shay B Cohen, and Benjamin Han. 2025. Eliciting in-context retrieval and reasoning for long-context large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3176–3192, Vienna, Austria. Association for Computational Linguistics.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

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

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. 2023. Roformer: Enhanced transformer with rotary position embedding. *Preprint*, arXiv:2104.09864.

Minzheng Wang, Longze Chen, Fu Cheng, Shengyi Liao, Xinghua Zhang, Bingli Wu, Haiyang Yu, Nan Xu, Lei Zhang, Run Luo, Yunshui Li, Min Yang, Fei Huang, and Yongbin Li. 2024. Leave no document behind: Benchmarking long-context LLMs with extended multi-doc QA. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5627–5646, Miami, Florida, USA. Association for Computational Linguistics.

Wenhan Xiong, Jingyu Liu, Igor Molybog, Hejia Zhang, Prajjwal Bhargava, Rui Hou, Louis Martin, Rashi Rungta, Karthik Abinav Sankararaman, Barlas Oguz, Madian Khabsa, Han Fang, Yashar Mehdad, Sharan Narang, Kshitiz Malik, Angela Fan, Shruti Bhosale, Sergey Edunov, Mike Lewis, and 2 others. 2024. Effective long-context scaling of foundation models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4643–4663, Mexico City, Mexico. Association for Computational Linguistics.

Peng Xu, Wei Ping, Xianchao Wu, Lawrence McAfee, Chen Zhu, Zihan Liu, Sandeep Subramanian, Evelina Bakhturina, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Retrieval meets long context large language models. *Preprint*, arXiv:2310.03025.

Peng Xu, Wei Ping, Xianchao Wu, Chejian Xu, Zihan Liu, Mohammad Shoeybi, and Bryan Catanzaro. 2025. Chatqa 2: Bridging the gap to proprietary llms in long context and rag capabilities. *Preprint*, arXiv:2407.14482.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *Preprint*, arXiv:1809.09600.

Howard Yen, Tianyu Gao, Minmin Hou, Ke Ding, Daniel Fleischer, Peter Izsak, Moshe Wasserblat, and Danqi Chen. 2025. Helmet: How to evaluate long-context language models effectively and thoroughly. *Preprint*, arXiv:2410.02694.

Qingchen Yu, Zifan Zheng, Shichao Song, Zhiyu Li, Feiyu Xiong, Bo Tang, and Ding Chen. 2025. xfinder: Large language models as automated evaluators for reliable evaluation. *Preprint*, arXiv:2405.11874.

Tan Yu, Anbang Xu, and Rama Akkiraju. 2024. In defense of rag in the era of long-context language models. *Preprint*, arXiv:2409.01666.

Tianjun Zhang, Shishir G. Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E. Gonzalez. 2024a. Raft: Adapting language model to domain specific rag. *Preprint*, arXiv:2403.10131.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Hao, Xu Han, Zhen Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. 2024b. ∞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.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *Preprint*, arXiv:2506.05176.

A Related Work

Long Context vs. RAG. Recent work has explored the trade-offs between LCLMs and RAG, revealing inconsistent findings across different settings (Li et al., 2024a). While some studies suggest that retrieval helps models with smaller context windows (4k) but not longer ones (16k-32k) (Bai et al., 2024b), others demonstrate that RAG consistently outperforms LCLMs at context lengths up to 32k (Xu et al., 2024, 2025; Li et al., 2024b; Yu et al., 2024). Conversely, Bai et al. (2025) show that LCLMs can perform better without RAG at context lengths up to 128k, while Jiang et al. (2024) demonstrate that RAG benefits from longer retrieval units. These mixed findings indicate ongoing debate about whether LCLMs can effectively replace RAG or whether retrieval remains essential for long-context understanding.

Fine-tuning LCLMs. Several approaches have attempted to improve long-context capabilities through fine-tuning on question answering tasks with distractor documents (Qiu et al., 2025; Zhang et al., 2024a; Jin et al., 2024). RAFT (Zhang et al., 2024a) trains models to ignore irrelevant documents when answering questions, while Qiu et al. (2025) use challenging distractors from strong retrievers and require models to generate both document IDs and content. However, both focus exclusively on supervised fine-tuning (SFT) and evaluate primarily on in-domain RAG benchmarks, limiting insights into out-of-domain generalization and whether models genuinely improve attention to relevant content. Jin et al. (2024) explore answer-only training and reasoning distillation from stronger models, demonstrating that SFT with reasoning improves performance, though their analysis remains limited to in-domain settings.

Our work extends this line of research in three key directions: (1) we explore reinforcement learning (GRPO) beyond standard SFT, with and without reasoning objectives; (2) we evaluate systematically on both in-domain and out-of-domain benchmarks; and (3) we analyze the mechanisms underlying performance improvements through attention-based document ranking and KV-cache compression robustness, rather than focusing solely on downstream task metrics.

B Training Data Details

Hard Negative Refinement. Since retrieving hard negatives from Wikipedia can inadvertently include documents that are actually relevant to the question, we apply a two-stage preprocessing pipeline to identify and promote such documents from hard negatives to relevant passages. In the first stage, we apply fuzzy matching heuristics, including Jaccard similarity, character-level F1, and n-gram matching, between each gold passage and each hard negative, using a loose similarity threshold. This ensures high recall, capturing the vast majority of true positives within the input context. However, this approach can introduce false positives; therefore, in the second stage, we employ a strong LLM as a judge to filter out promoted documents that are indeed false positives. This two-stage procedure results in a refined set of truly relevant documents, increasing the average number of relevant passages from 2 to approximately 4 per instance. This refinement makes the model more resilient to out-of-domain cases where more than 2 passages may be needed to answer the question.

In our experiments, we use a fuzzy similarity threshold of 0.6 for the initial matching stage. For the second stage, we employ DeepSeek-R1-Distill-Qwen-32B (DeepSeek-AI et al., 2025) with thinking enabled and temperature set to 0.0 as the judge to filter false positives.

Training Data Examples. Table 5 shows representative instances from our training data, highlighting the challenging nature of the task. Each example contains a question paired with a long context comprising multiple documents, where only a small subset (indicated by Gold IDs and shown in bold) contains information relevant to answering the question. The majority of documents serve as hard negatives, which are topically related but irrelevant passages that create a realistic information retrieval scenario. For instance, in the first example, while many documents discuss various magazines, only documents 10 and 11 contain the specific founding dates needed to determine which magazine was started first. Similarly, the second example includes multiple documents about Indian hotel chains and cities, but only documents 0 and 4 provide the critical information linking the Oberoi family to Delhi. We posit that this sparse relevant information embedded within extensive hard negatives may force models to develop selective attention mechanisms, mimicking the challenges

faced by RAG systems when processing retrieved contexts.

C Reward Functions

We define the following reward functions for model output y:

Answer Only (AO). The model receives a reward for generating the correct answer:

$$R_{\text{AO}}(y) = \mathbb{1}[\text{ans}(y) \approx a^*] \quad (6)$$

where $\text{ans}(y)$ extracts the answer from output y, and $\approx$ denotes sub-exact match allowing for minor lexical variations.

Document IDs + Answer (ID). The model is rewarded for correctly identifying relevant document IDs and providing the correct answer:

$$R_{\text{ID}}(y) = \mathbb{1}[\text{ids}(y) = \mathcal{D}^*] + \mathbb{1}[\text{ans}(y) \approx a^*] \quad (7)$$

where $\text{ids}(y)$ extracts the cited document indices from the output.

Document IDs + Content + Answer (ID+C). The model is rewarded for generating the IDs and reproducing the content of relevant documents:

$$R_{\text{ID+C}}(y) = \mathbb{1}[\text{ids}(y) = \mathcal{D}^*] + \mathbb{1}[\text{cont}(y) \approx \{d_i : i \in \mathcal{D}^*\}] + \mathbb{1}[\text{ans}(y) \approx a^*] \quad (8)$$

where $\text{cont}(y)$ extracts the document content reproduced in the output.

Document IDs + Quotes + Answer (ID+Q). The model is rewarded for identifying relevant documents and extracting pertinent quotes:

$$R_{\text{ID+Q}}(y) = \mathbb{1}[\text{ids}(y) = \mathcal{D}^*] + \mathbb{1}[\text{quotes}(y) \subseteq \{d_i : i \in \mathcal{D}^*\}] + \mathbb{1}[\text{ans}(y) \approx a^*] \quad (9)$$

where $\text{quotes}(y)$ extracts quoted text spans from the output.

Reasoning + LLM-as-a-Judge (R+Judge). The policy is prompted to reason over the context and justify its answer. We parse the output to extract cited document IDs $\mathcal{D}_{\text{cited}}$ and employ an LLM judge J to evaluate relevance:

$$R_{\text{R+Judge}}(y) = J(\text{reason}(y), \mathcal{D}_{\text{cited}}, \mathcal{D}^*, q, c) + \mathbb{1}[\text{ans}(y) \approx a^*] \quad (10)$$

The judge J assigns a score based on whether cited documents are in $\mathcal{D}^*$ or contain information genuinely useful for answering the question. This relaxed reward addresses the limitation that annotated relevant documents may be incomplete, as topically related documents can provide valid reasoning paths.

D Training Settings

We train all models using GRPO with the VERL framework (Sheng et al., 2024) and FSDP for distributed training. We split the data into 95% train and 5% development, resulting in 28,500 training samples and 1,500 validation samples. The training batch size is set to 256 and validation batch size to 128. We set the maximum prompt length to 32,768 tokens to accommodate long contexts, with a maximum response length of 2,048 tokens, applying middle truncation for sequences exceeding these limits. We use the AdamW optimizer with a learning rate of 1×10^{-6}, a PPO mini-batch size of 64, and a micro-batch size of 2 per GPU. Training proceeds for up to 2 epochs with model checkpointing every 10 steps. For GRPO, we sample $G = 5$ outputs from the current policy for each training instance to compute group-relative advantages. The KL divergence coefficient β is set to 0.001 with a low-variance KL loss formulation, applied only in the loss function and not in the reward computation itself. To optimize memory usage, we enable gradient checkpointing and activation offloading, with Ulysses sequence parallelism (Jacobs et al., 2024) set to 4 and tensor model parallelism of 2 for rollout generation. We use vLLM for efficient rollout generation with chunked prefill enabled and a maximum of 34,816 batched tokens, while reference model computations use FSDP with parameter offloading. We cap GPU memory utilization at 60%. In Table 6 we provide the judge prompt used to train our models with the reward function $R_{\text{R+Judge}}(y)$ (see Section 2 for details).

E Evaluation Prompts

Table 7 presents the prompt templates used for evaluation across all out-of-domain benchmarks. All prompts are taken directly from the official benchmark papers. The `{instruction}` component in the Loong financial split can be found in the original paper (Wang et al., 2024). For the in-domain benchmark HELMET, each model trained with a different reward function is evaluated using a dis-

Question	Answer	Training Context (abbreviated)	Gold IDs
Which magazine was started first Arthur's Magazine or First for Women?	Arthur's Magazine	[DOC 0] British art critics ... [DOC 1] The English Woman's Journal ... [DOC 2] 1300 subscribers ... [DOC 3] Magazines published in UK ... [DOC 4] French male writers ... [DOC 5] Published in USA ... [DOC 6] Magazines with year ... [DOC 7] Feminism and family ... [DOC 8] UK Monthly magazines ... [DOC 9] (magazine) ... **[DOC 10] Arthur's Magazine (1844–1846) American literary periodical ... [DOC 11] First for Women ... woman's magazine published by Bauer Media ...**	10, 11
The Oberoi family is part of a hotel company that has a head office in what city?	Delhi	**[DOC 0] The Oberoi Group is a hotel company with its head office in Delhi.** [DOC 1] Taj Hotels ... headquartered in Mumbai ... [DOC 2] Natural History Society ... [DOC 3] Taj Dubai ... **[DOC 4] The Oberoi family is an Indian family involved in hotels through The Oberoi Group.** [DOC 5] Food and drink companies based in Boston ... [DOC 6] City serves as headquarters ... Kolkata ... [DOC 7] Bangalore, Taj Connemara ... [DOC 8] Observer Research Foundation based in Delhi ... [DOC 9] Companies based in Miami ... [DOC 10] Survey of India ... Kolkata ...	0, 4
Musician and satirist Allie Goertz wrote a song about the "The Simpsons" character Milhouse, who Matt Groening named after who?	Richard Nixon	[DOC 0] Simpson ... Harry Shearer as Mr. Burns ... [DOC 1] Jon Lovitz as Artie Ziff ... **[DOC 2] Allison Beth "Allie" Goertz (born March 2, 1991) American musician.** [DOC 3] Krusty the Clown ... Bart ... **[DOC 4] Her videos are posted on YouTube under Cossbysweater.** [DOC 5] Simpsons ... Jerry Nelson ... **[DOC 6] Goertz is known for her satirical songs based on pop culture topics.** [DOC 7] Ralph Wiggum ... **[DOC 8] Milhouse Mussolini van Houten ... fictional character ... created by Matt Groening ...** [DOC 9] Guitar ... parody ... [DOC 10] by "Weird Al" Yankovic ... [DOC 11] Beverly Hills 90210 ... [DOC 12] Baby bunnies ... [DOC 13] Allegra's Window ...	2, 4, 6, 8

Table 5: Examples of training data instances showing questions with long-context inputs containing both relevant documents (bold) and hard negative distractors. Gold IDs indicate the document indices required to answer each question.

tinct prompt structure that matches its training objective. Tables 8 and 9 present the complete prompt templates for all five reward functions applied to the RAG split of HELMET. These prompts are designed to elicit the specific output formats required for computing rewards during training, ensuring consistency between training and evaluation.

F Metrics

We employ benchmark-specific evaluation metrics tailored to each task type. For multiple-choice question answering benchmarks (LB-v2 and ∞Bench MC), we use accuracy as the primary metric. To extract model answers, we employ xFinder (Yu et al., 2025), an LLM-based answer extraction tool that has demonstrated higher agreement with human judgment compared to standard regex-based extractors (Molfese et al., 2025). For open-ended question answering tasks (∞Bench QA and HELMET), we use sub-exact match following the recommendation of Yen et al. (2025), who demonstrate that this metric better correlates with true model performance than strict exact match. Sub-exact match allows for minor lexical variations while still requiring the core answer to be correct. For the summarization task in ∞Bench (SUM split), we use Rouge-L to measure the overlap between generated summaries and reference summaries. For the Loong financial split, we follow the official evaluation protocol and use an LLM judge to score model outputs on a scale from 0 to 100. The judge prompt template is provided in the original paper (Wang et al., 2024) and evaluates responses based on correctness and reasoning quality over financial documents. We employ DeepSeek-R1-Distill-Qwen-32B (DeepSeek-AI et al., 2025) with thinking enabled and temperature set to 0.0 as judge.

G Model Selection

To inform our choice of base model for training experiments, we conduct a preliminary evaluation comparing several state-of-the-art long-context lan-

guage models against their RAG-enhanced counterparts on out-of-domain benchmarks.

Models. We evaluate the following models: Qwen2.5-7B-Instruct (128k context), Qwen2.5-7B-Instruct-1M (1M context), glm-4-9b-chat (128k context), glm-4-9b-chat-1m (1M context), and internlm2_5-7b-chat-1m (1M context) (Qwen et al., 2025; GLM et al., 2024; Cai et al., 2024). For RAG baselines, we employ Qwen3-Embedding-4B as the retriever (Zhang et al., 2025). When inputs exceeds the model's supported context window, we apply truncation at the end of the input context.

Benchmarks. We assess performance on three long-context benchmarks: ∞Bench (Zhang et al., 2024b) (MC, QA, and SUM splits), LB-v2 (Bai et al., 2025), and the English financial split of Loong (Wang et al., 2024). These benchmarks provide diverse evaluation scenarios spanning multiple-choice questions, open-ended QA, summarization, and numerical reasoning tasks.

RAG Configuration. Following recent findings that advocate for longer chunking strategies with LCLMs (Jiang et al., 2024), we chunk each document into 2048-token segments. For each benchmark instance, we retrieve passages to construct contexts of varying lengths $k \in \{4\text{k}, 8\text{k}, 16\text{k}, 32\text{k}, 64\text{k}, 128\text{k}, 1\text{M}\}$, allowing us to analyze performance degradation as context length increases and identify when RAG provides advantages over full-context processing.

Results and Analysis. Table 10 presents the comparative results. We observe that RAG consistently outperforms full-context processing for most models, with optimal performance typically achieved at moderate retrieval lengths (32k-128k tokens). However, the results reveal important nuances: while Qwen2.5-7B-Instruct-1M shows substantial improvements with RAG (e.g., 78.6% on ∞Bench MC with 32k RAG vs. 72.0% full context), glm-4-9b-chat-1m demonstrates the opposite pattern, achieving its best performance with full-context processing on several tasks (37.3% on ∞Bench QA vs. 35.3% best RAG performance). The performance gap is particularly pronounced on the challenging Loong financial split, where RAG configurations frequently outperform full-context processing by 2-5 points for Qwen models. These mixed results suggest that while some models struggle to effectively attend to relevant information within extended contexts, others have developed stronger selective attention capabilities during pre-training.

Based on these findings, we select Qwen2.5-7B-Instruct-1M as our base model for subsequent training experiments. This choice is motivated by three key factors: (1) it exhibits the clearest and most consistent performance gap between full-context and RAG settings (6.6 points on ∞Bench MC, 3.6 points on ∞Bench QA, and 4.8 points on Loong), indicating substantial room for improvement through targeted training on selective attention; (2) unlike glm-4-9b-chat-1m which already demonstrates strong full-context capabilities, Qwen2.5-7B-Instruct-1M represents models that genuinely benefit from retrieval mechanisms, making it an ideal testbed for our hypothesis; and (3) its 1M context window allows us to evaluate whether models trained on moderate-length contexts (16k-32k) can generalize to much longer sequences at inference time.

H Detailed Results

In-Domain Results. Table 11 presents performance on HELMET's retrieval subsets across varying context lengths. The $R_{\text{AO}}(y)$ reward function achieves the strongest and most consistent improvements, with average gains of 20.1 points on HotpotQA (81.1 vs. 61.0), 6.6 points on NQ (67.7 vs. 61.1), and 5.5 points on TriviaQA (92.1 vs. 86.6) over the base model. Notably, $R_{\text{AO}}(y)$ demonstrates superior robustness to context length scaling, maintaining 74.3% accuracy on HotpotQA at 128k tokens compared to 52.3% for the base model—a 22-point advantage at the longest context length.

The $R_{\text{ID}}(y)$ reward function also yields substantial gains, averaging 75.2 on HotpotQA and 91.0 on TriviaQA. However, strategies requiring more constrained outputs—specifically $R_{\text{ID+C}}(y)$ and $R_{\text{ID+Q}}(y)$—show marked performance degradation as context length increases. For instance, $R_{\text{ID+C}}(y)$ achieves 90.2% at 4k on TriviaQA but drops to 83.5% at 128k, performing worse than the base model at longer contexts. This pattern suggests that overly restrictive training objectives may impede the model's ability to process extended contexts effectively.

Comparing against the upper bound (w/ gold docs) and lower bound (w/o context), we observe that fine-tuned models not only substantially improve over the base, but they also surpass the performance obtained when the relevant documents are

provided, particularly on HotpotQA (81.1 vs. 72.2 gold docs) and TriviaQA (92.1 vs. 85.8 gold docs), indicating that the training successfully enables selective attention without sacrificing the model's ability to leverage its full context when beneficial.

Out-of-Domain Results. Table 12 evaluates generalization to out-of-domain benchmarks where inputs consist of long, cohesive documents rather than retrieved passages. Results reveal mixed patterns across tasks. The $R_{\text{R+Judge}}(y)$ approach demonstrates the most substantial improvement on the Loong financial split, achieving 58.8 (+8.9 over base), while $R_{\text{AO}}(y)$ yields the largest gain on ∞Bench QA (38.2, +8.2 points). However, improvements are modest or absent on other benchmarks: multiple-choice accuracy remains largely unchanged (70.3–72.5 vs. 72.0 base), and summarization shows minimal variation (29.7–31.7 vs. 31.2 base).

Notably, RAG baselines at moderate retrieval lengths (32k–128k tokens) frequently outperform all fine-tuned full-context models. For instance, RAG@32k achieves 78.6% on ∞Bench MC compared to 72.5% for the best fine-tuned model ($R_{\text{R+Judge}}(y)$). This gap is particularly pronounced on LB-v2, where RAG@32k scores 38.0% versus 32.6% for the best full-context model. These results suggest that while fine-tuning improves selective attention for in-domain retrieval scenarios, it provides limited advantages for tasks requiring holistic document understanding—domains where RAG's filtering mechanism retains its efficacy.

LLM Judge Evaluation Prompt Template

```
You are an expert evaluator assessing AI model answers to questions using supporting documents.
You will be provided with:
- A Question
- A set of Relevant Documents (the gold standard grounding sources)
- The Correct Answer
- An AI Model Solution

Background:
The AI model had access to a large pool of documents (indexed 0...N). Only a subset is truly
relevant. Other documents may appear in citations but are simply distractors (not fabricated).
The model's goal is to correctly answer the question while grounding its reasoning in the
relevant documents.

Your task:
Evaluate the model's solution objectively and consistently according to the criteria below.
Do not use information outside the provided inputs.

—

[Question]: {question}
[Relevant Documents]: {gold_docs}
[Correct Answer]: {answer}
[AI Model Solution]: {solution}

—

EVALUATION CRITERIA

Criterion 1: Reasoning Quality (1 or 0)
Score 1 if the solution shows: clear logical flow from evidence to conclusion, no
contradictions or fallacies, and coherent, well-structured reasoning.
Score 0 if the reasoning is flawed, contradictory, or incoherent.

Criterion 2: Document Grounding (1 or 0)
Score 1 if the solution: uses information primarily from relevant documents, represents
those documents accurately (no distortions), and does not rely significantly on irrelevant or
external knowledge. Score 0 if it misuses documents or ignores relevant evidence.

Criterion 3: Answer Correctness (1 or 0)
Score 1 if the final answer matches the provided correct answer.
Score 0 otherwise (including partial or incomplete answers).

—

RESPONSE FORMAT
For each criterion, provide a 1-2 sentence justification followed by the score:

Reasoning Quality Justification: [Your explanation]
\boxed{Criterion 1: 1 or 0}

Document Grounding Justification: [Your explanation]
\boxed{Criterion 2: 1 or 0}

Answer Correctness Justification: [Your explanation]
\boxed{Criterion 3: 1 or 0}
```

Table 6: Prompt template for LLM judge evaluation in the R+Judge reward function. The judge evaluates model outputs across three binary criteria: reasoning quality, document grounding, and answer correctness. The final reward is computed as the sum of all three scores.

Benchmark	Prompt Template
∞Bench	

QA Split	`System: You are an expert in answering questions about books.`
	`User: Read the book below and answer the question.`
	
	`{context}`
	
	`Question: {question}`
	
	`Format your response as follows: "The answer is (insert answer here)".`

SUM Split	`System: You are an expert in summarizing books.`
	`User: Summarize the book below.`
	
	`{context}.`

MC Split	`System: You are an expert in multiple-choice question answering about books.`
	`User: Read the book below and answer the question.`
	
	`{context}`
	
	`Question: {question}`
	
	`Only one of the following options is correct:`
	
	`A. {choice_a}`
	`B. {choice_b}`
	`C. {choice_c}`
	`D. {choice_d}`
	
	`Format your response as follows: "The answer is (insert answer here)".`

LB-v2 (ALL split)	

ALL Split	`System: You are an expert in multiple-choice question answering.`
	`User: Please read the following text and answer the question below.`
	
	`<text>`
	`{DOC}`
	`</text>`
	
	`What is the correct answer to this question: {Q}`
	`Choices:`
	`(A) {C_A}`
	`(B) {C_B}`
	`(C) {C_C}`
	`(D) {C_D}`
	
	`Format your response as follows: "The correct answer is (insert answer here)".`

Loong	

Financial Split	`System: You are an expert in answering questions about financial reports.`
	`User: {context}`
	
	`{instruction}`
	
	`{question}.`

Table 7: Prompt templates used for evaluation across out-of-domain benchmarks. All prompts follow a consistent format with system instructions defining the task domain and user prompts providing context and questions. Variables enclosed in braces (e.g., `{context}`, `{question}`) are replaced with actual benchmark data during evaluation.

Reward Function	Prompt Template (KILT split)
Answer Only (AO)	System: You are an expert in question answering. User: Use the given documents to write a concise and short answer to the question. {context} Question: {question} Write your concise and short answer in the following format: Answer: <your answer>.
ID + Answer (ID)	System: You are an expert in question answering. User: Documents: {context} Instruction: Given the set of documents above, answer the following question by following these steps: 1. Identify the relevant documents. Each document is identified by a tag in the form [DOC i], where i is the index of the document in the context. - If you find relevant documents, output their IDs exactly as shown, separated by commas. Example: [DOC i], [DOC j], [DOC k]. - If no documents are relevant, output only: [DOC -1]. 2. On a new line, provide your answer in the following format: The answer is: <your answer here>. Question: {question}.
ID + Content + Answer (ID+C)	System: You are an expert in question answering. User: Documents: {context} Instruction: You are given a set of documents and a question. Your task is to: 1. Identify which documents are most relevant to answering the question 2. Extract and reproduce the IDs and the full content of those relevant documents 3. Provide the final answer based on the relevant documents Follow this exact format in your response: Relevant documents: [DOC X] <full content of first relevant document> [DOC Y] <full content of second relevant document> (continue for all relevant documents) The answer is: <your final answer> Important guidelines: - Only include documents that directly help answer the question - Reproduce the ID and the complete content of each relevant document exactly - The final answer should be concise and directly address the question Question: {question}.

Table 8: HELMET prompt templates: Part 1.

Reward Function	Prompt Template (KILT split)
ID + Quote + Answer (ID+Q)	System: You are an expert in question answering. User: Documents: {context} Instruction: Given the set of documents above, answer the following question by following these steps: 1. Extract relevant quotes: Find and extract short quotes or passages ($\leq$30 tokens each) from the documents that help answer the question. Present each quote in the following format: Quote 1: "<exact text from document>" Quote 2: "<exact text from document>" (Continue as needed) 2. List the source documents using this exact format: Relevant Document IDs: [DOC i], [DOC j], [DOC k] (Where i, j, k are the indices of documents that contain your selected quotes) If no documents are relevant, use: Relevant Document IDs: [DOC -1] 3. Provide your final answer in the following format: The answer is: <your answer here> Important: Keep quotes short ($\leq$30 tokens), select only the most relevant passages, and ensure your document IDs correspond to the documents containing your quotes. Question: {question}.
Reasoning + Judge (R+Judge)	System: You are an expert in question answering. User: Documents: {context} Instruction: You are given a set of documents and a question. Your task is to analyze all documents, reason through the question step-by-step, and provide a well-grounded answer. Follow this exact format in your response: **Step 1: Question Analysis** Break down what the question is asking and identify the key information needed. **Step 2: Document Review** Briefly assess which documents contain relevant information for answering the question. Consider how each document relates to the question's requirements. **Step 3: Reasoning** Work through the problem step-by-step using information from the relevant documents. For each reasoning step, clearly reference which document(s) support your logic. Use the format: 'According to [DOC X], ...' or 'Document Y states that ...' to ground your reasoning in the provided sources. **Step 4: Answer** The answer is: <your final answer> Important guidelines: - Show your complete reasoning process - Ground each reasoning step in specific document evidence - Clearly reference document IDs when using information from them - Keep your final answer concise but ensure it directly addresses the question Question: {question}

Table 9: HELMET prompt templates: Part 2.

Setting	Model	Ctx. Len.	Max. Input Len.	∞Bench MC	∞Bench QA	∞Bench Sum	LB-V2 ALL	Loong Fin.
Full Context								
	Qwen2.5-7B-Instruct	128k	128k	64.2	27.1	31.9	34.9	44.8
	Qwen2.5-7B-Instruct-1M	1M	1M	72.0	30.0	31.2	32.0	49.9
	glm-4-9b-chat	128k	128k	70.3	36.2	27.9	29.1	49.1
	glm-4-9b-chat-1m	1M	1M	77.3	37.3	27.7	30.4	49.6
	internlm2_5-7b-chat-1m	1M	1M	71.6	36.7	20.0	26.0	32.9
RAG								
	Qwen2.5-7B-Instruct	128k	128k	72.1	28.2	31.1	32.7	49.2
			64k	72.5	31.9	32.2	32.8	49.9
			32k	74.7	29.6	30.5	34.8	52.0
			16k	74.7	29.3	27.1	35.8	48.6
			8k	69.0	25.1	22.4	32.2	46.8
			4k	57.6	19.1	19.3	28.4	35.9
	Qwen2.5-7B-Instruct-1M	1M	1M	69.1	29.1	29.1	30.4	50.7
			128k	74.2	33.6	29.1	35.2	54.7
			64k	77.7	30.8	26.8	35.0	52.8
			32k	78.6	29.3	25.1	38.0	47.8
			16k	74.7	27.6	25.2	35.2	54.9
			8k	71.2	24.2	23.8	32.4	47.9
			4k	59.8	18.8	19.4	31.6	39.6
	glm-4-9b-chat	128k	128k	75.5	33.6	26.2	31.1	50.7
			64k	74.7	32.8	23.9	31.4	48.8
			32k	76.4	29.6	22.8	30.2	45.3
			16k	71.6	27.3	20.6	30.4	43.4
			8k	68.5	21.9	19.5	30.4	43.3
			4k	54.1	15.4	15.9	29.0	31.9
	glm-4-9b-chat-1m	1M	1M	74.7	33.3	27.2	31.8	47.6
			128k	75.5	35.3	25.9	34.0	47.3
			64k	77.3	31.3	25.1	32.6	46.1
			32k	74.7	28.2	23.1	32.8	44.7
			16k	76.4	25.1	21.6	31.8	44.1
			8k	69.4	21.9	20.1	31.6	43.1
			4k	55.0	17.1	17.3	27.2	31.7
	internlm2_5-7b-chat-1m	1M	1M	65.5	36.7	18.2	25.6	34.9
			128k	69.4	35.3	17.8	26.4	37.1
			64k	72.0	35.0	16.7	25.6	38.9
			32k	70.7	31.3	15.9	29.6	38.9
			16k	72.5	27.6	15.7	28.0	40.7
			8k	67.2	25.1	15.1	28.8	36.6
			4k	57.6	17.7	12.8	29.2	30.9

Table 10: Comparative performance of long-context models in Full Context and RAG settings. MC: Multiple Choice (Accuracy ↑), QA: Question Answering (Sub-Exact Match ↑), Sum: Summarization (Rouge-L ↑), ALL: Overall LB-v2 (Accuracy ↑), Fin.: Financial split from Loong (LLM Judge Score ↑). Ctx. Len.: Maximum context window. Max. Input Len.: Maximum number of tokens provided to the model for inference. Bold indicates best performance within each model group.

Subset	Model	4k	8k	16k	32k	64k	128k	Avg.
HotpotQA (SubEM ↑, 300 instances)								
	Qwen2.5-7B-Instruct-1M	70.3	68.7	64.3	57.3	53.0	52.3	61.0
	w/ gold docs	76.3	76.0	73.3	73.0	71.3	63.0	72.2
	w/o context	27.0	27.0	27.0	27.0	27.0	27.0	27.0
	+ $R_{\text{AO}}(y)$	85.3	**84.3**	**85.0**	**80.0**	**77.7**	**74.3**	**81.1**
	+ $R_{\text{ID}}(y)$	83.7	81.0	77.0	75.0	70.3	64.0	75.2
	+ $R_{\text{ID+C}}(y)$	**86.0**	74.7	69.3	60.7	60.7	56.0	67.9
	+ $R_{\text{ID+Q}}(y)$)	84.7	80.0	76.3	74.0	69.7	60.0	74.1
	+ $R_{\text{R+Judge}}(y)$	84.3	78.0	75.3	66.7	62.3	57.0	70.6
NaturalQuestions (SubEM ↑, 600 instances)								
	Qwen2.5-7B-Instruct-1M	63.7	63.7	61.5	60.7	60.0	57.0	61.1
	w/ gold docs	73.0	74.7	73.8	71.5	66.8	64.3	70.7
	w/o context	32.5	32.5	32.5	32.5	32.5	32.5	32.5
	+ $R_{\text{AO}}(y)$	70.2	69.2	67.7	67.5	66.8	**64.8**	67.7
	+ $R_{\text{ID}}(y)$	70.3	66.8	66.3	65.8	62.2	61.5	65.5
	+ $R_{\text{ID+C}}(y)$	66.2	62.7	55.2	55.5	52.7	49.2	56.9
	+ $R_{\text{ID+Q}}(y)$	71.2	63.5	59.8	58.3	53.7	48.2	59.1
	+ $R_{\text{R+Judge}}(y)$	**80.8**	**77.3**	**74.7**	**72.0**	**67.2**	64.7	**72.8**
TriviaQA (SubEM ↑, 600 instances)								
	Qwen2.5-7B-Instruct-1M	84.8	86.8	87.2	87.7	87.2	85.8	86.6
	w/ gold docs	87.3	88.7	88.2	87.2	87.3	76.2	85.8
	w/o context	56.2	56.2	56.2	56.2	56.2	56.2	56.2
	+ $R_{\text{AO}}(y)$	89.2	90.0	**92.7**	**93.2**	**93.7**	**93.5**	92.1
	+ $R_{\text{ID}}(y)$	89.8	91.8	91.2	92.3	90.3	90.3	91.0
	+ $R_{\text{ID+C}}(y)$	90.2	86.3	85.5	84.3	85.2	83.5	85.8
	+ $R_{\text{ID+Q}}(y)$	87.8	86.5	83.0	83.7	86.7	85.0	85.5
	+ $R_{\text{R+Judge}}(y)$	**93.5**	**93.2**	92.0	92.7	92.3	89.8	**92.3**

Table 11: Performance on HELMET retrieval subsets across varying context lengths. Base: Qwen2.5-7B-Instruct-1M. "w/ gold docs" gives the model explicit information about which documents are relevant; "w/o context" provides no context (evaluates parametric knowledge). Bold indicates best performance per column.

Setting	Model	Max Len	∞Bench			LB-v2	Loong
			MC (Acc. ↑)	QA (SubEM ↑)	Sum (R-L ↑)	ALL (Acc. ↑)	Fin. (Judge ↑)
Full Context (unaltered)							
	Qwen2.5-7B-Instruct-1M	1M	72.0	30.0	31.2	32.0	49.9
	+ $R_{\text{AO}}(y)$	1M	70.3	**38.2**	31.0	31.2	55.3
	+ $R_{\text{ID}}(y)$	1M	71.6	34.7	29.7	31.0	51.2
	+ $R_{\text{ID+C}}(y)$	1M	70.7	31.3	30.2	31.4	47.5
	+ $R_{\text{ID+Q}}(y)$	1M	70.7	30.8	30.5	32.0	51.5
	+ $R_{\text{R+Judge}}(y)$	1M	72.5	30.7	**31.7**	32.6	**58.8**
KV-Cache with RetrievalAttn							
	Qwen2.5-7B-Instruct-1M	1M	62.0	27.6	27.3	30.2	39.5
	+ $R_{\text{AO}}(y)$	1M	65.5	33.6	28.0	29.6	41.7
	+ $R_{\text{ID}}(y)$	1M	64.2	29.3	26.0	30.0	43.0
	+ $R_{\text{ID+C}}(y)$	1M	63.3	27.1	27.7	29.6	39.8
	+ $R_{\text{ID+Q}}(y)$	1M	62.9	27.1	28.1	32.0	39.7
	+ $R_{\text{R+Judge}}(y)$	1M	63.8	28.2	28.8	31.2	41.5
RAG							
	Qwen2.5-7B-Instruct-1M	1M	69.1	29.0	29.1	30.4	50.7
		128k	74.2	33.6	29.1	35.2	54.7
		64k	77.7	31.0	26.8	35.0	52.8
		32k	**78.6**	29.3	25.1	**38.0**	47.8
		16k	74.7	27.6	25.2	35.2	54.9
		8k	71.2	24.2	23.8	32.4	47.9
		4k	59.8	18.8	19.4	31.6	39.6

Table 12: Performance comparison across different training strategies and inference settings. Numbers in parentheses indicate the number of test instances. Bold values indicate best performance within each major setting group. MC: Multiple Choice (Accuracy ↑), QA: Question Answering (SubEM ↑), Sum: Summarization (Rouge-L ↑), ALL: Overall (Accuracy ↑), Fin.: Financial (Judge Score ↑).