HIGHLIGHT-THEN-SUMMARIZE: LEARNING TO COMPRESS EVIDENCE FOR LONG-CONTEXT UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-context understanding is a fundamental capability for large language models to reason over lengthy documents, multi-turn conversations, and code. However, large language models often struggle with irrelevant and redundant information in long contexts, where task-relevant evidence can be sparse and scattered across distant positions, limiting their ability to reliably solve long context tasks. To address this problem, we propose **Highlight-Then-Summarize (H2S)**, a compress-then-reason paradigm that identifies question-relevant evidence from long contexts and summarizes it into a focused representation, enabling LLMs to reason with less interference from irrelevant and redundant information. We construct H2S-Dataset, comprising 6,647 examples from 11 benchmark families with an average context length of 43.9K tokens, and introduce H2S-Bench for evaluation. Trained on H2S-Dataset, H2S-14B achieves an average score of 32.60 on H2S-Bench, outperforming Qwen3.8-27B by 10.17 points and achieving leading performance among open-source models. Further analysis shows that H2S-14B achieves the highest Evidence-Summary Quality (ESQ) score and retains 97.1% of its 16K-budget performance with only a 4K output budget, indicating that H2S produces higher-quality yet more compact reasoning. Our contributions are threefold: (1) a compress-then-reason paradigm, H2S, that focuses reasoning on relevant evidence while suppressing irrelevant context; (2) H2S-Dataset, a diverse training set comprising 6,647 examples from 11 benchmark families to support learning under the H2S paradigm; and (3) a reinforcement learning method, H2S-RL, with process-level rewards for evidence selection and summary construction. We hope this work provides a practical approach to improving long-context reasoning through evidence-focused summary. Code and H2S-Dataset will be released at `https://anonymous.4open.science/r/Highlight-Then-Summarize-3DF2`.

1 INTRODUCTION

Long-context understanding is increasingly important for tasks such as long-document question answering, multi-document reasoning, extended dialogue modeling, and repository-level code understanding Bai et al. (2024b); Jimenez et al. (2024). Despite substantial advances in the context-window capacity of state-of-the-art LLMs Guo et al. (2025); Team et al. (2026); OpenAI (2025), effective reasoning over long inputs remains difficult because task-relevant evidence is typically sparse, distributed across distant regions of the context, and embedded within substantially more non-essential content Liu et al. (2024); Hsieh et al. (2024). Beyond evidence grounding, successful reasoning further requires establishing query-conditioned relations among dispersed evidence, including entity associations, temporal dependencies, logical conditions, and numerical constraints. For example, resolving a repository-level bug may require tracing a failing API call across its interface definition, an implementation in another file, and a configuration change introduced elsewhere in the codebase.

Existing methods primarily improve how models access or select relevant context. Recent long-context reinforcement learning approaches induce structured retrieval-and-reasoning trajectories, directly reward contextual grounding, or optimize heterogeneous long-context capabilities (Wang

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

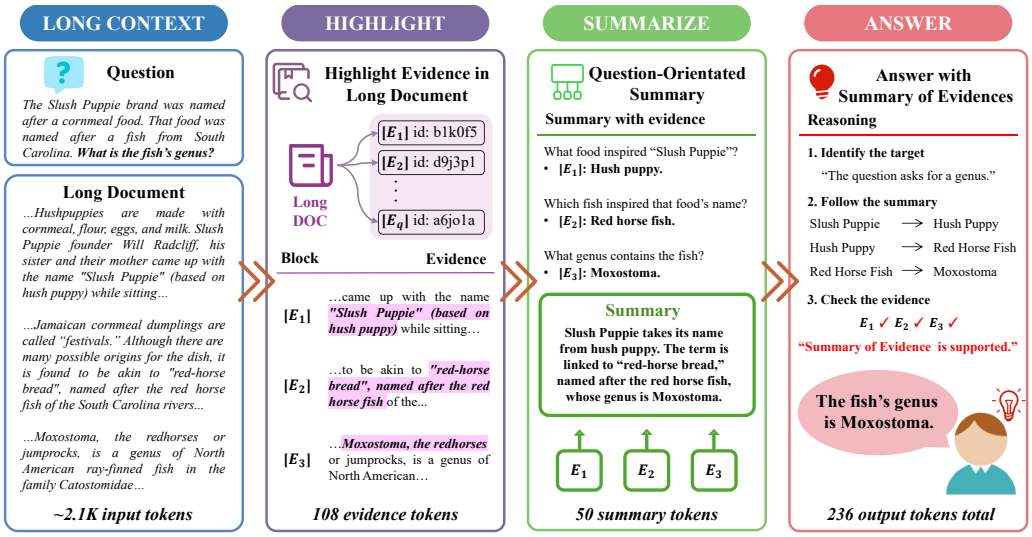


Figure 1: **Highlight-Then-Summarize.** H2S transforms a block-structured long document into source-addressable evidence and then a question-conditioned summary before producing the final answer in one autoregressive trajectory.

et al., 2025; Chen et al., 2026; Lv et al., 2026). These advances make evidence easier to find, but evidence selection alone does not specify how multiple passages should be combined. Their integration often remains implicit in final-answer generation, where the model must simultaneously retain fragmented evidence, infer the relations among it, and complete the task. This entanglement makes evidence difficult to use consistently and can encourage unnecessarily long reasoning.

We propose **Highlight-Then-Summarize (H2S)**, a compress-then-reason paradigm that externalizes both evidence selection and evidence integration before final answering, as illustrated in Figure 1. H2S first highlights source-addressable evidence, then organizes the selected information into a question-conditioned summary, and finally produces the answer conditioned on these intermediate representations. Highlight preserves a traceable path to the document, whereas Summarize converts fragmented passages into a compact, reasoning-ready state. Both are generated by one autoregressive model, and the original document remains available throughout generation; H2S therefore organizes rather than prunes the context. A motivating study supports this design: adding only 1.04K tokens of self-generated question-conditioned summary to a 15.19K-token document improves Qwen3.5-Flash by 13.3 points.

To supervise this behavior, we construct **H2S-Dataset**, comprising 6,647 examples from 11 benchmark families, including 4,228 examples for supervised fine-tuning and 2,419 for reinforcement learning. Each example follows the structured Evidence–Summary–Answer format. We further introduce **H2S-RL**, which uses a deterministic trajectory-level reward to jointly evaluate the final answer, evidence grounding and span coverage, summary quality, and output structure. This design provides direct feedback on both intermediate stages instead of relying only on final-answer correctness. We evaluate the resulting models on **H2S-Bench**, a seven-task suite spanning retrieval, reasoning, aggregation, citation, and summarization over long contexts.

Under a shared 128K input and 4K output budget, H2S-7B and H2S-14B achieve average scores of 28.56 and 32.60 on H2S-Bench. H2S-14B obtains the strongest overall result among the evaluated open-source models, outperforming the larger QwenLong-L1-32B and Qwen3.8-27B by 5.01 and 10.17 points, respectively. H2S-RL further improves the 7B and 14B supervised checkpoints by 4.88 and 3.44 points. Process-level analyses support the intended mechanism: H2S-14B achieves the highest Evidence–Summary Quality score, removing its summary lowers average performance by 8.47 points, and extending its output budget from 4K to 16K yields only 0.96 additional points. These results indicate that explicitly organizing evidence can improve long-context reasoning without relying on increasingly long generation.

Our contributions are threefold:

(1) We propose Highlight-Then-Summarize, a compress-then-reason paradigm that externalizes evidence localization and integration as source-grounded evidence and a question-conditioned summary before final answering.

(2) We construct H2S-Dataset, a diverse long-context training set of 6,647 structured Evidence–Summary–Answer examples from 11 benchmark families.

(3) We introduce H2S-RL, which jointly evaluates final-answer correctness, evidence grounding and span coverage, summary quality, and output structure. H2S-14B achieves the strongest overall result among the evaluated open-source models on H2S-Bench.

2 Related Work

2.1 Long-Context Tasks

Long-context understanding encompasses a broad set of capabilities rather than a single task. Existing benchmarks therefore evaluate complementary aspects of long-context processing. LongBench and LongBench v2 cover a diverse range of tasks, including question answering, summarization, long-form dialogue, code understanding, and reasoning (Bai et al., 2024c;b). FRAMES emphasizes factual retrieval, multi-document information gathering, and reasoning (Krishna et al., 2024), while AA-LCR focuses on information synthesis across long documents (Artificial Analysis, 2025). MRCR evaluates whether models can identify and track requested information in long multi-turn contexts containing distractors (Vodrahalli et al., 2024), and HELMET assesses retrieval, question answering, summarization, citation, and ranking in application-oriented settings (Yen et al., 2024).

Collectively, these benchmarks assess long-context capabilities spanning information retrieval, cross-context understanding, and reasoning over distributed evidence. These capabilities require models to move beyond locating relevant information toward selecting and organizing evidence across the context. However, existing benchmarks focus primarily on final outputs, offering limited assessment of this intermediate evidence-processing stage. Our work targets this gap by measuring how models identify, consolidate, and summarize distributed evidence before reasoning.

2.2 Evidence Processing for Long-Context Reasoning

Evidence Retrieval and Attribution. Retrieval-augmented generation decouples information access from answer generation by retrieving relevant passages as external evidence (Lewis et al., 2020). Lexical retrieval, dense retrieval, and rank fusion constitute standard approaches for evidence localization (Robertson & Zaragoza, 2009; Cormack et al., 2009). Attribution-oriented methods instead emphasize the correspondence between generated claims and supporting sources: ALCE evaluates citation correctness and completeness (Gao et al., 2023), Self-RAG integrates retrieval with self-reflective generation (Asai et al., 2023), and LongCite introduces fine-grained citation supervision for long-context question answering (Zhang et al., 2024). Collectively, these methods concern the localization of relevant evidence and its attribution to source text. H2S introduces explicit supervision for evidence identification, positioning question-relevant evidence extraction as a distinct intermediate objective preceding summarization and reasoning.

Context Engineering for Compression. Question-aware prompt compression selectively preserves relevant information in long inputs (Jiang et al., 2024). Recent work also compresses growing interaction histories into compact states for continued reasoning. ReSum periodically summarizes agent trajectories into concise reasoning states for long-horizon search (Wu et al., 2026). A-MEM constructs and updates structured memory notes with explicit links among stored information (Xu et al., 2025), while Memory-R1 learns memory operations for retaining, updating, and discarding information (Yan et al., 2025). These methods perform context compression as an external mechanism for managing long interaction histories or persistent memories. In contrast, H2S internalizes compression into the reasoning process by supervising the construction of a question-conditioned evidence summary as an intermediate representation for downstream reasoning.

2.3 LONG-CONTEXT REINFORCEMENT LEARNING

Recent work has explored reinforcement learning for improving information acquisition and utilization in long-context settings. LoongRL constructs distractor-rich, multi-step tasks to train models for planning, retrieval, reasoning, and verification (Wang et al., 2025). LongRLVR introduces verifiable context rewards that supervise the context selected prior to answer generation (Chen et al., 2026). GoLongRL extends long-context reinforcement learning to multiple capability-oriented tasks and employs reward balancing for multitask alignment (Lv et al., 2026). Existing approaches mainly optimize evidence retrieval or context selection, while leaving the subsequent organization of multiple evidence items unsupervised. Our method instead applies process supervision to both evidence selection and question-conditioned summary construction, explicitly optimizing the transition from retrieved evidence to a compact intermediate representation before reasoning.

3 METHODOLOGY

3.1 WHY A FEW SUMMARY TOKENS HELP

Cognitive theories of text comprehension suggest that effective understanding relies on organizing local information into a goal-conditioned macrostructure Kintsch & van Dijk (1978). This motivates a natural question: **Can a compact, question-conditioned representation improve reasoning over long documents?** To investigate this question, we evaluate question-conditioned summaries on DocQA-RL-1.6K (Wan et al., 2025), using Qwen3.5-Flash as the answer model. We compare reasoning over the original document with reasoning over summaries generated solely from the question and document, without access to the reference answer. *Claude Summary* is generated by Claude Sonnet 4.6, whereas *Self-Summary* is produced by Qwen3.5-Flash in a separate pass.

Table 1: **Even a few summary tokens can substantially improve long-document QA.**

Input to Qwen3.5-Flash	Context tokens	Acc.
LongDoc	15.19K	72.6
Claude Summary	0.46K	93.0
LongDoc + Claude Summary	15.19K + 0.46K	**96.7**
LongDoc + Self-Summary (S)	15.19K + 0.14K	69.3
LongDoc + Self-Summary (M)	15.19K + 0.60K	77.8
LongDoc + Self-Summary (L)	15.19K + 1.04K	85.9

As shown in Table 1, a compact question-conditioned summary can substantially improve downstream reasoning despite containing only a small fraction of the original context. A 0.46K-token Claude summary, approximately 3% of the 15.19K-token document, increases accuracy from 72.6 to 93.0 when used alone, while combining it with the original document further improves performance to 96.7. Similar gains are observed with self-generated summaries: a 1.04K-token self-summary improves accuracy from 72.6 to 85.9. These results suggest that explicitly restructuring distributed evidence into a compact, question-conditioned representation can make long-context information more effective for subsequent reasoning, providing direct motivation for the evidence-to-summary stage in H2S. Full results across answer models and document-length ranges are reported in Appendix A.8.

3.2 HIGHLIGHT-THEN-SUMMARIZE PARADIGM

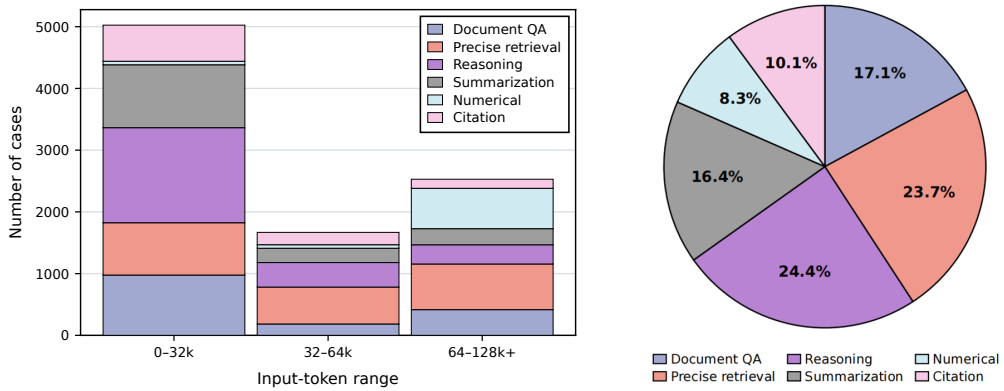
Highlight-Then-Summarize (H2S) is a compress-then-reason paradigm that externalizes evidence selection and evidence integration as explicit intermediate representations before final-answer generation. Given a question q and a long document divided into source-addressable blocks $D = (B_1, \ldots, B_M)$, H2S follows

$$(D, q) \xrightarrow{\text{Highlight}} E_q \xrightarrow{\text{Summarize}} S_q \xrightarrow{\text{Reason}} a. \tag{1}$$

The Highlight stage produces question-relevant evidence

$$E_q = \{(u_i, s_i)\}_{i=1}^{m}, \qquad s_i \subseteq B(u_i), \tag{2}$$

where u_i is a stable source-block identifier, $B(u_i)$ is the corresponding block, and s_i is its supporting span. The Summarize stage then maps these traceable but potentially scattered spans into a question-conditioned summary S_q that retains their answer-relevant content and makes their relations explicit. Thus, S_q is neither a generic document summary nor a concatenation of retrieved passages, but a compact representation organized for the reasoning required by q.



(a) Length Analysis of H2S Dataset **(b) Task Category Analysis**

Figure 2: **Composition of H2S-Dataset.** H2S-Dataset spans diverse long-context tasks and input lengths, with reasoning and precise retrieval accounting for 24.4% and 23.7% of the dataset, respectively, while examples cover input lengths from 0–32K to 64–128K+ tokens.

H2S realizes the three operations within a single autoregressive trajectory. Its conditional distribution factorizes as

$$p_\theta(E_q, S_q, a \mid D, q) = p_\theta(E_q \mid D, q)\, p_\theta(S_q \mid D, q, E_q)\, p_\theta(a \mid D, q, E_q, S_q). \tag{3}$$

The three factors respectively model source-grounded localization, question-conditioned evidence integration, and task completion. Unlike direct supervision of $p_\theta(a \mid D, q)$, this factorization makes the first two operations observable and therefore available for supervision and process-level evaluation.

As shown in Figure 1, Highlight and Summarize form complementary interfaces: E_q preserves provenance but may remain fragmented, whereas S_q integrates the selected information into a compact reasoning state. Both are generated by the same model, and the original document remains available throughout generation. H2S therefore organizes the context rather than pruning it, preserving access to the source while making the information needed for reasoning explicit.

3.3 Construction of H2S-Dataset

Most long-context datasets supervise only the final answer, leaving E_q and S_q in Equation 3 unobserved. Following prior work on long-context alignment data (Bai et al., 2024a), we construct H2S-Dataset by augmenting existing instances (D, q, a) with source-grounded evidence and a question-conditioned summary, yielding (D, q, E_q, S_q, a). H2S-Dataset comprises 6,647 examples from 11 benchmark families, with 4,228 examples for SFT and 2,419 for RL. The construction pipeline is shown in Figure 3. Source and length distributions are reported in Appendix A.2.

Blocking for Long Document. We first segment each document at semantic boundaries and assign every block a stable identifier while preserving its source position. Hybrid lexical and semantic retrieval forms a high-recall candidate pool. We then decompose the question into atomic information needs and extract question-relevant claims, binding each claim to a `BLOCK_ID` and its supporting span. This produces traceable evidence supervision E_q. Retrieval is used only for offline data construction; H2S reads the full supplied document without an external retriever at inference time.

Summary of Evidence Generation. We remove unsupported and duplicate claims, organize the retained claims by information need, and synthesize the question-conditioned summary S_q. The information needs act as a coverage scaffold, preventing a fluent summary from omitting part of the question. Consequently, S_q integrates source-linked claims rather than concatenating retrieved passages. A complete construction example is provided in Appendix A.5.

Quality control. We retain an instance only if it passes three checks: *faithfulness*, which verifies block identifiers and source-span alignment; *coverage*, which requires every information need to be supported; and *answerability*, which tests whether the answer can be derived from the question and constructed summary. Failed instances return for reconstruction. The accepted record preserves the

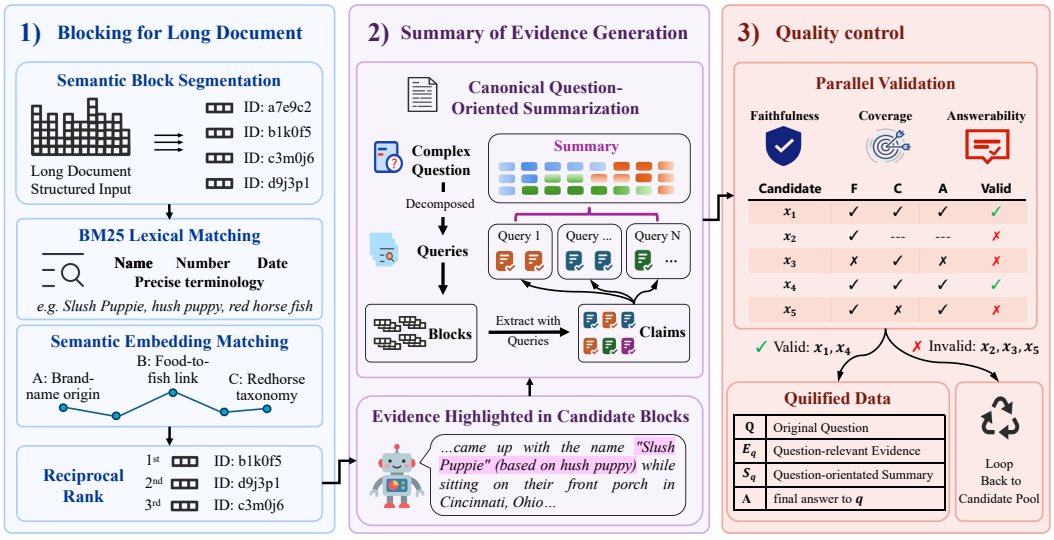


Figure 3: **Dataset Construction of Highlight-Then-Summarize.** Each training instance is constructed through semantic blocking, candidate retrieval, query-guided evidence extraction, question-conditioned summarization, and quality verification, with failed instances iteratively reconstructed until the required criteria are satisfied.

original task answer and adds E_q and S_q as supervision for evidence localization and information integration.

3.4 H2S-RL

H2S-RL employs GRPO (Shao et al., 2024) to optimize each generated trajectory $y = (E, S, a)$ with a deterministic programmatic reward that jointly evaluates the final answer and the evidence-to-summary process, without an online LLM judge. It uses three quality signals and one structure gate. **(1) Answer quality** R_{answer} follows the native output contract of each task. **(2) Evidence quality** R_{evidence} comprises two complementary components: R_{ground} is the fraction of evidence items whose block identifiers and span locators can be resolved in the source document, while R_{span} is the span-level F1 between the resolved and reference evidence intervals. **(3) Summary quality** R_{summary} is ROUGE-L (Lin, 2004) between the generated and reference question-conditioned summaries after removing local evidence identifiers. **(4) Structure quality** $R_{\text{structure}}$ gates trajectories with missing, partially valid, or complete Evidence–Summary–Answer structure and source fields.

$$\mathcal{R}_{\text{H2S}}(y) = R_{\text{structure}}(y)\left[\lambda R_{\text{answer}}(a) + (1-\lambda)\frac{2R_{\text{evidence}}(E)R_{\text{summary}}(S)}{R_{\text{evidence}}(E) + R_{\text{summary}}(S)}\right]. \tag{4}$$

This design treats evidence selection and summary construction as complementary stages: the process reward remains low when either stage fails, rather than allowing one strong component to conceal the other. The structure gate further prevents malformed trajectories from exploiting otherwise high component scores. Consequently, the within-prompt advantages used by GRPO favor outputs that improve task success through a complete and source-grounded intermediate process, without requiring an online LLM judge. Metric routing and optimization details are reported in Appendix A.1.

4 Experiments

4.1 Experimental Setup

We train H2S-7B and H2S-14B initialized from Qwen2.5-7B/14B-Instruct-1M, using full-parameter SFT followed by GRPO. Training adopts a progressive context-length curriculum of

32K→64K→128K. We evaluate the resulting models on H2S-Bench, a suite of 2,575 examples spanning seven long-context capabilities: numerical reasoning (DocFinQA), multi-document reasoning (Frames), citation-grounded answering (LongCite), precise retrieval (MRCR), distributed-information aggregation (AA-LCR), general long-context understanding (LongBenchV2), and long-document summarization (HELMET-Summ). Performance is assessed exclusively on the final answer using the native evaluation metric of each benchmark. All models are evaluated under an identical input configuration, receiving the same system prompt, question, and block-structured document, with a maximum input context of 128K tokens and an output budget of 4K tokens. Further details on the training and evaluation protocol are provided in Appendix A.1 and Appendix A.7.

4.2 Main Results

Table 2: **Main results on seven long-context benchmarks.** All models use a maximum input length of 128K and a maximum output length of 4K. Closed-source models are included for reference; among all open-source models, including our models, the best result in each column is shown in **bold**, and the second-best result is shown in **bold and underlined**.

Category	Model	DocFinQA	Frames	LongCite	MRCR	AA-LCR	LongBenchV2	HELMET-Summ	Average
Closed-source	Claude Opus 5	56.11	27.89	12.17	100.00	26.45	64.18	15.13	43.13
	GPT-5.5	49.57	36.05	12.62	92.38	52.34	46.15	9.45	42.65
	DeepSeek-V4-Pro	34.83	37.41	6.78	72.14	47.77	48.56	12.48	37.14
	Kimi-K2.5	51.14	37.41	10.83	87.58	39.95	50.00	14.41	41.62
	GLM-5.2	49.57	36.05	12.62	92.38	40.85	48.80	16.52	42.40
Open-source	DeepSeek-R1-Distill-Qwen-7B	0.06	8.50	1.31	0.40	0.06	1.92	5.29	2.51
	DeepSeek-R1-Distill-Qwen-14B	11.69	23.13	5.25	38.28	8.75	23.32	11.81	17.46
	Qwen3-4B-Thinking-2507	14.61	18.71	3.68	25.45	9.88	18.99	10.80	14.59
	Qwen3.8-27B	**40.52**	32.31	4.94	46.89	15.69	15.14	1.51	22.43
	Qwen3.5-35B-A3B	24.72	18.03	2.13	8.62	6.11	6.97	0.06	9.52
	QwenLong-L1-32B	**35.03**	34.35	8.16	40.28	**26.20**	**37.98**	11.13	27.59
	QwenLong-L1.5-30B-A3B	9.34	25.17	1.03	0.00	5.37	30.52	3.92	10.76
	Qwen2.5-7B-Instruct-1M	5.00	13.95	5.37	4.01	7.31	24.04	7.16	9.55
	Qwen2.5-14B-Instruct-1M	15.70	23.47	8.32	25.05	13.56	**35.82**	**12.03**	19.14
Ours	H2S-7B-SFT	20.45	31.29	8.65	58.12	13.96	27.88	5.42	23.68
	H2S-7B	26.33	32.97	**15.76**	59.72	21.47	32.11	11.58	28.56
	H2S-14B-SFT	31.49	**37.76**	10.50	**65.13**	22.40	28.37	8.45	**29.16**
	H2S-14B	30.83	**37.72**	**13.88**	**66.13**	**26.86**	35.42	**17.38**	**32.60**

H2S demonstrates strong overall performance across long-context benchmarks. As shown in Table 2, H2S-14B achieves the highest average score among the evaluated open-source models, reaching 32.60 and surpassing the larger QwenLong-L1-32B and Qwen3.8-27B by 5.01 and 10.17 points, respectively. H2S-7B attains an average score of 28.56, exceeding QwenLong-L1-32B by 0.97 points despite using fewer than one quarter of its parameters. Moreover, H2S-14B ranks among the top two open-source models on five of the seven benchmarks, spanning cited answering, precise retrieval, distributed-information aggregation, and long-document summarization. These results suggest that the gains of H2S generalize across heterogeneous long-context reasoning settings rather than being driven by performance on a particular benchmark or task type.

H2S-RL delivers consistent gains over SFT. Starting from the SFT checkpoints, H2S-RL raises the average by 4.88 points for 7B and 3.44 points for 14B. These gains are consistent across model scales and are concentrated on tasks that require selecting and integrating information distributed through long inputs: for example, H2S-RL improves LongCite, AA-LCR, and HELMET-Summ by 7.11, 7.51, and 6.16 points for 7B, and by 3.38, 4.46, and 8.93 points for 14B. This consistent improvement over strong SFT checkpoints shows that trajectory-level optimization adds value beyond learning the output structure by imitation. The following process-level analyses examine whether these final-answer gains are accompanied by better evidence and summaries.

4.3 Analysis of Summarization Quality

To determine whether better final answers arise from a better intermediate process, we introduce **Evidence–Summary Quality (ESQ)**. ESQ is high only when a model both recovers the reference evidence blocks and produces a reference-aligned question-conditioned summary; it is computed as their case-level harmonic mean followed by macro-averaging.

The complete definition and per-model component scores are provided in Appendix A.3. The upper-right region of Figure 4 corresponds to strong final answers together with high-quality intermediate representations. H2S-14B reaches the highest ESQ among all evaluated models at 31.02, followed by GLM-5.2 at 30.50; H2S-7B reaches 26.60 and substantially outperforms the open-source baselines. Both H2S-SFT and H2S improve markedly over their corresponding base models, while H2S-RL further raises ESQ from 25.22 to 26.60 for 7B and from 29.93 to 31.02 for 14B. The gains in final-answer score are therefore accompanied by better evidence selection and summary construction rather than answer fitting alone.

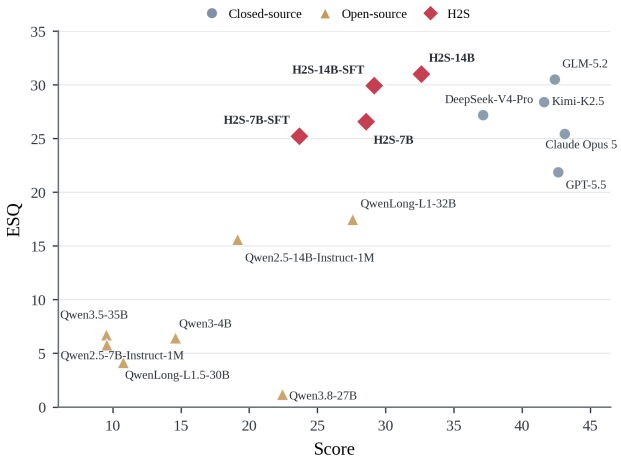


Figure 4: **Analysis of Summarization Quality.** Final-answer score versus Evidence–Summary Quality (ESQ). Higher is better on both axes.

4.4 Ablation Study

Table 3: **Detailed ablation of evidence and summary generation.** *Baseline* removes both evidence and summary generation; *w/o evidence* and *w/o summary* remove the corresponding intermediate component, respectively. Avg is the unweighted macro-average over the seven tasks.

Model	DocFinQA	Frames	LongCite	MRCR	AA-LCR	LongBenchV2	HELMET-Summ	Avg
Baseline-7B	4.82	21.43	3.92	34.80	10.71	29.09	2.65	15.35
w/o evidence-7B	23.56	30.61	9.97	54.60	14.60	31.01	7.52	24.55
w/o summary-7B	13.64	29.93	4.23	42.40	10.98	21.63	7.81	18.66
H2S-7B	26.33	32.97	15.76	59.72	21.47	32.11	11.58	**28.56**
Baseline-14B	25.52	29.59	4.06	39.00	13.19	31.73	12.89	22.28
w/o evidence-14B	25.84	37.41	10.64	59.80	17.34	35.82	11.60	28.35
w/o summary-14B	24.36	33.67	4.94	47.40	17.86	28.61	12.09	24.13
H2S-14B	30.83	37.72	13.88	66.13	26.86	35.42	17.38	**32.60**

Evidence and Summary Ablation. We ablate the two intermediate components at inference time while retaining the final-answer stage. During autoregressive decoding, we mask the ablated component from subsequent attention while keeping the input, checkpoint, and final-answer stage unchanged. Table 3 reports the average performance under the same 128K input and 4K output limits as the main evaluation. Both components are necessary. Removing evidence lowers the average by 4.01 points for H2S-7B and 4.25 for H2S-14B, showing that explicit localization provides information that cannot be recovered reliably from the raw context during answering. Removing the summary is more damaging, with drops of 9.90 and 8.47 points; removing both causes the largest drops, 13.21 and 10.32 points. These results support our central claim that localization alone is insufficient: scattered evidence must be compressed into a question-conditioned representation before reasoning.

Output Budget Ablation. Table 4 compares H2S with open-source baselines under maximum output budgets of 4K, 8K, and 16K tokens. The input length is fixed at 128K for all settings, and task-level results are reported in Appendix A.4. Increasing the output budget substantially improves several thinking baselines. For example, Qwen3.8-27B increases from 22.43 at 4K to 43.72 at 16K. In contrast, H2S is substantially less sensitive to the output budget: H2S-7B improves by only 1.92 points from 4K to 16K, while H2S-14B improves by 0.96 points, with most of the gain already realized at 8K. Notably, H2S-14B reaches 32.60 under the 4K budget, outperforming Qwen3.8-27B

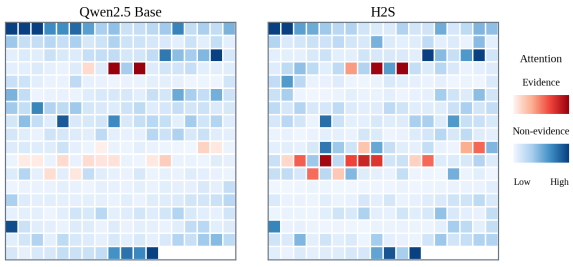


Figure 5: **Attention Distribution of Long Document Analysis.** H2S improves long-context localization by inducing more concentrated attention over query-relevant evidence in the source document.

Model	Score		
	4K	8K	16K
Qwen3-4B-Thinking-2507	14.59	22.84	26.21
Qwen3.8-27B	22.43	37.56	43.72
Qwen3.5-35B-A3B	9.52	30.44	35.15
QwenLong-L1.5-30B-A3B	10.76	18.62	25.56
H2S-7B	28.56	30.39	30.48
H2S-14B	32.60	33.44	33.56

Table 4: **Output budget ablation.** Average score under different output budgets.

by 10.17 points at the same output length. These results indicate that H2S achieves strong long-context performance with a relatively modest generation budget.

4.5 Attention Distribution of Long Document Analysis

To inspect how H2S changes document reading, we compare block-level attention from Qwen2.5-7B-Instruct-1M and H2S-7B at the onset of evidence generation. Figure 5 shows all 318 source blocks from the same tokenized LongCite input in document order. H2S more than doubles the share of attention assigned to the 16 annotated evidence blocks while leaving attention over most non-evidence blocks comparatively diffuse. This source-traceable case study illustrates the evidence-focused reading behavior that complements the aggregate ablation results; it is a mechanism example rather than a performance estimate.

5 Conclusion

We present Highlight-Then-Summarize (H2S), a compress-then-reason paradigm that explicitly structures long-context reasoning through source-grounded evidence localization and question-conditioned summarization. Supported by H2S-Dataset and H2S-RL, H2S consistently improves final-answer performance and intermediate reasoning quality across seven long-context benchmarks while retaining strong performance under limited output budgets. A current limitation is that H2S is less naturally suited to tasks requiring broadly distributed context coverage, while process-level evaluation remains dependent on construction-time evidence and summary references. Future work will explore more adaptive compression objectives and stronger process supervision, with the broader goal of making long-context reasoning more structured, efficient, and reliable.

Reproducibility Statement

We support reproducibility through detailed descriptions of the method, data, training, and evaluation protocols, together with an anonymous code release. Section 3 specifies the H2S factorization, data-construction pipeline, and H2S-RL reward. Section 4 defines the shared evaluation setting and reports the main comparisons and ablations. Appendix A.1 provides model initialization, optimization hyperparameters, and evaluation details; Appendices A.2 and A.7 report the training-data statistics and the complete context-length curriculum. Appendix A.3 defines ESQ and reports its component scores, while Appendix A.4 provides task-level results under different output budgets. We additionally provide a complete data-construction example in Appendix A.5 and qualitative success and failure cases in Appendix A.9. The anonymous repository includes data-processing utilities, H2S-SFT and H2S-RL configurations, the programmatic reward, task-specific evaluation code, data schemas, and executable examples. Code and H2S-Dataset will be released publicly upon publication.

References

Artificial Analysis. Artificial analysis long context reasoning benchmark. Benchmark, 2025.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*, 2023.

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. Longalign: A recipe for long context alignment of large language models. *arXiv preprint arXiv:2401.18058*, 2024a.

Yushi Bai et al. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. *arXiv preprint arXiv:2412.15204*, 2024b.

Yushi Bai et al. Longbench: A bilingual, multitask benchmark for long context understanding. *Proceedings of ACL*, 2024c.

Guanzheng Chen, Michael Qizhe Shieh, and Lidong Bing. Longrlvr: Long-context reinforcement learning requires verifiable context rewards. *arXiv preprint arXiv:2603.02146*, 2026.

Gordon V. Cormack, Charles L. A. Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference*, pp. 758–759, 2009.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. *arXiv preprint arXiv:2305.14627*, 2023.

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

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

Huiqiang Jiang, Qianhui Wu, Chin-Yew Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. Longllmlingua: Accelerating and enhancing llms in long context scenarios via prompt compression. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations*, volume 2024, pp. 54107–54157, 2024.

Walter Kintsch and Teun A. van Dijk. Toward a model of text comprehension and production. *Psychological Review*, 85(5):363–394, 1978.

Satyapriya Krishna et al. Fact, fetch, and reason: A unified evaluation of retrieval-augmented generation. *arXiv preprint arXiv:2409.12941*, 2024.

Patrick Lewis et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33, 2020.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*, pp. 74–81, 2004.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173, 2024.

Minxuan Lv, Tiehua Mei, Tanlong Du, Junmin Chen, Zhenpeng Su, Ziyang Chen, Ziqi Wang, Zhennan Wu, Ruotong Pan, Jian Liang, Ruiming Tang, and Han Li. Golongrl: Capability-oriented long context reinforcement learning with multitask alignment. *arXiv preprint arXiv:2605.19577*, 2026.

OpenAI. Openai o3 and o4-mini system card. `https://openai.com/index/o3-o4-mini-system-card/`, April 2025. Accessed 2026-09-25.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009.

Zhihong Shao et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026.

Kiran Vodrahalli et al. Michelangelo: Long context evaluations beyond haystacks via latent structure queries. *arXiv preprint arXiv:2409.12640*, 2024.

Fanqi Wan, Weizhou Shen, Shengyi Liao, Yingcheng Shi, Chenliang Li, Ziyi Yang, Ji Zhang, Fei Huang, Jingren Zhou, and Ming Yan. Qwenlong-l1: Towards long-context large reasoning models with reinforcement learning. *arXiv preprint arXiv:2505.17667*, 2025.

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

Xixi Wu, Kuan Li, Yida Zhao, Liwen Zhang, Litu Ou, Huifeng Yin, Zhongwang Zhang, Xinmiao Yu, Ding-Chu Zhang, Yong Jiang, Pengjun Xie, Fei Huang, Minhao Cheng, Shuai Wang, Hong Cheng, and Jingren Zhou. Resum: Unlocking long-horizon search intelligence via context summarization, 2026. URL https://openreview.net/forum?id=PjIK38mwKm.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-mem: Agentic memory for llm agents. *arXiv preprint arXiv:2502.12110*, 2025.

Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Hinrich Schuetze, Volker Tresp, and Yunpu Ma. Memory-r1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. *arXiv preprint arXiv:2508.19828*, 2025.

Howard Yen et al. Helmet: How to evaluate long-context language models effectively and thoroughly. *arXiv preprint arXiv:2410.02694*, 2024.

Jiajie Zhang, Yushi Bai, Xin Lv, et al. Longcite: Enabling llms to generate fine-grained citations in long-context qa. *arXiv preprint arXiv:2409.02897*, 2024.

A Supplementary Material

This appendix provides additional experimental and implementation details, including training configurations, data statistics, process-level evaluation, component ablations, output-budget results, data-construction examples, and qualitative case studies.

AI use statement

Generative AI tools were used solely to assist with language editing and polishing the presentation of this paper. All AI-assisted text was reviewed and revised by the authors, who take full responsibility for the final content.

A.1 Detailed Experimental Setup

Training hyperparameters. Both H2S models are trained with full-parameter updates from Qwen2.5-7B-Instruct-1M and Qwen2.5-14B-Instruct-1M. SFT runs for one epoch with a learning rate of 1×10^{-5}, a maximum input length of 128K, and a maximum target length of 4K. RL starts from the corresponding SFT checkpoint and uses full-parameter GRPO; each curriculum stage runs for one epoch with a learning rate of 1×10^{-6}, samples seven candidates per question, and allows at most 2K generated tokens. All stages use bf16, a cosine learning-rate schedule, 5% warmup, gradient checkpointing, and ZeRO-3.

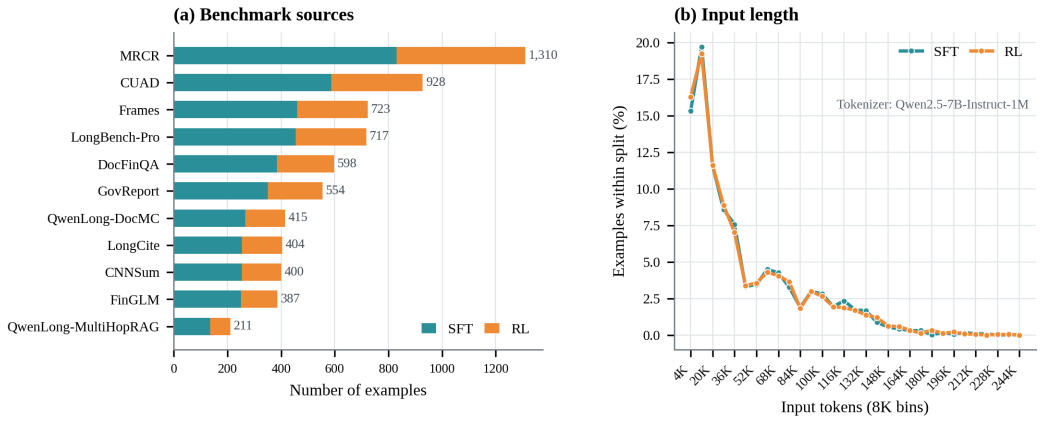


Figure A6: **Benchmark and input-length distributions of the final training data.** The left panel merges different task configurations from the same benchmark; LongBench-Pro contains T1–T11. The right panel shows the within-split length proportions for SFT and RL. The two splits have similar task-source and length distributions.

Reward implementation. The final-answer term follows each task's native output contract: numerical tolerance for numerical answers, choice accuracy for multiple-choice tasks, normalized exact match or token F1 for free-form QA, set F1 or NDCG for structured outputs, ROUGE-L for summarization, and joint content–citation quality for cited answers. The evidence term used during RL is span-level F1 over resolved source intervals; it is distinct from the block-level evidence recall used for the ESQ analysis in Section 4.3. Citation coverage and information-need coverage are logged for diagnosis but are not additional terms in Equation 4.

Comparison models. The main table compares five closed-source models, nine open-source models (including two Qwen2.5-Instruct base models), and our H2S-SFT and H2S variants at both model sizes. Every model is evaluated with a maximum input length of 128K and a maximum output length of 4K, and outputs that cannot be parsed receive a score of zero. We use each benchmark's native metric without additional model-specific prompt tuning.

A.2 Training Data Overview

The final training set contains 4,228 SFT examples and 2,419 RL examples, for a total of 6,647 examples from 11 benchmark families, spanning precise retrieval, document question answering, reasoning, summarization, numerical calculation, and citation. Figure A6 summarizes the benchmark sources and input-length distributions. The largest sources are MRCR, CUAD, Frames, and LongBench-Pro, while no single task accounts for most of the training set.

The SFT and RL splits have similar length profiles: approximately 55% of examples are no longer than 32K tokens, while about 26% exceed 64K tokens. The mean and median lengths are 44,097 and 28,290 tokens for SFT, and 43,656 and 27,726 tokens for RL, respectively. The 90th-percentile lengths are 108,497 tokens for SFT and 108,151 for RL. Examples longer than 128K are retained in the complete data resource but are not used in the current 128K training stages.

A.3 Evidence–Summary Quality Subset and Detailed Results

For case i, let E_i denote recall over reference evidence blocks and S_i denote summary ROUGE-L, both normalized to $[0, 1]$. We define

$$\mathrm{ESQ} = \frac{100}{N} \sum_{i=1}^{N} \frac{2E_i S_i}{E_i + S_i}, \tag{5}$$

Table 5: **Final-answer and evidence–summary quality.** Final Answer is the seven-task Avg from Table 2. E-Recall is the macro-average of per-example recall over reference evidence blocks; predictions are the 8-character block IDs parsed from `<evidence>`. Construction-time IDs are aligned to equivalent IDs in the actual inference blockization by normalized character matching before ID-only scoring. S-RL is ROUGE-L between the parsed `<summary>` and the reference summary. ESQ is the macro-average of the per-example harmonic mean of E-Recall and S-RL. Missing or unparsable components receive zero.

Category	Model	Final Answer ↑	E-Recall ↑	S-RL ↑	ESQ ↑
Closed-source	Claude Opus 5	43.13	56.31	23.93	25.45
	GPT-5.5	42.65	41.10	28.61	21.87
	DeepSeek-V4-Pro	37.14	45.47	33.20	27.18
	Kimi-K2.5	41.62	48.96	32.51	28.41
	GLM-5.2	42.40	51.75	34.88	30.50
Open-source	Qwen3-4B-Thinking-2507	14.59	14.94	11.70	6.44
	Qwen3.8-27B	22.43	2.71	3.01	1.18
	Qwen3.5-35B-A3B	9.52	15.32	7.61	6.73
	QwenLong-L1-32B	27.59	34.31	25.93	17.47
	QwenLong-L1.5-30B-A3B	10.76	7.74	5.30	4.17
	Qwen2.5-7B-Instruct-1M	9.55	14.49	18.16	5.80
	Qwen2.5-14B-Instruct-1M	19.14	28.32	25.84	15.61
Ours	H2S-7B-SFT	23.68	36.12	30.73	25.22
	H2S-7B	28.56	38.85	32.48	26.60
	H2S-14B-SFT	29.16	43.06	36.78	29.93
	H2S-14B	32.60	44.94	38.19	31.02

with the case-level value set to zero when both components are zero. The harmonic mean requires the model to perform well on both evidence selection and summary construction; a high score on either component cannot compensate for failure on the other.

E-Recall is an ID-level measure rather than span or citation matching: each reference evidence block is counted once, and a predicted valid 8-character block UUID receives credit when it matches the construction-time ID or an equivalent ID located by normalized character matching in the actual inference input. This alignment is necessary because the stored construction annotations and inference inputs can use different block segmentations. The column is therefore reported as E-Recall, not Evidence F1. Summary quality is evaluated independently with ROUGE-L, and the ESQ column is the case-level harmonic mean of the two components.

A.4 Detailed Output-Budget Results

Table 6 provides the task-level results corresponding to Table 4. Complete is the percentage of generations that either terminate before reaching the output limit or contain a complete final-answer field.

A.5 LongBench-Pro Construction Example

We use the RL example `v3_lbp_T10_1249_167a08600320` from LongBench-Pro to illustrate the complete transformation from a long document to a training target. The example contains 17,688 input tokens. Given a product manual, the question asks whether using oven cleaner on the non-stick coating of the AF-500 should be classified as *Warning*, *Caution*, or *Note*.

Step 1. Block segmentation and retrieval. The document is segmented at semantic boundaries. BM25 and embedding retrieval are used in parallel, and the candidates are ranked with RRF. The two blocks directly relevant to the answer are:

Table 6: **Detailed performance under different output budgets.** Avg denotes the unweighted macro-average across the seven tasks, and Complete denotes the percentage of completed generations.

Model	Budget	DocFinQA	Frames	LongCite	MRCR	AA-LCR	LongBenchV2	HELMET-Summ	Avg	Complete
Qwen3-4B-Thinking-2507	4K	14.61	18.71	3.68	25.45	9.88	18.99	10.80	14.59	53.75%
	8K	24.30	25.51	4.82	47.49	12.31	31.73	13.73	22.84	86.60%
	16K	26.82	29.93	4.92	56.11	15.16	36.34	14.18	26.21	95.15%
Qwen3.8-27B	4K	40.52	32.31	4.94	46.89	15.69	15.14	1.51	22.43	41.17%
	8K	50.12	39.46	7.22	95.39	28.26	38.46	4.00	37.56	68.78%
	16K	51.38	42.18	10.25	97.60	42.14	54.33	8.16	43.72	86.60%
Qwen3.5-35B-A3B	4K	24.72	18.03	2.13	8.62	6.11	6.97	0.06	9.52	14.83%
	8K	39.40	29.25	8.34	75.95	23.23	31.49	5.39	30.44	56.23%
	16K	40.02	32.31	9.39	82.97	30.08	39.90	11.37	35.15	70.76%
QwenLong-L1.5-30B-A3B	4K	9.34	25.17	1.03	0.00	5.37	30.52	3.92	10.76	15.46%
	8K	23.83	36.73	2.00	1.40	27.03	33.65	5.67	18.62	53.86%
	16K	30.72	40.82	5.23	6.61	36.45	48.32	10.78	25.56	86.91%
H2S-7B	4K	26.33	32.97	15.76	59.72	21.47	32.11	11.58	28.56	78.14%
	8K	26.33	33.33	17.16	61.93	22.21	35.77	16.02	30.39	92.00%
	16K	26.33	33.33	17.17	62.16	22.21	35.77	16.42	30.48	93.13%
H2S-14B	4K	30.83	37.72	13.88	66.13	26.86	35.42	17.38	32.60	87.11%
	8K	30.83	37.72	14.54	67.24	27.41	35.97	20.40	33.44	93.05%
	16K	30.83	37.72	14.54	67.44	27.41	35.97	20.98	33.56	93.98%

`m6wyi3kt`: "Do not use harsh chemicals (e.g., oven cleaner) on the non-stick surface—they will degrade the coating over time."
`g7wu5mov`: "Caution: Indicates a situation that may cause minor personal injury or slight appliance damage if not followed."

Step 2. Subquery decomposition and atomic-claim extraction. The question is decomposed into two information needs: the safety classification of using oven cleaner and the meaning of *Caution* for the AF-500. The pipeline extracts two atomic facts: the operation is classified as *Caution*, and *Caution* indicates possible minor personal injury or slight appliance damage. A separate subquery about the CM-800 does not yield a supported claim and is excluded from the summary.

Step 3. Summary and answer construction. The supported claims are deduplicated and organized by information need to produce the following training target:

> **Summary:** The AF-500 manual classifies using an oven cleaner on the non-stick coating as *Caution*. In this manual, *Caution* indicates possible minor personal injury or slight appliance damage if the instruction is not followed.
> **Answer:** `[Answer] Caution`

Step 4. Quality control. The pipeline checks block IDs, source-span alignment, claim–evidence alignment, question coverage, and whether the question can be answered using the summary alone. The evidence in this example matches the source text exactly, so the example is recorded with `stage6_status=PASS` and `answerability=true`.

This example shows that the pipeline does not construct a summary by directly concatenating retrieved blocks. It first removes unsupported or irrelevant information and then organizes traceable atomic facts into a question-conditioned working representation.

A.6 LongBench-Pro Model Behavior

This section presents one successful case and one failure case to illustrate how structured training changes model behavior and where explicit compression can still fail. For readability, we retain only the evidence, summary, and final choice that determine the outcome.

Success case: inferring a missing option from stated information. The LongBench-Pro-T3 example `v3_lbp_T3_0058_ef8456ccf657` contains 8,720 input tokens. The question states that Yan'an, Suide, and Qingyang were among the regions where a unified progressive tax was piloted,

and asks which of Guanzhong, Suide, Qingyang, and Yan'an was not mentioned. The key source text is:

> `jidnkfv1`: "In August 1943, a unified progressive tax was piloted in the three counties of Yan'an, Suide, and Qingyang."

Qwen2.5-14B-Instruct-1M selects D, while H2S-14B-SFT and H2S-14B both select the correct answer A. The trained models' intermediate outputs can be summarized as:

> **Evidence:** The source lists only Yan'an, Suide, and Qingyang.
> **Summary:** Among the four options, the region not listed in the source is Guanzhong.
> **Answer:** `[Answer] A`

The question is easy to localize but requires converting the three listed regions into the missing option. The explicit summary separates this relational judgment from final answer generation, allowing the model to answer directly from the organized representation.

Failure case: a claim extends beyond its evidence span. The LongBench-Pro-T11 example `v3_lbp_T11_0602_d12ef35fdad8` contains 31,340 input tokens and asks which statement about *The Vegetarian* is strictly supported by the document; the correct answer is C. Qwen2.5-14B-Instruct-1M selects C, whereas H2S-14B-SFT and H2S-14B select B. The block identified by the model, `we85ooxe`, states that *The Vegetarian* is a Booker International Prize-winning work by Han Kang, that vegetarianism is an extreme resistance to a patriarchal social system, and that Yeong-hye wishes to become plant-like without harming others. H2S-14B nevertheless produces the following claim:

> "The novel was described by the International Booker Prize judges as unsettling, subtle, and beautiful, while exploring the body, desire, and social discipline."

The source does not contain this evaluation or attribute it to the judges. The unsupported claim is carried into the summary and causes the model to select B. The failure is therefore not a retrieval failure: the model found a relevant block, but the claim exceeded the cited span. This case shows that block-level provenance makes evidence traceable but does not guarantee claim faithfulness; future quality control should strengthen claim–span entailment checks, especially for entities, numbers, negation, and attribution.

A.7 RL Length Curriculum

We use a curriculum that progressively increases the context length:

$$\text{SFT} \longrightarrow \text{RL-32K} \longrightarrow \text{RL-64K} \longrightarrow \text{RL-128K}. \tag{6}$$

The 32K stage is initialized from the SFT checkpoint, and each subsequent stage continues from the preceding checkpoint. The maximum sequence lengths for the 32K and 64K stages are 34,816 and 67,584, respectively. In the 128K stage, the 7B model uses 133,120 tokens and the 14B model uses 131,072 tokens. Each stage includes the prompt and at most 2,048 completion tokens.

Each stage runs for one epoch with a learning rate of 1×10^{-6} and eight sampled candidates per question, using full-parameter GRPO, bf16, gradient checkpointing, and ZeRO-3. All stages use the same Evidence–Summary–Answer protocol and process rewards; only the permitted context length changes. The curriculum first stabilizes evidence selection and structured output on shorter documents, then gradually expands the range of information that the model must locate and compress.

A.8 Question-Conditioned Summary Motivation

Table 7 expands the DocQA-RL-1.6K motivating study in Section 3. Summaries are generated from only the question and document, without access to the reference answer. *Claude Summary* denotes a summary generated by Claude Sonnet 4.6, while *Qwen Self-Summary* denotes one generated by Qwen3.5-Flash itself in a separate pass. Context lengths are means computed uniformly with the Qwen2.5-7B-Instruct-1M tokenizer; a sum gives the document and added-summary lengths separately. Avg is the unweighted mean of the three document-length ranges.

Table 7: **Full results for question-conditioned summaries on long-document QA.** Accuracy is reported by document-length range and as their unweighted average; context tokens report the mean document and summary lengths, and Δ Avg is measured against LongDoc for the same answer model.

Answer model	Answer-time input	Context tokens	0–20K	20–40K	>40K	Avg	Δ Avg
Claude Opus 4.6	LongDoc	15.19K	98.9	89.9	78.3	89.0	–
Claude Opus 4.6	Claude Summary	0.46K	100.0	96.6	95.7	**97.4**	+8.4
Claude Opus 4.6	LongDoc + Claude Summary	15.19K + 0.46K	100.0	96.6	87.0	94.5	+5.5
Qwen3.5-Flash	LongDoc	15.19K	77.3	75.3	65.2	72.6	–
Qwen3.5-Flash	Claude Summary	0.46K	97.7	94.4	87.0	93.0	+20.4
Qwen3.5-Flash	LongDoc + Claude Summary	15.19K + 0.46K	97.7	96.6	95.7	**96.7**	+24.1
Qwen3.5-Flash	LongDoc + Qwen Self-Summary (S)	15.19K + 0.14K	78.4	73.0	56.5	69.3	-3.3
Qwen3.5-Flash	LongDoc + Qwen Self-Summary (M)	15.19K + 0.60K	84.1	79.8	69.6	77.8	+5.2
Qwen3.5-Flash	LongDoc + Qwen Self-Summary (L)	15.19K + 1.04K	85.2	89.9	82.6	85.9	+13.3

Table 8: **Construction statistics for the SFT and RL data.** A case is an independent question–context instance; a block is the unit used for document segmentation and retrieval; a subquery is a decomposed information need; a claim is an independently verifiable atomic fact; and a citation links a claim to its source text. Quality-control counts follow their evaluated objects: faithfulness counts citations that exactly match the source, coverage counts subqueries supported by cited evidence, and answerability counts cases that can be solved from the summary alone. An additional 7,436 citations pass normalized fuzzy-0.95 matching and are reported in the appendix.

Stage	Statistic	Unit	SFT	RL	Total
Blocks Retrieval	Raw candidates	case	5,300	3,000	8,300
	Segmentation	block	1,127,610	657,020	1,784,630
	Ranked retrieval	block	193,031	109,550	302,581
Claims Summaries	Subquery decompose	subquery	17,712	10,232	27,944
	claim extraction	claim	22,107	12,927	35,034
	Canonical claims	claim	21,211	12,325	33,536
	Summary construct	case	5,160	2,945	8,105
Faithfulness Coverage Answerability	Faithfulness	citation	24,523	14,598	39,121
	Coverage	subquery	15,473	8,832	24,305
	Answerability	case	4,539	2,600	7,139
	Final data	**case**	**4,228**	**2,419**	**6,647**

A.9 Qualitative Case Studies

We present seven curated case studies to illustrate how H2S transforms evidence from long documents into question-conditioned summaries and final answers. The cases cover numerical reasoning, cross-document synthesis, evidence-selection errors, arithmetic errors, and evaluation sensitivity. Text shown under **Evidence**, **Summary**, and **Answer** is reproduced verbatim from the original model outputs. Long-document excerpts preserve the original wording, with omitted content marked by `[...]`.

Among the seven examples, Cases 3 and 7 are labeled OOD under our evaluation split. Case 3 illustrates successful cross-document evidence synthesis on AA-LCR, whereas Case 7 illustrates a failure on LongBenchV2. These examples provide qualitative views of H2S behavior under OOD evaluation; as curated cases, they should not be interpreted as independent evidence of aggregate OOD generalization.

A.9.1 Successful Evidence Compression and Reasoning

Case 1: Numerical reasoning from two evidence spans.

Case 1: Numerical Reasoning from Two Evidence Spans

Split: ID **Input length:** 64,784 tokens **Models:** H2S-7B and H2S-14B

Original Question
"what is the percentage change in annual operating cash flow from 2009 to 2010?"

Long Document Excerpt `[vj2m69ie]`
Liquidity and Capital Resources
The following table presents selected financial information and statistics as of and for the three years ended September 24, 2011 (in millions):

	2011	2010	2009
Cash, cash equivalents and marketable securities	$81,570	$51,011	$33,992
[...]	[...]	[...]	[...]
Annual operating cash flow	$37,529	$18,595	$10,159

Cash, cash equivalents and marketable securities increased $30.6 billion or 60% during 2011. [...]

H2S-7B Output
Evidence
1) "Annual operating cash flow was $18,595 million in 2010." `[vj2m69ie]`
2) "Annual operating cash flow was $10,159 million in 2009." `[vj2m69ie]`

Summary
"Annual operating cash flow was $10,159 million in 2009 and $18,595 million in 2010. The percentage change from 2009 to 2010 is approximately 83.0%."

Answer
"83%"
Stored score: 1.0

H2S-14B Output
Evidence
1) "The annual operating cash flow for 2009 was $10,159 million." `[vj2m69ie]`
2) "The annual operating cash flow for 2010 was $18,595 million." `[vj2m69ie]`

Summary
"The annual operating cash flow was $10,159 million in 2009 and $18,595 million in 2010. Therefore, the percentage change from 2009 to 2010 is (18,595 - 10,159) / 10,159 * 100 = 82.99%."

Answer
"82.99%"
Stored score: 1.0

Reference Answer
"83%"

Analysis
The source block contains several financial variables and three years of values. H2S selects only the two operands required by the question, assigns them to the correct years, and explicitly performs the percentage-change calculation in the summary. The final predictions are accepted under numeric-tolerance evaluation.

Figure A7: **Numerical reasoning from a compact evidence set.** Red highlighting marks $18,595 under 2010 and $10,159 under 2009 in the source excerpt. The lower portion shows how these values are transformed into calculation-ready summaries and final answers.

Case 2: Multi-hop reasoning across distributed passages.

Case 2: Multi-Hop Reasoning across Distributed Passages

Split: ID **Input length:** 2,059 tokens **Models:** H2S-7B and H2S-14B

Original Question
"The flavored ice brand Slush Puppie is actually named after a food made from cornmeal-batter. That food was itself originally named after a fish native to South Carolina. What's the genus name of the fish?"

Long Document Excerpt 1 `[3d02d842]`
History
Slush Puppie founder Will Radcliff (1939–2014) decided to start the company after seeing a slush-making machine at a Chicago trade fair in 1970. Radcliff, his sister and their mother came up with the name "Slush Puppie" (based on hush puppy) while sitting on their front porch in Cincinnati, Ohio.

Long Document Excerpt 2 `[e6ef3604]`
Hushpuppy
A hush puppy (or hushpuppy) is a small, savory, deep-fried round ball made from cornmeal-based batter. Hushpuppies are frequently served as a side dish with seafood and other deep-fried foods.

Long Document Excerpt 3 [8aaa80a7]
Name
The first recorded use of the word "hush-puppy" dates to 1899. The name has no verified origin. Etymology is attributed to a variety of sources. Although there are many possible origins for the dish, it is found to be akin to "red-horse bread", named after the red horse fish of the South Carolina rivers. [...]

Long Document Excerpt 4 [fc827d39]
Moxostoma
Moxostoma, the redhorses or jumprocks, is a genus of North American ray-finned fish in the family Catostomidae. [...]

H2S-7B Output
Evidence
1) "The flavored ice brand Slush Puppie is named after the food hushpuppy." [3d02d842]
2) "Hushpuppies are named after the red horse fish of the South Carolina rivers." [8aaa80a7]
3) "The genus name of the red horse fish is Moxostoma." [fc827d39]
Summary
"The flavored ice brand Slush Puppie is named after the food hushpuppy. Hushpuppies are named after the red horse fish of the South Carolina rivers. The genus name of the red horse fish is Moxostoma."
Answer
"Therefore, the answer is Moxostoma."
Stored score: 1.0

H2S-14B Output
Evidence
1) "The flavored ice brand Slush Puppie is named after a food made from cornmeal-batter, specifically hush puppies." [d7e9a65f, 3d02d842]
2) "Hush puppies are small, savory, deep-fried round balls made from cornmeal-based batter." [e6ef3604]
3) "The name hush puppy is attributed to a variety of sources, but it is found to be akin to 'red-horse bread', named after the red horse fish of the South Carolina rivers." [8aaa80a7]
4) "The red horse fish is a member of the genus Moxostoma." [fc827d39]
Summary
"The flavored ice brand Slush Puppie is named after a food made from cornmeal-batter, specifically hush puppies. Hush puppies are small, savory, deep-fried round balls made from cornmeal-based batter. The name hush puppy is attributed to a variety of sources, but it is found to be akin to 'red-horse bread', named after the red horse fish of the South Carolina rivers. The red horse fish is a member of the genus Moxostoma. Therefore, the genus name of the fish is Moxostoma."
Answer
"Therefore, the answer is Moxostoma."
Stored score: 1.0

Reference Answer
"Therefore, the answer is Moxostoma."

Analysis
"A readable multi-hop chain: Slush Puppie → hushpuppy → red horse fish → genus Moxostoma. It clearly demonstrates organization of scattered facts before answering, and both models are correct."

Figure A8: **Multi-hop reasoning across distributed passages.** The four source excerpts establish the Slush Puppie to hushpuppy to red horse fish to Moxostoma chain, which both model summaries preserve.

Case 3: OOD Cross-document synthesis over 123K tokens.

Case 3: Cross-Document Synthesis over 123K Tokens

Split: OOD **Input length:** 122,862 tokens **Models:** H2S-7B and H2S-14B

Original Question
"Throughout all the documents, there are two main LLM threats that are discussed recurrently. These issues are clearly stated in a paper where Baum is one of the authors. In a different paper by Liu and Hu, these two threats are categorised under what umbrella term?"

Long Document Excerpt 1 [2nydst2j]
Soft Begging: Modular and Efficient Shielding of LLMs against Prompt Injection and Jailbreaking based on Prompt Tuning
Simon Ostermann, Kevin Baum, Christoph Endres, Julia Masloh, Patrick Schramowski
Abstract
Prompt injection (both direct and indirect) and jailbreaking are now recognized as significant issues for large language models (LLMs), particularly due to their potential for harm in application-integrated contexts. This extended abstract explores a novel approach to protecting LLMs from such attacks, termed "soft begging." [...]

Long Document Excerpt 2 [i88la43s]
Exploring Vulnerabilities and Protections in Large Language Models: A Survey
Frank Weizhen Liu Chenhui Hu

Abstract
As Large Language Models (LLMs) increasingly become key components in various AI applications, understanding their security vulnerabilities and the effectiveness of defense mechanisms is crucial. This survey examines the security challenges of LLMs, focusing on two main areas: Prompt Hacking and Adversarial Attacks, each with specific types of threats. Under Prompt Hacking, we explore Prompt Injection and Jailbreaking Attacks, discussing how they work, their potential impacts, and ways to mitigate them. `[...]`

H2S-7B Output
Evidence
1) "The two main LLM threats discussed in the paper by Baum are prompt injection and jailbreaking." `[2nydst2j]`
2) "In the paper by Liu and Hu, these two threats are categorized under the umbrella term 'Prompt Hacking'." `[i88la43s]`

Summary
"The two main LLM threats discussed in the paper by Baum are prompt injection and jailbreaking. In the paper by Liu and Hu, these two threats are categorized under the umbrella term 'Prompt Hacking'."

Answer
"Therefore, the answer is Prompt Hacking."
Stored score: 1.0

H2S-14B Output
Evidence
1) "The two main LLM threats discussed recurrently are prompt injection and jailbreaking." `[i88la43s]`
2) "In the Liu and Hu paper, these two threats are categorized under the umbrella term 'Prompt Hacking'." `[i88la43s]`

Summary
"The two main LLM threats discussed recurrently are prompt injection and jailbreaking. In the Liu and Hu paper, these two threats are categorized under the umbrella term 'Prompt Hacking'."

Answer
"Therefore, the answer is Prompt Hacking."
Stored score: 1.0

Reference Answer
"Prompt Hacking"

Analysis
This example requires evidence from two different papers. The first block identifies prompt injection and jailbreaking as the relevant threats, while the second establishes the category relation between those threats and Prompt Hacking. H2S preserves the provenance of both claims and combines them into a directly answerable representation.

Figure A9: **Successful OOD cross-document synthesis over a 122,862-token context.** The two source excerpts are drawn from different papers. H2S identifies the recurring threats and connects them to their umbrella category while preserving source provenance.

Case 4: Ordinal retrieval and exact reproduction.

Case 4: Ordinal Retrieval and Exact Reproduction

Split: ID **Input length:** 64,331 tokens **Models:** H2S-7B and H2S-14B

Original Question
"Prepend cidZrU4URG to the 2nd (1 indexed) short essay about platforms. Do not include any other text in your response."

Long Document Excerpt `[f5jnuira]`
User: write a short essay about platforms
Assistant: Platforms play a transformative role in our contemporary society, serving as the backbone of digital interaction and connectivity. In the broadest sense, platforms are systems or technologies that facilitate interactions, transactions, and collaborations between users, businesses, and services. They have revolutionized the way we communicate, work, and engage with the world around us, enabling an unprecedented level of accessibility and efficiency.
A significant evolution in platforms has been seen with the rise of social media, which has reshaped how we connect and share information. Social media platforms like Facebook, Twitter, and Instagram have become integral parts of daily life, providing spaces for individuals to express themselves, access news, and engage with communities. `[...]`
In conclusion, platforms have become indispensable in today's digital landscape, catalyzing unparalleled connectivity and innovation. They embody the fusion of technology and human interaction, offering immense opportunities while posing new challenges. As platforms continue to evolve, it will be paramount to address these challenges responsibly and harness their potential to foster a more connected, equitable, and efficient world.

H2S-7B Output

Evidence
1) "The second short essay about platforms is the one that begins with 'Platforms play a transformative role in our contemporary society' and ends with 'foster a more connected, equitable, and efficient world.'" `[f5jnuira]`
2) "The final response is the string 'cidZrU4URG' prepended to the second short essay about platforms." `[f5jnuira]`
Summary
"The second short essay about platforms is the one that begins with 'Platforms play a transformative role in our contemporary society' and ends with 'foster a more connected, equitable, and efficient world.' The final response is the string 'cidZrU4URG' prepended to that essay."
Answer
"cidZrU4URGPlatforms play a transformative role in our contemporary society, serving as the backbone of digital interaction and connectivity. In the broadest sense, platforms are systems or technologies that facilitate interactions, transactions, and collaborations between users, businesses, and services. They have revolutionized the way we communicate, work, and engage with the world around us, enabling an unprecedented level of accessibility and efficiency.
`[...]`
In conclusion, platforms have become indispensable in today's digital landscape, catalyzing unparalleled connectivity and innovation. They embody the fusion of technology and human interaction, offering immense opportunities while posing new challenges. As platforms continue to evolve, it will be paramount to address these challenges responsibly and harness their potential to foster a more connected, equitable, and efficient world."
Stored score: 1.0

H2S-14B Output
Evidence
1) "The second short essay about platforms is the one that begins with 'Platforms play a transformative role in our contemporary society' and ends with 'a more connected, equitable, and efficient world.'" `[f5jnuira]`
Summary
"The second short essay about platforms is the one that begins with 'Platforms play a transformative role in our contemporary society' and ends with 'a more connected, equitable, and efficient world.' Its full text is: 'Platforms play a transformative role in our contemporary society, serving as the backbone of digital interaction and connectivity. In the broadest sense, platforms are systems or technologies that facilitate interactions, transactions, and collaborations between users, businesses, and services. [...] As platforms continue to evolve, it will be paramount to address these challenges responsibly and harness their potential to foster a more connected, equitable, and efficient world.'"
Answer
"cidZrU4URGPlatforms play a transformative role in our contemporary society, serving as the backbone of digital interaction and connectivity. In the broadest sense, platforms are systems or technologies that facilitate interactions, transactions, and collaborations between users, businesses, and services. They have revolutionized the way we communicate, work, and engage with the world around us, enabling an unprecedented level of accessibility and efficiency.
`[...]`
In conclusion, platforms have become indispensable in today's digital landscape, catalyzing unparalleled connectivity and innovation. They embody the fusion of technology and human interaction, offering immense opportunities while posing new challenges. As platforms continue to evolve, it will be paramount to address these challenges responsibly and harness their potential to foster a more connected, equitable, and efficient world."
Stored score: 1.0

Reference Answer
"cidZrU4URGPlatforms play a transformative role in our contemporary society, serving as the backbone of digital interaction and connectivity. [...] As platforms continue to evolve, it will be paramount to address these challenges responsibly and harness their potential to foster a more connected, equitable, and efficient world."

Analysis
"A precise-retrieval case at about 64K tokens. Both models locate the second essay, summarize its boundaries, and return the exact requested text with the prefix."

Figure A10: **Ordinal retrieval and exact reproduction at about 64K tokens.** The source excerpt contains the target essay, while the two trajectories identify its ordinal position and reproduce it with the requested prefix.

A.9.2 Failure Modes and Evaluation Sensitivity

Case 5: Evidence selection and arithmetic errors.

Case 5: Evidence Selection and Arithmetic Errors

Split: ID **Input length:** 100,115 tokens **Models:** H2S-7B and H2S-14B

Original Question
"if you held 1000 shares on may 30 , 2014 , how much would you receive in dividends?"

Distractor Document Excerpt `[38eb3185]`
On February 10, 2015, we announced that our board of directors declared a quarterly cash dividend on our common stock of $0.0675 per share. The dividend will be paid on March 10, 2015 to all stockholders of record as of the close of business on February 25, 2015.

Relevant Document Excerpt [9c9e1c33]

Dividend Amount	Declaration Date	Record Date	Payment Date
$0.0425	February 12, 2014	February 25, 2014	March 10, 2014
$0.0425	May 8, 2014	May 27, 2014	June 10, 2014
$0.0425	July 31, 2014	August 25, 2014	September 10, 2014
$0.0675	November 6, 2014	November 25, 2014	December 10, 2014

H2S-7B Output
Evidence
1) "On May 30, 2014, the company declared a quarterly cash dividend of $0.0675 per share." [38eb3185]
2) "The dividend will be paid on March 10, 2015 to all stockholders of record as of the close of business on February 25, 2015." [38eb3185]
3) "If you held 1000 shares on May 30, 2014, you would receive $675 in dividends, calculated as 1000 shares multiplied by the dividend per share of $0.0675." [38eb3185]
Summary
"On May 30, 2014, the company declared a quarterly cash dividend of $0.0675 per share. The dividend will be paid on March 10, 2015 to all stockholders of record as of the close of business on February 25, 2015. If you held 1000 shares on May 30, 2014, you would receive $675 in dividends, calculated as 1000 shares multiplied by the dividend per share of $0.0675."
Answer
"The correct answer is (B)."
Stored score: 0.0

H2S-14B Output
Evidence
1) "The dividend declared on May 8, 2014 was $0.0425 per share." [9c9e1c33]
2) "The dividend declared on May 8, 2014 was paid on June 10, 2014." [9c9e1c33]
Summary
"The dividend declared on May 8, 2014 was $0.0425 per share, and it was paid on June 10, 2014. Therefore, if you held 1000 shares on May 30, 2014, you would receive $4.25 in dividends."
Answer
"4.25"
Stored score: 0.0

Reference Answer
"42.5"

Analysis
The visible source blocks allow the two failures to be distinguished. H2S-7B selects an unrelated 2015 dividend event and incorrectly attributes it to the requested date. Its trajectory therefore fails during evidence selection and subsequently introduces additional arithmetic and formatting errors. H2S-14B selects the correct May 2014 dividend but computes $1{,}000 \times 0.0425$ as 4.25 rather than 42.5, localizing its failure to downstream arithmetic.

Figure A11: **Different failure locations on the same question.** The upper source portion highlights the incorrect 2015 block in red for H2S-7B and the correct May 2014 table row in red for H2S-14B. The lower portion compares the two Evidence–Summary–Answer trajectories.

Case 6: Attribution and temporal-linking failure.

Case 6: Attribution and Temporal-Linking Failure

Split: ID **Input length:** 510 tokens **Models:** H2S-7B and H2S-14B

Original Question
"Rosie Ruiz was disqualified from the Boston Marathon. The subsequent winner placed in what position of the 1988 Grandma's Marathon?"

Long Document Excerpt [a1d2b72b]
Jacqueline_Gareau
Jacqueline Gareau (born March 10, 1953) is a Canadian runner who won the Boston Marathon on April 21, 1980. Gareau led the women's field for most of the race, only to find another runner, Rosie Ruiz, wearing the traditional victor's laurels when she crossed the finish line. Ruiz was later disqualified after it was determined she had cheated and did not run the entire race. Gareau was awarded the victory in a special ceremony one week later. Her official time for the 1980 marathon, 2:34:28, was the fastest time recorded for a woman in the event's history at the time.
Gareau met Ruiz two years after the marathon as she prepared to run a 10K race in Miami, Florida. The encounter was brief and Ruiz refused (as she continued to do until her death) to concede that she did not win the 1980 marathon.
Gareau served as the Grand Marshal of the 2005 Boston Marathon and was allowed to "break the tape" in a special ceremony. She married her former coach, Montreal banker Gilles Lapierre. She has a son, Yannick Lapierre, who participates in Nordic skiing.

H2S-7B Output
Evidence

1) "Rosie Ruiz was disqualified from the Boston Marathon." `[a1d2b72b]`
2) "The subsequent winner of the Boston Marathon was Jacqueline Gareau, who placed first." `[a1d2b72b]`
3) "Jacqueline Gareau placed first in the 1988 Grandma's Marathon." `[a1d2b72b]`
Summary
"Rosie Ruiz was disqualified from the Boston Marathon. The subsequent winner of the Boston Marathon was Jacqueline Gareau, who placed first. Jacqueline Gareau also placed first in the 1988 Grandma's Marathon."
Answer
"Therefore, the answer is First."
Stored score: 0.0

H2S-14B Output
Evidence
1) "Jacqueline Gareau won the Boston Marathon on April 21, 1980, after Rosie Ruiz was disqualified." `[a1d2b72b]`
2) "Jacqueline Gareau placed 10th in the 1988 Grandma's Marathon." `[a1d2b72b]`
Summary
"Jacqueline Gareau won the Boston Marathon on April 21, 1980, after Rosie Ruiz was disqualified. Gareau placed 10th in the 1988 Grandma's Marathon."
Answer
"Therefore, the answer is 10th."
Stored score: 0.0

Reference Answer
"Therefore, the answer is Jacqueline Gareau placed first in the 1988 Grandma's Marathon."

Analysis
"An attribution/temporal-linking failure: both models identify Jacqueline Gareau as the subsequent Boston winner, but disagree on her 1988 Grandma's Marathon placement and both score zero."

Figure A12: **Attribution and temporal-linking failure.** Both trajectories identify Jacqueline Gareau as the subsequent Boston winner, but their linked placement claims for the 1988 Grandma's Marathon disagree.

Case 7: Evidence selection and summary drift.

Case 7: OOD Evidence Selection and Summary Drift

Split: OOD **Input length:** 15,515 tokens **Models:** H2S-7B and H2S-14B

Original Question
"What is the correct answer to this question: Which of the following statements is incorrect?
Choices:
(A) By adjusting the residual gradually during the diffusion process, the model can generate high-resolution images more efficiently.
(B) A complex noise control scheme was designed to flexibly control the switching speed and noise intensity during the diffusion process.
(C) In the forward process, the optimization of θ is achieved by minimizing the negative evidence lower bound
(D) The real data set consists of pictures taken by the camera, photos searched on the Internet, and pictures used in literature"

Long Document Excerpt 1 `[fc428223]`
Fig. 2.
Construction process of training data. Due to the lack of publicly available sonar HR image dataset, we use natural image dataset for training. A large degradation space is used to generate LR images. Moreover, we randomly sample from the task-level LR to construct training-level LR.
`[...]`
Finally, we use the global residual connection to obtain the feature map, and input it into the reconstruction module to generate the SR images $S = \{s_1, s_2, \ldots\}$
$S = \mathrm{HRM}(Y_0 + Y_{\mathrm{DF}}) \quad (9)$
`[...]`
In training stage, we use L1 loss in pixelwise to optimize network parameters
$L = \sum \|s_i - x_i\|_1 \quad (10)$
where b represents the task size of input and $\|\cdot\|_1$ denotes L1 norm.

Long Document Excerpt 2 `[097320dc]`
IV. EXPERIMENT
A. Data
We adopt the DIV2K [51] as the training dataset, which includes 800 training HR images. As mentioned in Section III-A, we first construct a large-scale degradation space to generate task-level LR, and then randomly sample to obtain training-level LR. Finally, training HR–LR pairs are formed by HR and training-level LR.
The testing datasets include synthetic datasets and real sonar image datasets. For the synthetic datasets, we employ BSD100 [52], Urban100 [53], and General100 [54]. Each of these datasets has 100 HR images, and the simulated LR images are generated by various degradation parameters. For the real sonar image datasets, we select three representative images from the KLSG-II [55]. It should be noted that these images do not have HR references.

H2S-7B Output
Evidence
1) "The real data set consists of pictures taken by the camera, photos searched on the Internet, and pictures used in literature." [097320dc]
2) "In the forward process, the optimization of θ is achieved by minimizing the negative evidence lower bound." [fc428223]
3) "A complex noise control scheme was designed to flexibly control the switching speed and noise intensity during the diffusion process." [fc428223]
4) "By adjusting the residual gradually during the diffusion process, the model can generate high-resolution images more efficiently." [fc428223]
Summary
"The correct answer is (C). The real data set consists of pictures taken by the camera, photos searched on the Internet, and pictures used in literature. In the forward process, the optimization of θ is achieved by minimizing the negative evidence lower bound. A complex noise control scheme was designed to flexibly control the switching speed and noise intensity during the diffusion process. By adjusting the residual gradually during the diffusion process, the model can generate high-resolution images more efficiently."
Answer
"The correct answer is (C)."
Stored score: 1.0

H2S-14B Output
Evidence
1) "The training dataset used is the DIV2K dataset, which includes 800 training HR images." [097320dc]
2) "The real sonar image datasets include three representative images from the KLSG-II dataset." [097320dc]
3) "The training data construction process uses a natural image dataset for training due to the lack of publicly available sonar HR image dataset." [fc428223]
Summary
"The training dataset used is the DIV2K dataset, which includes 800 training HR images. The real sonar image datasets include three representative images from the KLSG-II dataset. The training data construction process uses a natural image dataset for training due to the lack of publicly available sonar HR image dataset. Therefore, the real data set does not consist of pictures taken by the camera, photos searched on the Internet, and pictures used in literature; it consists of natural images and sonar images from the KLSG-II dataset."
Answer
"The correct answer is (D)."
Stored score: 0.0

Reference Answer
"C"

Analysis
"A strict-support multiple-choice failure. H2S-7B is correct while H2S-14B selects a different option after its summary focuses on irrelevant dataset details, illustrating summary-induced distraction."

Figure A13: **Evidence selection and summary drift in an OOD example of H2S. H2S-7B identifies the incorrect statement as (C), whereas the H2S-14B summary concentrates on dataset details and selects (D).**