CCoR: Reasoning-Aware Context Compression via Submodular Information Gain

Anonymous authors
Paper under double-blind review

Abstract

Context compression has emerged as a promising paradigm to reduce high computational costs and redundant noise in Large Language Models (LLMs) across data-intensive scenarios. Existing methods span task-agnostic and task-aware approaches. Task-agnostic methods sacrifice fine-grained details by operating independently of downstream queries, whereas current task-aware paradigms remain strictly query-centric. Consequently, only text semantically relevant to the query is selectively retained, inadvertently discarding critical logical scaffolding required for multi-step derivations. In this work, we propose **CCoR** (**C**ontext **C**ompression f**o**r **R**easoning), a novel framework that preserves essential reasoning evidence rather than relying solely on query semantic relevance. Specifically, CCoR formulates context importance evaluation as a submodular maximization problem to rigorously capture sufficiency and necessity for model reasoning. Building upon this, CCoR introduces an adaptive compression mechanism that preserves high-importance chunks into distinct representations while condensing redundant content into latent summaries, optimized jointly via supervised alignment and reasoning-preservation reinforcement learning. Extensive experiments show that CCoR substantially outperforms existing baselines, achieving robust reasoning preservation under stringent compression rates. Notably, under $8\times$ compression, CCoR achieves an average 9.23% F1 gain over state-of-the-art (SOTA) across three long-context benchmarks using LLaMA-3.1-8B-Instruct.

1 Introduction

Large Language Models (LLMs) have achieved remarkable success across diverse applications Guo et al. (2025); El-Kishky et al. (2025). However, deploying LLMs in data-intensive scenarios such as Retrieval-Augmented Generation (RAG) Yang et al. (2018); Hwang et al. (2024) and multi-agent workflows Li et al. (2024b) often involves processing massive input contexts. This massive scale induces severe computational bottlenecks due to the quadratic complexity of self-attention Katharopoulos et al. (2020), while the proliferation of irrelevant noise heavily degrades generation fidelity Tang et al. (2026d). To address these challenges, context compression has emerged as a promising paradigm to reduce sequence lengths and eliminate redundancy.

Existing context compression strategies generally fall into two primary categories: task-agnostic Jiang et al. (2024); Ge et al. (2023); Ye et al. (2025); Tang et al. (2025); Liao et al. (2025) and task-aware Cao et al. (2024); Liskavets et al. (2025); Zhao et al. (2025); Fang et al. (2025) methods. As shown in Figure 1 (A), task-agnostic methods compress the input context independently of downstream queries to retain general semantic coverage Pan et al. (2024). However, without guidance from the specific target task, these methods inevitably discard fine-grained details or dilute critical signals under high compression ratios Tang et al. (2026c). Conversely, as shown in Figure 1 (B), task-aware methods incorporate the query during compression to filter and retain task-relevant information. Despite their effectiveness in specific retrieval tasks, current task-aware paradigms suffer from a fundamental limitation: they are *strictly query-centric*. They rely heavily on query-centric semantic similarity to evaluate and optimize context importance Tang et al. (2026b;c; 2025); Hwang et al. (2024). Consequently, they implicitly assume that an input's utility is measured solely by its semantic alignment with the query, completely overlooking the reality that *semantic relevance does not guarantee reasoning necessity.* Modern LLM outputs rely heavily on intermediate reasoning paths, logical state tracking, and multi-step derivation, which often demand specialized evidence

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.

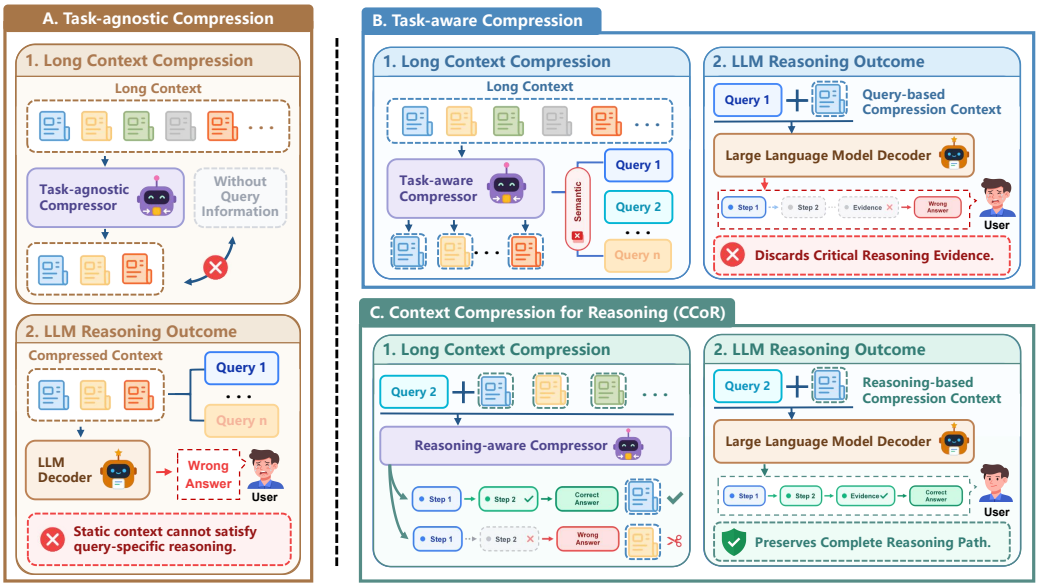

Figure 1: **Comparison of context compression paradigms.** Unlike existing methods, our proposed **CCoR** preserves reasoning evidence to achieve accurate LLM reasoning under compression.

beyond surface-level query matching Wei et al. (2022); Sprague et al. (2025). As shown in Figure 1 (B), by indiscriminately pruning or merging content based solely on query-dependent semantic similarity, existing methods discard crucial logical scaffolding, severely impairing the model's capacity to execute internal reasoning and recover robust execution trajectories Jiang et al. (2024).

Motivated by this observation, in this work, we investigate a critical research question to context compression: ***"How can we preserve the essential evidence required for model reasoning within compressed contexts, rather than relying solely on query semantic relevance?"*** As shown in Figure 1 (C), we propose **CCoR** (**C**ontext **C**ompression f**o**r **R**easoning), a novel framework designed to jointly eliminate semantic redundancy and reconstruct reasoning under high compression rates. To accurately capture the true utility of context chunks for model reasoning beyond superficial query semantic, we formulate this selection as a submodular maximization problem Krause & Golovin (2014). This allows us to rigorously evaluate the sufficiency and necessity of context information for model reasoning, enabling autonomous and reliable importance annotation via an efficient greedy strategy Nemhauser et al. (1978). Building upon this, CCoR introduces an adaptive compression mechanism: high-importance chunks are preserved into distinct representations to retain necessary reasoning evidence, whereas less critical ones are collectively compressed into compact latent summary representations guided by the query under stringent compression constraints. CCoR then concatenates these individual representations with the latent summary vectors to form a hybrid context representation, which is optimized via supervised training to align the decoder with the compressed space. We further introduce a reinforcement learning strategy driven by reasoning-preservation objectives, optimizing the decoder's utilization of the hybrid representations, which ultimately achieves robust reasoning preservation under context compression.

Our contributions are as follows: **(1) Reasoning Information Modeling (§5.1):** We formulate context importance evaluation as a submodular maximization problem, rigorously modeling the sufficiency and necessity of context chunks for model reasoning. This overcomes the limitations of query-centric semantic relevance and provides a principled, discriminative framework for identifying essential reasoning evidence. **(2) Context Compression for Reasoning (§5.2):** We propose CCoR, a novel framework featuring an adaptive context compression mechanism. CCoR preserves high-importance chunks into distinct representations while collectively compressing redundant content into latent summaries, optimized jointly through supervised alignment and reasoning-preservation reinforcement learning. **(3) Outstanding Performance (§6):** We conduct comprehensive evaluations across three complex reasoning and long-context benchmarks. Experimental results demonstrate that CCoR substantially outperforms existing task-aware and task-agnostic compression baselines, achieving robust reasoning preservation under high compression ratios.

2 PRELIMINARY

This section briefly introduces definitions of context Compression, which is the primary application scenario of our work.

Context Compression. Let $\mathcal{M}_d$ denote a large language model parameterized by θ, and let $\mathcal{Q}$ represent the user query accompanied by a massive input context sequence $\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$ consisting of N tokens. In standard generation, the model computes the probability distribution of the target output via autoregressive decoding. To precisely capture the information flow during complex multi-step reasoning, we explicitly decompose the model's output into an explicit reasoning chain $\mathcal{R} = \{r_1, r_2, \ldots, r_M\}$ and a final answer $\mathcal{Y} = \{y_1, y_2, \ldots, y_K\}$:

$$P_d(\mathcal{R}, \mathcal{Y} \mid \mathcal{Q}, \mathcal{X}; \theta) = \prod_{i=1}^{M} P_d(r_i \mid \mathcal{Q}, \mathcal{X}, r_{<i}; \theta) \prod_{j=1}^{K} P_d(y_j \mid \mathcal{Q}, \mathcal{X}, \mathcal{R}, y_{<j}; \theta). \tag{1}$$

When processing long contexts, the quadratic memory and computational footprint of the self-attention mechanism necessitates context compression Jiang et al. (2024); Liao et al. (2025). The compression process maps the raw input sequence $\mathcal{X}$ into a significantly shorter compressed context or representation $\tilde{\mathcal{X}} = \{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_T\}$, where $T \ll N$, under a strict compression budget ratio $\gamma = M/N$. The downstream model then performs reasoning and generation conditioned on this compressed context:

$$(\hat{\mathcal{R}}, \hat{\mathcal{Y}}) \sim P_d(\mathcal{R}, \mathcal{Y} \mid \mathcal{Q}, \tilde{\mathcal{X}}; \theta). \tag{2}$$

Therefore, the core objective of context compression is to preserve a compressed context $\tilde{\mathcal{X}}$ that enables the model to yield faithful reasoning $\hat{\mathcal{R}}$ and answer $\hat{\mathcal{Y}}$ comparable to the uncompressed counterpart. However, existing task-aware methods Tang et al. (2026b;c); Cao et al. (2024) predominantly rely on the query's semantic relevance as the sole criterion for selecting $\tilde{\mathcal{X}}$, fundamentally overlooking the complex information demanded by the reasoning trajectory $\mathcal{R}$. In §4, we empirically reveal the significant gap between semantic relevance and reasoning-critical information, and subsequently introduce our reasoning-aware context compression framework (CCoR) in §5.

3 RELATED WORKS

LLM with Long Context. Long-context capabilities have become increasingly important for large language models (LLMs), supporting applications such as retrieval-augmented generation Yang et al. (2018); Hwang et al. (2024), in-context learning Bai et al. (2024), conversational memory Maharana et al. (2024), and agentic workflows Li et al. (2024b). Prior studies have explored context-window extension Yang et al. (2025) and efficient long-sequence modeling Xu et al. (2026) to enable LLMs to process increasingly large amounts of contextual information. Beyond simply retrieving relevant content Chen et al. (2026b), many long-context tasks require models to integrate evidence distributed across distant parts of the input and perform multi-step reasoning over such information Bai et al. (2025). However, processing lengthy contexts remains computationally expensive and often introduces substantial redundancy. This motivates context compression as a complementary direction for constructing compact representations that preserve not only useful information, but also the evidence necessary for downstream reasoning.

Context Compression. Existing context compression methods can be broadly categorized into task-agnostic and task-aware approaches. Task-agnostic methods compress contexts without downstream queries, typically through token pruning Pan et al. (2024), summarization, or latent representations such as memory and soft tokens Dai et al. (2025); Ge et al. (2023); Xu et al. (2023); Tang et al. (2025); Guo et al. (2026), aiming to preserve general semantic information for broad applicability. However, without task-specific guidance, they may discard fine-grained evidence required by a particular query. Task-aware methods instead incorporate the query into compression, using query-guided summarization Cui et al. (2025), token merging Tang et al. (2026c), or relevance-based pruning Tang et al. (2026b;a) to retain task-specific information. While effective, these methods still largely estimate context importance through query-conditioned relevance signals Cao et al. (2024). Such relevance does not necessarily reflect reasoning utility, as intermediate or bridging evidence may be weakly aligned with the query yet essential for multi-step derivation Ma et al. (2026). In contrast, our work models context importance according to its contribution to the downstream reasoning process, enabling more faithful preservation of reasoning-critical evidence under compression.

4 Information Gap Between Relevance and Reasoning

In this section, we conduct a pilot study to investigate the information loss inflicted on model reasoning by query-centric semantic similarity, which serves as the prevailing relevance metric adopted by existing task-aware compression methods Tang et al. (2026b;c); Hwang et al. (2024); Chirkova et al. (2025); Cao et al. (2024).

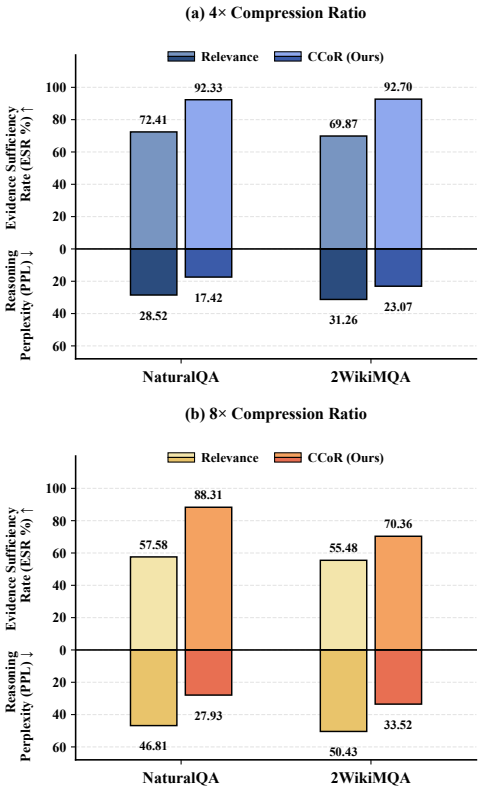

Figure 2: Pilot study on the information gap between relevance and reasoning.

Experimental Setup. We construct an evaluation benchmark by sampling 3,000 queries from long-context QA datasets, specifically NaturalQA Kwiatkowski et al. (2019) and 2WikiMQA Ho et al. (2020). Using Qwen3-8B Yang et al. (2025) as the backbone model, we first process the full, uncompressed context for each query to establish the ground-truth reasoning paths and answers. We then segment each long context into uniform text chunks of L_{seg} tokens. Under target compression ratios α of $4\times$ and $8\times$, where the retained chunk count $k = \lfloor \frac{L_{\text{org}}}{\alpha L_{\text{seg}}} \rfloor$ is determined by the original context length L_{org}, we calculate the cosine similarity between chunk embeddings and the query, subsequently retaining the top-k chunks with the highest semantic relevance.

Evaluation Metrics. To quantify reasoning preservation in selected chunks, we evaluate two metrics: (1) **Evidence Sufficiency Rate (ESR)**, where Qwen3-235B-A22B-Instruct Yang et al. (2025) judges whether retained chunks contain sufficient evidence to reproduce the original reasoning path; and (2) **Generation Perplexity** Cooper & Scholak (2024), measured on Qwen3-8B over the original reasoning chain, where higher perplexity reflects greater reasoning flow disruption.

Experimental Results. As depicted in Figure 2, conventional query-centric relevance selection exhibits severe evidence loss and reasoning flow disruption as compression tightens. Under $8\times$ compression, its Evidence Sufficiency Rate (ESR) plummets to 57.58% on NaturalQA Kwiatkowski et al. (2019) and 55.48% on 2WikiMQA Ho et al. (2020), while the generation perplexity on ground-truth reasoning trajectories surges to 46.81 and 50.43, respectively. These empirical findings validate our central premise: relying solely on semantic similarity inadvertently discards critical intermediate evidence required for multi-step reasoning. In contrast, CCoR effectively bridges this gap, maintaining substantially higher evidence coverage (up to 92.70% ESR) and lower perplexity across all settings. Qualitative examples further illustrating this gap between semantic relevance and reasoning utility are provided in Appendix C.3. We detail the chunk selection mechanism of CCoR in §5.1.

5 Context Compression for Reasoning

To address the reasoning information loss revealed in §4, we propose **CCoR**, a novel reasoning-aware compression framework. As illustrated in Figure 3, CCoR establishes a multi-stage pipeline that first identifies reasoning-critical context chunks, and then adaptively compresses residual content while preserving reasoning capabilities.

5.1 Reasoning-Aware Chunk Selection

To identify context segments that are truly essential for model reasoning rather than merely semantically aligned with the query, we formulate chunk selection as a submodular maximization problem Krause & Golovin (2014). Given a query $\mathcal{Q}$, input context sequence $\mathcal{X}$, and ground-truth answer

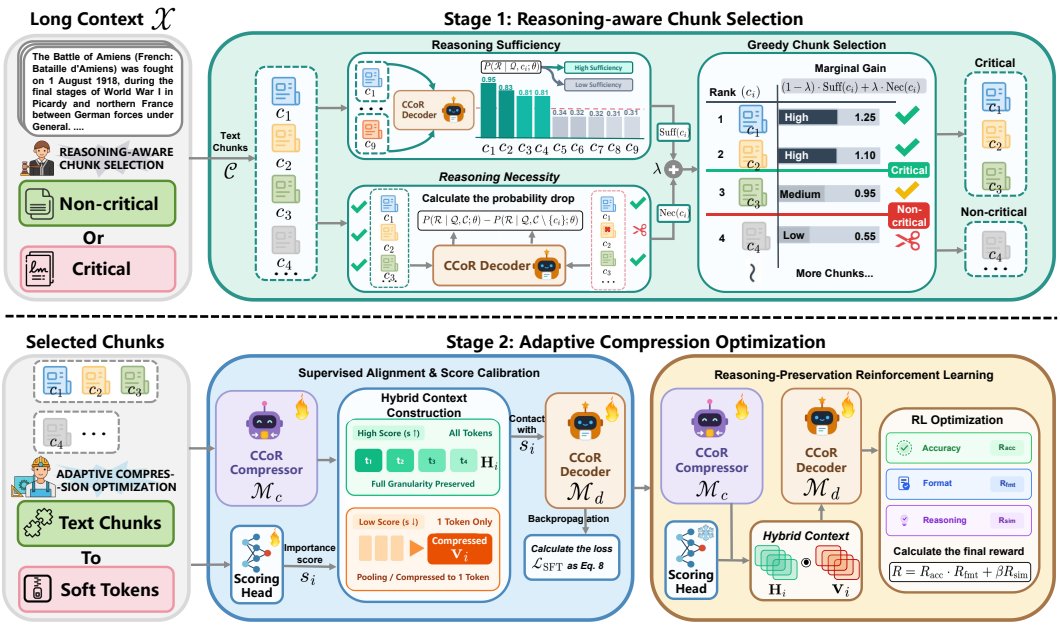

Figure 3: Overview of CCoR framework. **Stage 1 (Reasoning-Aware Chunk Selection §5.1)** selects critical context chunks via submodular maximization based on sufficiency and necessity. **Stage 2 (Adaptive Compression Optimization §5.2)** compresses residual context into latent summaries, followed by alignment and RL to sustain reasoning capabilities.

$\mathcal{Y}$ from the training set, we first obtain explicit reasoning paths generated by the decoder model $\mathcal{M}_d$. We filter out erroneous trajectories, retaining only valid reasoning chains $\mathcal{R} = \{r_1, r_2, \ldots, r_M\}$ that successfully yield $\mathcal{Y}$. Next, we partition the long context $\mathcal{X}$ into a candidate set of uniform text chunks $\mathcal{C} = \{c_1, c_2, \ldots, c_U\}$, where each chunk contains L_{seg} tokens. Under a target compression ratio α, our objective is to select a subset of top-k critical chunks $\mathcal{C}^* \subseteq \mathcal{C}$, where the retention budget $k = \lfloor \frac{L_{\text{org}}}{\alpha L_{\text{seg}}} \rfloor$ is governed by the original context length L_{org}.

To quantify the contribution of context chunks toward faithfully reproducing the valid reasoning trajectory $\mathcal{R}$, we evaluate both their reasoning sufficiency and necessity. Reasoning sufficiency measures the likelihood sum of generating correct reasoning chain given a candidate subset $S \subseteq \mathcal{C}$:

$$\text{Suff}(S) = P_d(\mathcal{R} \mid \mathcal{Q}, S; \theta) = \textstyle\sum_{i=1}^{M} P_d(r_i \mid \mathcal{Q}, S, r_{<i}; \theta). \tag{3}$$

Conversely, reasoning necessity captures the indispensable logical contribution of subset S by measuring the probability when S is omitted from the complete candidate set $\mathcal{C}$:

$$\text{Nec}(S) = \textstyle\sum_{i=1}^{M} \left[1 - P_d(r_i \mid \mathcal{Q}, \mathcal{C} \setminus S, r_{<i}; \theta)\right]. \tag{4}$$

By integrating both dimensions, we define the chunk subset utility function $f(S)$ over candidate subset S and formulate the submodular optimization problem as:

$$\mathcal{C}^* = \arg \max_{S \subseteq \mathcal{C}, |S| \leq k} f(S) = \arg \max_{S \subseteq \mathcal{C}, |S| \leq k} \left[(1-\lambda) \cdot \text{Suff}(S) + \lambda \cdot \text{Nec}(S)\right], \tag{5}$$

where $0 < \lambda < 1$ balances sufficiency and necessity. Directly maximizing Equation 5 is NP-hard, so we employ an efficient greedy search strategy Nemhauser et al. (1978), as detailed in Algorithm 1. Chunks are iteratively selected by evaluating the marginal gain $\Delta f(c_i \mid S) = f(S \cup \{c_i\}) - f(S)$ along the sequence. To quantify evidence importance, we normalize these marginal gains across all chunks to obtain a target importance score $s_i \in [0, 1]$ for each c_i. Segments providing substantial marginal gains are recognized as critical reasoning evidence; as the marginal gains diminish toward zero, subsequent chunks yield negligible additional utility, yielding the optimal top-k chunk subset $\mathcal{C}^*$. We design $f(S)$ as a submodularity-inspired objective to model the diminishing returns of evidence accumulation. Although $f(S)$ may not strictly satisfy theoretical submodularity for complex LLMs, such approximations have been empirically validated as highly effective in practical language modeling tasks Chen et al. (2026a); Wei et al. (2015); Chang et al. (2026). We provide preliminaries and theoretical guarantees in Appendix A.1 and Appendix A.2.

Algorithm 1 Greedy Reasoning-Aware Chunk Selection

Require: Candidate chunks $\mathcal{C} = \{c_1, \ldots, c_U\}$, query $\mathcal{Q}$, target budget k, balance factor λ.
Ensure: Optimal top-k chunk subset $\mathcal{C}^*$.
1: Initialize selected set $\mathcal{C}^* \leftarrow \emptyset$.
2: **while** $|\mathcal{C}^*| < k$ **do**
3: **for** each $c_i \in \mathcal{C} \setminus \mathcal{C}^*$ **do**
4: Compute marginal gain $\Delta f(c_i \mid \mathcal{C}^*) = f(\mathcal{C}^* \cup \{c_i\}) - f(\mathcal{C}^*)$ using Eq. 5.
5: **end for**
6: Select best chunk: $c^* \leftarrow \arg\max_{c_i \in \mathcal{C} \setminus \mathcal{C}^*} \Delta f(c_i \mid \mathcal{C}^*)$.
7: Update selected set: $\mathcal{C}^* \leftarrow \mathcal{C}^* \cup \{c^*\}$.
8: **end while**
9: **return** $\mathcal{C}^*$

5.2 Adaptive Compression Optimization

Following chunk selection, the long context $\mathcal{X}$ is partitioned into critical chunks $\mathcal{C}^*$ and non-critical chunks $\mathcal{C} \setminus \mathcal{C}^*$. To maintain reasoning scaffolding while minimizing context length, we employ a lightweight compressor model $\mathcal{M}_c$ to construct a hybrid context representation $\tilde{\mathcal{X}}$.

Hybrid Context Construction. For critical chunks $c_i \in \mathcal{C}^*$, we retain their complete token hidden state sequences $\mathbf{H}_i = \mathcal{M}_c(c_i) = [\mathbf{h}_{i,1}, \ldots, \mathbf{h}_{i,L_{\text{seg}}}]$. For non-critical chunks $c_i \in \mathcal{C} \setminus \mathcal{C}^*$, we condense their L_{seg} tokens into a single latent vector $\mathbf{v}_i$. Specifically, we extract the query vector $\mathbf{r}_q$ via mean pooling over query token hidden state, and compute query-guided semantic weights to aggregate token hidden states:

$$w_{i,t} = \text{Softmax}_t\left(\text{Sim}(\mathbf{h}_{i,t}, \mathbf{r}_q)\right), \quad \mathbf{v}_i = \sum_{t=1}^{L_{\text{seg}}} w_{i,t}\mathbf{h}_{i,t}, \tag{6}$$

where $\text{Sim}(\mathbf{h}_{i,t}, \mathbf{r}_q)$ denotes semantic cosine similarity. Since computing submodular importance via greedy search is computationally prohibitive during inference, we attach a lightweight scoring head $g(\cdot)$ to the compressor. It predicts an importance score $\text{Sigmoid}_i(g(c_i, \mathcal{Q})) = \hat{s}_i \in [0, 1]$ for each chunk c_i, approximating the normalized submodular utility computed in §5.1. To accentuate fine-grained evidence saliency, we modulate the token representations by multiplying them with their corresponding normalized score $\hat{s}_i$. All preserved token sequences and compressed vectors are then concatenated according to their original sequential order, forming the hybrid context sequence $\tilde{\mathcal{X}} = \{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_U\}$:

$$\tilde{x}_i = \begin{cases} \mathbf{H}_i\hat{s}_i, & \text{if } c_i \in \mathcal{C}^*, \\ \mathbf{v}_i\hat{s}_i, & \text{if } c_i \in \mathcal{C} \setminus \mathcal{C}^*. \end{cases} \tag{7}$$

Supervised Alignment & Score Calibration. We first perform end-to-end supervised fine-tuning (SFT) on the compressor $\mathcal{M}_c$, scoring head $g(\cdot)$, and downstream decoder $\mathcal{M}_d$. The objective combines auto-regressive generation loss with binary cross-entropy (BCE) classification loss for score calibration:

$$\mathcal{L}_{\text{SFT}} = -\log P_d(\mathcal{R}, \mathcal{Y} \mid \mathcal{Q}, \tilde{\mathcal{X}}) + \sum_{i=1}^{U} \text{BCE}(\hat{s}_i, s_i). \tag{8}$$

Reasoning-Preservation Reinforcement Learning. To further empower the decoder $\mathcal{M}_d$ to decode robust reasoning trajectories from compressed latent summaries, we incorporate a reinforcement learning stage using Group Relative Policy Optimization (GRPO) Shao et al. (2024). The total reward R evaluates the generated trajectory $\hat{\mathcal{R}}$ and answer $\hat{\mathcal{Y}}$ across three dimensions:

$$R = R_{\text{acc}} \cdot R_{\text{fmt}} + \beta R_{\text{sim}}, \quad R_{\text{sim}} = \text{Sim}(\hat{\mathcal{R}}, \mathcal{R}), \tag{9}$$

where R_{acc} is an outcome-based accuracy reward, R_{fmt} enforces formatting compliance and R_{sim} measures semantic similarity over the compressor's hidden states. This semantic restoration reward explicitly encourages the decoder to recover faithful intermediate reasoning trajectories despite context compression. Detailed optimization settings and the practical reward instantiation are provided in Appendix B.3, while the limitations of CCoR are discussed in Appendix A.3.

6 EXPERIMENTS

Table 1: Experimental results on three QA benchmarks. **Bold** denotes the best results for each backbone under each compression constraint. **F1** and **EM** denote F1 score and Exact Match, respectively. **AVG** reports the arithmetic mean over the three benchmarks. **Closed-book** uses only the question as input, whereas **Original Prompt** leverages the full context.

Methods	NaturalQA		2WikiMQA		HotpotQA		AVG	
	F1	EM	F1	EM	F1	EM	F1	EM
Backbone: LLaMA-3.1-8B-Instruct								
Closed-book	36.51	23.77	21.15	14.60	26.26	17.89	27.97	18.75
Original Prompt	56.04	43.28	54.03	44.83	70.08	55.30	60.05	47.80
EXIT	38.32	25.69	23.45	14.54	40.29	29.22	34.02	23.15
4× Compression Constraint								
LLMLingua-2-large	46.66	33.15	37.07	30.38	50.89	36.95	44.87	33.49
LongLLMLingua	58.25	45.61	39.32	33.09	57.41	41.55	51.66	40.08
ICAE	50.35	38.49	35.59	28.98	37.95	26.97	41.30	31.48
Activation Beacon	58.97	47.04	44.20	36.69	57.44	43.78	53.54	42.50
PCC	42.11	29.72	47.03	38.85	45.57	31.10	44.90	33.22
COMI	62.09	49.83	50.31	41.60	51.72	37.58	54.71	43.00
RAM	62.77	51.15	57.50	49.51	65.65	52.28	61.97	50.98
CCoR	**71.29**	**60.68**	**71.51**	**63.51**	**67.56**	**53.18**	**70.12**	**59.12**
8× Compression Constraint								
LLMLingua-2-large	36.36	24.14	27.23	22.84	37.60	26.28	33.73	24.42
LongLLMLingua	51.97	38.27	33.47	28.84	50.21	36.27	45.22	34.46
ICAE	46.99	35.89	36.33	30.39	36.79	26.43	40.04	30.90
Activation Beacon	55.09	41.21	42.98	35.53	48.57	36.29	48.88	37.68
PCC	49.59	37.02	38.10	31.55	43.19	30.02	43.63	32.86
COMI	56.67	43.31	48.00	38.69	47.53	33.27	50.73	38.42
RAM	61.24	48.96	49.57	41.79	57.42	44.78	56.08	45.18
CCoR	**69.92**	**59.10**	**64.47**	**56.21**	**61.54**	**47.76**	**65.31**	**54.36**
Backbone: Qwen3-8B								
Closed-book	22.86	11.17	27.58	22.15	25.64	16.27	25.36	16.53
Original Prompt	60.44	46.40	48.45	39.33	67.68	53.06	58.86	46.26
EXIT	31.20	21.02	32.33	27.93	41.68	30.70	35.07	26.55
4× Compression Constraint								
LLMLingua-2-large	43.32	28.96	35.60	29.76	48.67	34.99	42.53	31.24
LongLLMLingua	57.30	43.50	39.34	32.78	56.80	41.44	51.15	39.24
ICAE	32.43	21.69	28.64	23.06	32.37	22.08	31.15	22.28
PCC	44.06	25.46	25.93	19.47	33.04	19.78	34.34	21.57
COMI	23.70	14.43	33.10	27.01	32.59	21.70	29.80	21.05
RAM	56.03	44.41	45.08	38.44	57.97	45.05	53.03	42.63
CCoR	**65.65**	**53.45**	**60.73**	**52.58**	**58.87**	**45.58**	**61.75**	**50.54**
8× Compression Constraint								
LLMLingua-2-large	33.22	20.49	29.85	26.33	36.35	25.20	33.14	24.01
LongLLMLingua	49.61	35.33	34.97	29.80	**49.60**	35.11	44.73	33.41
ICAE	35.67	23.47	30.91	25.36	32.85	22.27	33.14	23.70
PCC	35.84	19.44	25.16	19.20	23.91	12.43	28.30	17.02
COMI	23.53	14.65	31.99	26.49	30.99	21.03	28.84	20.72
RAM	54.97	42.93	38.52	32.69	48.89	37.18	47.46	37.60
CCoR	**63.81**	**52.13**	**47.93**	**41.25**	49.02	**37.39**	**53.59**	**43.59**

Training Setup. We instantiate CCoR on two open-source backbones with strong reasoning capability, `Llama-3.1-8B-Instruct` Grattafiori et al. (2024) and `Qwen3-8B` Yang et al. (2025). We conduct experiments on three long-context question answering benchmarks: NaturalQuestions Kwiatkowski et al. (2019), 2WikiMQA Ho et al. (2020), and HotpotQA Yang et al. (2018). We evaluate performance under both $4\times$ and $8\times$ compression constraint. Additional dataset statistics and implementation details are provided in Appendix B.1 and Appendix B.3. Following Hwang et al. (2024), we adopt Exact Match (EM) and F1 Yang et al. (2018) as evaluation metrics. EM evaluates exact answer correctness, while F1 captures token-level overlap and partial matching, jointly assessing the quality and effectiveness of compressed contexts for downstream QA tasks.

Baselines. We evaluate CCoR against representative context compression baselines: (1) **Task-agnostic Compression**, which reduces context independently of the downstream query, including LLMLingua-2-large Pan et al. (2024), ICAE Ge et al. (2023), Activation Beacon Zhang et al. (2024), and PCC Dai et al. (2025); and (2) **Task-aware Compression**, which leverages query/task information to guide context compression, including LongLLMLingua Jiang et al. (2024), EXIT Hwang et al. (2024), COMI Tang et al. (2026c), and RAM Tang et al. (2026b). For fair comparison, all baselines are reproduced per their official settings using the same backbone and datasets as ours. Detailed descriptions of all baselines are provided in Appendix B.2.

6.1 Main Results

As shown in Table 1, CCoR achieves superior performance across all three QA benchmarks, consistently outperforming existing compression approaches in overall average F1 and Exact Match score across both LLaMA-3.1-8B and Qwen3-8B backbones under both $4\times$ and $8\times$ compression constraints. We observe that context compression is not merely a trade-off for efficiency,under appropriate constraints, compressed contexts can even outperform the uncompressed *Original Prompt*. For instance, under $4\times$ compression on LLaMA-3.1-8B, competitive baselines such as RAM surpass full context by 1.92% F1 score on average (and by 6.73% F1 score on NaturalQA). This confirms that filtering out irrelevant context effectively mitigates noise distraction present in full context. Regarding baseline performance, task-aware compression methods (e.g., COMI, RAM) generally outperform task-agnostic counterparts (e.g., LLMLingua-2, ICAE). For instance, RAM surpasses LLMLingua-2 by 17.10% F1 score on average on LLaMA-3.1-8B ($4\times$). This confirms that query guidance is crucial for retaining task-relevant context. In contrast, going beyond traditional query-conditioned relevance, CCoR models context importance through reasoning utility and substantially exceeds the SOTA task-aware baseline (RAM) by 8.15% and 8.72% average F1 score on LLaMA-3.1-8B and Qwen3-8B under $4\times$ compression, respectively. This underscores that prioritizing reasoning-critical evidence effectively purges irrelevant noise present in full context without losing multi-step derivation paths. We further provide an efficiency analysis of CCoR in Appendix C.2.

6.2 Ablation Study

To verify the effectiveness of each key component in CCoR, we conduct ablation studies on NaturalQA and 2WikiMQA, as shown in Table 2. We consider three ablation variants: *i) w/o RL* removes reasoning-preservation reinforcement learning and retains only supervised alignment; *ii) w/o Explicit Chunk Preservation* (ECP) uniformly compresses all chunks into soft representations without explicitly preserving critical chunks; *iii) w/o Reasoning-aware Selection* (RS) replaces reasoning-aware chunk selection with query–chunk semantic similarity. The ablations reveal three complementary roles in CCoR. Reasoning-preservation RL improves the model's ability to recover useful information from compressed representations, providing additional robustness when the learned importance scores are imperfect. Explicitly preserving critical chunks is crucial for maintaining fine-grained reasoning evidence, while soft tokens retain complementary information from the remaining context, together forming an effective hybrid representation. Reasoning-aware selection further ensures that the preserved chunks are chosen for their contribution to the reasoning process rather than merely their semantic similarity to the query.

Table 2: Ablation study on LLaMA-3.1-8B under $4\times$ compression. All EM and F1 scores are reported in percentage (%).

Methods	NaturalQA		2WikiMQA		HotpotQA	
	EM	F1	EM	F1	EM	F1
CCoR (Default)	**60.68**	**71.29**	**63.51**	**71.51**	**53.18**	**67.56**
w/o RL	57.14	68.39	63.24	70.86	52.57	66.99
w/o ECP	43.84	56.26	55.16	64.16	41.77	55.76
w/o RS	51.15	62.77	49.51	57.50	52.58	65.65

6.3 Effectiveness of CCoR Hybrid Context

To understand why CCoR's hybrid context is effective, we evaluate its reasoning preservation via *Information Gain* and *Token Gain* Kuratov et al. (2025). Information Gain is measured by the cross-entropy (CE) of the gold reasoning chain conditioned on the hybrid context, where lower values indicate richer evidence for trajectory reconstruction. Token Gain denotes the token-level F1 between the generated reasoning chain and gold CoT, quantifying how much preserved reasoning is recovered during generation. As shown in Table 3, during the different stages of SFT training,

Chain CE steadily decreases from 0.731 to 0.358, while Token Gain increases from 59.1 to 78.3, accompanied by consistent improvements in downstream performance. The evolution of these signals suggests that training progressively teaches the model to identify evidence that is critical to reasoning and preserve it within hybrid context, rather than merely compressing semantically relevant content. Consequently, the compressed context becomes increasingly informative for reconstructing reasoning paths, enabling compression without lossing reasoning capability.

Table 3: SFT Training dynamics of the hybrid context under $4\times$ compression.

Stage	Information Gain	Token Gain	NaturalQA		2WikiMQA	
	Chain CE↓	Token F1↑	EM	F1	EM	F1
Early	0.731	59.10	30.36	43.62	29.68	36.65
Middle	0.496	71.90	50.73	62.44	52.87	61.39
Final	**0.358**	**78.30**	**57.97**	**68.83**	**62.75**	**71.39**

6.4 Applicability to Extra Compression Rates

To evaluate the robustness of CCoR under different compression budgets, we conduct experiments with compression ratios ranging from $4\times$ to $32\times$. As illustrated in Figure 4, CCoR consistently outperforms competing methods on 2WikiMultihopQA across all settings, with parallel results for NaturalQuestions detailed in Appendix C.1. We attribute this resilience to the fact that critical reasoning information is typically concentrated within a limited number of chunks. By accurately identifying and retaining these vital segments, our reasoning-aware mechanism reduces evidence loss. Consequently, this targeted preservation significantly amplifies CCoR's performance advantage, particularly under highly aggressive compression ratios.

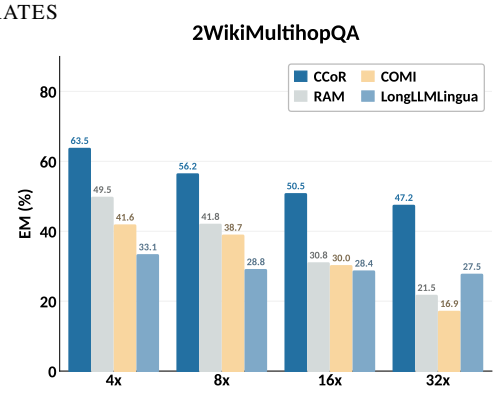

Figure 4: Performance under different compression ratios.

6.5 Applicability to Model Scaling

To evaluate the scalability of CCoR, we synchronously scale both the compressor and decoder models across four model scales using the Qwen3 backbone (1.7B, 4B, 8B, and 14B). As illustrated in Figure 5, CCoR demonstrates a clear and consistent scaling law behavior across all three QA benchmarks. Specifically, scaling the architecture from 1.7B to 14B yields substantial performance gains, improving F1 scores by 12.26% on HotpotQA and 10.53% on 2WikiMQA. These results confirm that synchronously expanding compressor and decoder capacities enables stronger reasoning evidence extraction during compression, while simultaneously bolstering the decoder's ability to reconstruct execution trajectories from compact hybrid contexts.

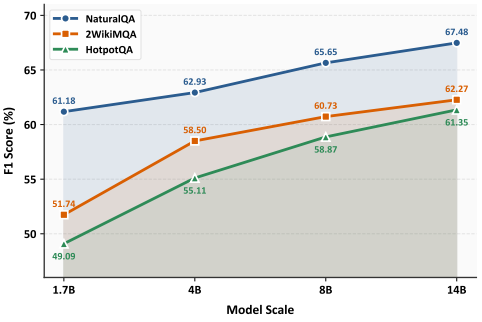

Figure 5: F1 score on three datasets across different model scales. The compressor and decoder use the same model.

7 Conclusion

In this work, we propose CCoR, a novel context compression framework that preserves essential reasoning evidence beyond superficial query-centric semantic relevance. CCoR formulates context selection as a submodular maximization problem to quantify the reasoning necessity of context chunks. It adaptively constructs a hybrid context representation by explicitly preserving high-importance chunks while compressing redundant content into latent summaries, optimized via supervised alignment and reasoning-preservation reinforcement learning. Extensive experiments across QA benchmarks show that CCoR consistently outperforms existing task-agnostic and task-aware baselines under various compression ratios. Our analysis confirms that CCoR effectively eliminates task-irrelevant noise without compromising multi-step derivation, providing a principled paradigm for context-intensive LLM reasoning.

AI USAGE STATEMENT

We used AI-based language tools solely for proofreading and polishing the manuscript, including improving grammar, clarity, and overall readability. All research ideas, methodology, experiments, analyses, and conclusions were developed and verified by the authors. The authors take full responsibility for the content of this paper.

REFERENCES

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, et al. 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)*, pp. 3119–3137, 2024.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, et al. 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)*, pp. 3639–3664, 2025.

Zhiwei Cao, Qian Cao, Yu Lu, Ningxin Peng, Luyang Huang, Shanbo Cheng, and Jinsong Su. Retaining key information under high compression ratios: Query-guided compressor for llms. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12685–12695, 2024.

Powei Chang, Jinpeng Zhang, Bowen Chen, Chenyu Wang, Chenlu Guo, Yixing Zhang, Yukang Gao, JianXiang Xiang, Yue Gao, Chaoqun Sun, et al. Spice: Submodular penalized information-conflict selection for efficient large language model training. *arXiv preprint arXiv:2601.23155*, 2026.

Ruoyu Chen, Xiaoqing Guo, Kangwei Liu, Siyuan Liang, Shiming Liu, Qunli Zhang, Laiyuan Wang, Hua Zhang, and Xiaochun Cao. Where mllms attend and what they rely on: Explaining autoregressive token generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17057–17066, 2026a.

Zhen Chen, Yibing Liu, Weihao Xie, Yu Liang, Peilin Chen, and Shiqi Wang. Raise: Rag design as an architecture search problem. *arXiv preprint arXiv:2605.30029*, 2026b.

Nadezhda Chirkova, Thibault Formal, Vassilina Nikoulina, and Stéphane Clinchant. Provence: efficient and robust context pruning for retrieval-augmented generation. 2025.

Nathan Cooper and Torsten Scholak. Perplexed: Understanding when large language models are confused. *arXiv preprint arXiv:2404.06634*, 2024.

Ziqiang Cui, Yunpeng Weng, Xing Tang, Peiyang Liu, Shiwei Li, Bowei He, Jiamin Chen, Yansen Zhang, Xiuqiang He, and Chen Ma. Less is more: Elevating rag via performance-driven context compression. *arXiv preprint arXiv:2508.19282*, 2025.

Yuhong Dai, Jianxun Lian, Yitian Huang, Wei Zhang, Mingyang Zhou, Mingqi Wu, Xing Xie, and Hao Liao. Pretraining context compressor for large language models with embedding-based memory. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28715–28732, 2025.

Ahmed El-Kishky, Alexander Wei, Andre Saraiva, Borys Minaiev, Daniel Selsam, David Dohan, Francis Song, Hunter Lightman, Ignasi Clavera, Jakub Pachocki, et al. Competitive programming with large reasoning models. *arXiv preprint arXiv:2502.06807*, 2025.

Yixiong Fang, Tianran Sun, Yuling Shi, and Xiaodong Gu. Attentionrag: Attention-guided context pruning in retrieval-augmented generation. *arXiv preprint arXiv:2503.10720*, 2025.

Tao Ge, Jing Hu, Lei Wang, Xun Wang, Si-Qing Chen, and Furu Wei. In-context autoencoder for context compression in a large language model. *arXiv preprint arXiv:2307.06945*, 2023.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

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

Ruishan Guo, Yibing Liu, Guoxin Ma, Yan Wang, Yueyang Zhang, Long Xia, Kecheng Chen, Zhiyuan Sun, and Daiting Shi. When less is more: The llm scaling paradox in context compression. 2026.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6609–6625, 2020.

Taeho Hwang, Sukmin Cho, Soyeong Jeong, Hoyun Song, Seung Yoon Han, and Jong C. Park. Exit: Context-aware extractive compression for enhancing retrieval-augmented generation. 2024.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. Longllmlingua: Accelerating and enhancing llms in long context scenarios via prompt compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1658–1677, 2024.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pp. 5156–5165. PMLR, 2020.

Andreas Krause and Daniel Golovin. Submodular function maximization. *Tractability*, 3(71-104): 3, 2014.

Yurii Kuratov, Mikhail Arkhipov, Aydar Bulatov, and Mikhail Burtsev. Cramming 1568 tokens into a single vector and back again: Exploring the limits of embedding space capacity. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19323–19339, 2025.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Chengzhengxu Li, Xiaoming Liu, Yichen Wang, Duyi Li, Yu Lan, and Chao Shen. Dialogue for prompting: A policy-gradient-based discrete prompt generation for few-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 18481–18489, 2024a.

Xinyi Li, Sai Wang, Siqi Zeng, Yu Wu, and Yi Yang. A survey on llm-based multi-agent systems: workflow, infrastructure, and challenges. *Vicinagearth*, 1(1):9, 2024b.

Huanxuan Liao, Wen Hu, Yao Xu, Shizhu He, Jun Zhao, and Kang Liu. Beyond hard and soft: Hybrid context compression for balancing local and global information retention. *arXiv preprint arXiv:2505.15774*, 2025.

Barys Liskavets, Maxim Ushakov, Shuvendu Roy, Mark Klibanov, Ali Etemad, and Shane K Luke. Prompt compression with context-aware sentence encoding for fast and improved llm inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 24595–24604, 2025.

Guoxin Ma, Xiaoming Liu, Hongyang Chen, Chengzhengxu Li, Zhaohan Zhang, Shengchao Liu, Yu Lan, Cong Wang, and Chao Shen. Deer: Disentangled mixture of experts with instance-adaptive routing for generalizable machine-generated text detection. *arXiv preprint arXiv:2511.01192*, 2025.

Guoxin Ma, Yibing Liu, Chengzhengxu Li, Yu Liang, Yan Wang, Yueyang Zhang, Kecheng Chen, Zhaohan Zhang, Zhiyuan Sun, and Daiting Shi. Thinking as compression: Your reasoning model is secretly a context compressor. *arXiv preprint arXiv:2605.28713*, 2026.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. Evaluating very long-term conversational memory of llm agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13851–13870, 2024.

George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294, 1978.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, et al. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 963–981, 2024.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. Proceedings of the International Conference on Learning Representations, 2025.

Jiwei Tang, Zhicheng Zhang, Shunlong Wu, Jingheng Ye, Lichen Bai, Zitai Wang, Tingwei Lu, Lin Hai, Yiming Zhao, Hai-Tao Zheng, et al. Gmsa: Enhancing context compression via group merging and layer semantic alignment. *arXiv preprint arXiv:2505.12215*, 2025.

Jiwei Tang, Zhijing Huang, Xinyu Zhang, Chen Jason Zhang, Jianxing Yu, Libin Zheng, Rui Meng, and Jian Yin. Beyond position bias: Shifting context compression from position-driven to semantic-driven. *arXiv preprint arXiv:2605.09463*, 2026a.

Jiwei Tang, Shilei Liu, Zhicheng Zhang, Qingsong Lv, Runsong Zhao, Tingwei Lu, Langming Liu, Haibin Chen, Yujin Yuan, Hai-Tao Zheng, et al. Read as human: Compressing context via parallelizable close reading and skimming. *arXiv preprint arXiv:2602.01840*, 2026b.

Jiwei Tang, Shilei Liu, Zhicheng Zhang, Yujin Yuan, Libin Zheng, Wenbo Su, and Bo Zheng. Comi: Coarse-to-fine context compression via marginal information gain. *arXiv preprint arXiv:2602.01719*, 2026c.

Zecheng Tang, Ji Baibei, Juntao Li, Lijun Wu, Haijia Gui, and Min Zhang. Revisiting long-context modeling from context denoising perspective. In *International Conference on Learning Representations*, volume 2026, pp. 153734–153759, 2026d.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Kai Wei, Rishabh Iyer, and Jeff Bilmes. Submodularity in data subset selection and active learning. In *International conference on machine learning*, pp. 1954–1963. PMLR, 2015.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *arXiv preprint arXiv:2310.04408*, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, and Chenxu Lv. Qwen3 technical report. 2025.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2369–2380, 2018.

Xubing Ye, Yukang Gan, Xiaoke Huang, Yixiao Ge, and Yansong Tang. Voco-llama: Towards vision compression with large language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29836–29846. IEEE, 2025.

Peitian Zhang, Zheng Liu, Shitao Xiao, Ninglu Shao, Qiwei Ye, and Zhicheng Dou. Long context compression with activation beacon. *arXiv preprint arXiv:2401.03462*, 2024.

Yunlong Zhao, Haoran Wu, and Bo Xu. Leveraging attention to effectively compress prompts for long-context llms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 26048–26056, 2025.

Appendix of CCoR

This comprehensive supplementary material provides in-depth insights into CCoR, covering preliminaries, experimental details, and additional results. The appendix is organized as follows:

CONTENTS

A Preliminaries 15
A.1 Submodular Optimization 15
A.2 Proof of the Approximation Guarantee 15
A.3 Limitations 16
B Experimental Details 17
B.1 Datasets 17
B.2 Baselines 17
B.3 Implementation Details 18
C Additional Results 18
C.1 Analysis on Extra Compression Rates 18
C.2 Analysis on Efficiency 19
C.3 Case Study 19

A PRELIMINARIES

A.1 SUBMODULAR OPTIMIZATION

In this appendix, we provide the foundational definitions and theoretical guarantees of submodular maximization, which constitute the theoretical cornerstone of the Reasoning-Aware Chunk Selection strategy employed in §5.1.

Definitions of Submodularity and Monotonicity. Let U be the ground set of all candidate context chunks, denoted as $\mathcal{C} = \{c_1, c_2, \ldots, c_U\}$ in §5.1. The goal of reasoning-aware chunk selection is to identify a subset $\mathcal{C}^* \subseteq \mathcal{C}$ that maximizes a utility function $f : 2^{\mathcal{C}} \to \mathbb{R}_{\geq 0}$ as Eq. 5 subject to a cardinality constraint $|\mathcal{C}^*| \leq k$. The effectiveness of the greedy optimization approach is predicated on two key properties of the utility function $f(S)$: submodularity and monotonicity.

Definition 1 (Submodularity). A set function $f : 2^{\mathcal{C}} \to \mathbb{R}$ is submodular if for any subsets $A \subseteq B \subset \mathcal{C}$ and an element $c \notin B$, the following diminishing returns property holds:

$$f(A \cup c) - f(A) \geq f(B \cup c) - f(B). \tag{10}$$

Here, the term $f(S \cup \{c\}) - f(S)$ represents the marginal gain of adding element c to the subset S.

Definition 2 (Monotonicity). A set function $f : 2^{\mathcal{C}} \to \mathbb{R}$ is monotonically non-decreasing (or simply monotonic) if for any subsets $A \subseteq B \subseteq \mathcal{C}$, we have:

$$f(A) \leq f(B). \tag{11}$$

Monotonicity guarantees that adding more elements to the selected subset never decreases the overall utility.

A.2 PROOF OF THE APPROXIMATION GUARANTEE

Here we provide a rigorous proof of the $(1-1/e)$ approximation bound for the greedy algorithm under a cardinality constraint $|\mathcal{C}^*| \leq k$, following the classic analysis by Nemhauser et al. Nemhauser et al. (1978). While the true objective function within CCoR derived from LLM likelihoods may not strictly satisfy standard submodularity definitions across all subsets, this framework has been empirically validated as highly effective in practical language modeling tasks Chen et al. (2026a); Wei et al. (2015); Chang et al. (2026). For the sake of this theoretical overview, we prove the bound assuming the target objective function f exhibits non-negative, monotone, and submodular properties as defined in standard discrete optimization.

Theorem 1 (Approximation Guarantee). Let $f : 2^{\mathcal{C}} \to \mathbb{R}_{\geq 0}$ be a non-negative, monotonically non-decreasing submodular function, and let $k \geq 1$. Let $\mathcal{C}_{OPT}$ be the global optimum of the cardinality-constrained maximization problem:

$$\mathcal{C}_{OPT} = \arg \max_{S \subseteq \mathcal{C}, |S| \leq k} f(S), \tag{12}$$

and let $\mathcal{C}^* = S_k$ be the solution found by the Standard Greedy Algorithm after k steps. Then, the following bound holds:

$$f(\mathcal{C}^*) \geq \left(1 - \frac{1}{e}\right) \cdot f(\mathcal{C}_{OPT}), \tag{13}$$

where e is the base of the natural logarithm, approximately 2.718.

Proof.

Let $S_t = \{c_1, c_2, \ldots, c_t\}$ be the subset of chunks selected by the greedy algorithm after t iterations, with $S_0 = \emptyset$. At each step t ($0 \leq t < k$), the greedy algorithm selects the element c_{t+1} that maximizes the marginal gain:

$$c_{t+1} = \arg \max_{c \in \mathcal{C} \setminus S_t} \left[f(S_t \cup \{c\}) - f(S_t) \right]. \tag{14}$$

To bound the remaining optimality gap at step t, we relate the current value $f(S_t)$ to the optimal value $f(\mathcal{C}_{OPT})$. By the submodularity of f (Definition 1), the marginal contribution of the optimal set $\mathcal{C}_{OPT}$ given the current set S_t satisfies:

$$f(\mathcal{C}_{OPT} \cup S_t) - f(S_t) \leq \sum_{o \in \mathcal{C}_{OPT}} \left[f(S_t \cup \{o\}) - f(S_t) \right]. \tag{15}$$

This follows by decomposing the joint marginal gain of $\mathcal{C}_{OPT}$ into a sequence of individual marginal gains and applying the diminishing returns property of submodularity.

Furthermore, by the monotonicity of f (Definition 2), adding $\mathcal{C}_{OPT}$ to S_t cannot decrease the function value, which implies:

$$f(\mathcal{C}_{OPT} \cup S_t) \geq f(\mathcal{C}_{OPT}). \tag{16}$$

Combining this with Equation (4) yields:

$$f(\mathcal{C}_{OPT}) - f(S_t) \leq f(\mathcal{C}_{OPT} \cup S_t) - f(S_t) \leq \sum_{o \in \mathcal{C}_{OPT}} \left[f(S_t \cup \{o\}) - f(S_t) \right]. \tag{17}$$

Since the greedy algorithm selects c_{t+1} to maximize the marginal gain over all available elements (which includes every individual element in $\mathcal{C}_{OPT}$), the marginal gain at step $t+1$ satisfies:

$$f(S_{t+1}) - f(S_t) = \max_{c \in \mathcal{C} \setminus S_t} \left[f(S_t \cup \{c\}) - f(S_t) \right] \geq \frac{1}{|\mathcal{C}_{OPT}|} \sum_{o \in \mathcal{C}_{OPT}} \left[f(S_t \cup \{o\}) - f(S_t) \right]. \tag{18}$$

Because f is monotone, all marginal gains are non-negative. Therefore, given that $|\mathcal{C}_{OPT}| \leq k$, we have:

$$f(S_{t+1}) - f(S_t) \geq \frac{1}{k} \left(f(\mathcal{C}_{OPT}) - f(S_t) \right). \tag{19}$$

Let $err_t = f(\mathcal{C}_{OPT}) - f(S_t)$ denote the optimality gap after t steps. Equation (7) can be rewritten in terms of err_t as:

$$err_t - err_{t+1} \geq \frac{1}{k} err_t \implies err_{t+1} \leq \left(1 - \frac{1}{k}\right) err_t. \tag{20}$$

Applying this recurrence relation inductively from $t = 0$ to k, we obtain:

$$err_k \leq \left(1 - \frac{1}{k}\right)^k err_0 \leq \left(1 - \frac{1}{k}\right)^k f(\mathcal{C}_{OPT}), \tag{21}$$

where we use $err_0 = f(\mathcal{C}_{OPT}) - f(\emptyset) \leq f(\mathcal{C}_{OPT})$ since $f(\emptyset) \geq 0$.

Using the standard inequality $1 - x \leq e^{-x}$ for all real x, we bound the term as:

$$\left(1 - \frac{1}{k}\right)^k \leq \left(e^{-1/k}\right)^k = \frac{1}{e}. \tag{22}$$

Substituting Equation (10) back into Equation (9) yields:

$$f(\mathcal{C}_{OPT}) - f(S_k) \leq \frac{1}{e} f(\mathcal{C}_{OPT}) \implies f(S_k) \geq \left(1 - \frac{1}{e}\right) f(\mathcal{C}_{OPT}). \tag{23}$$

Thus, the greedy algorithm guarantees a solution $\mathcal{C}^* = S_k$ that achieves at least a $(1 - 1/e)$ fraction of the optimal value, completing the proof. ■

A.3 LIMITATIONS

While CCoR demonstrates superior performance in preserving reasoning-critical evidence under strict compression constraints, it presents certain limitations that warrant future investigation. Primarily, our framework currently segments the long context into uniform text chunks with a fixed length. This rigid segmentation strategy might inadvertently fracture cohesive logical evidence or cross-sentence dependencies across chunk boundaries. Future work could explore dynamic boundary detection or semantic-level segmentation to preserve structural integrity more naturally. Furthermore, due to computational resource constraints, our current experiments are limited to models with up to 14B parameters, leaving the evaluation of CCoR on larger-scale LLMs (e.g., the 32B and 72B series) unexplored. While our results are promising, evaluating our framework on larger-scale models remains an important next step. This would offer a more comprehensive validation of our method's scalability and effectiveness in real-world scenarios.

B EXPERIMENTAL DETAILS

B.1 DATASETS

NaturalQuestions. NaturalQuestions Kwiatkowski et al. (2019) is a large-scale open-domain question answering dataset built from real Google search queries and Wikipedia articles. Its questions and answers reflect authentic user information needs rather than manually authored prompts. Each instance provides both short-answer spans and long-answer passages, supporting evaluation of precise answer extraction and long-context comprehension. In our setting, each example contains 20 documents, including one ground-truth document and 19 distractors.

2WikiMQA. 2WikiMQA Ho et al. (2020) is a multi-hop question answering dataset constructed from Wikipedia. It requires models to combine evidence from multiple documents and capture the relations between facts to answer complex questions. Compared with HotpotQA, 2WikiMQA emphasizes more structured reasoning patterns, such as comparison, bridging, and inference over entity relations.

HotpotQA. HotpotQA Yang et al. (2018) is a multi-hop question answering benchmark built on Wikipedia. Each question requires evidence from multiple articles, encouraging models to identify relevant entities, connect supporting facts across documents, and produce answers through cross-document reasoning rather than single-span extraction.

Table 4 summarizes the context length statistics of the test sets used in our experiments, comprising 22,576 examples in total. Their long and information-dense contexts provide a suitable setting for evaluating context compression methods.

Table 4: Statistics of context token lengths across datasets.

Dataset	# Samples	Avg. Tokens	Min. Tokens	Max. Tokens
NaturalQuestions	2,655	2,933	2,437	4,340
2WikiMultihopQA	12,576	1,018	201	4,367
HotpotQA	7,345	1,402	430	3,567
Total	22,576	1,784	–	–

B.2 BASELINES

We compare CCoR with representative task-agnostic and task-aware context compression methods.

Task-agnostic methods. **LLMLingua-2** (Pan et al., 2024) formulates context compression as token-level classification, using a bidirectional encoder to predict whether each token should be preserved or discarded. **ICAE** (Ge et al., 2023) encodes long contexts into a small number of dense memory slots that directly replace the original text as input to the downstream LLM. **Activation Beacon** (Zhang et al., 2024) progressively compresses context chunks into the layer-wise activations of inserted beacon tokens, retaining only the compressed KV representations for subsequent processing. **PCC** (Dai et al., 2025) employs a pretrained lightweight compressor to encode context segments into dense memory embeddings, which are projected into the representation space of the downstream LLM.

Task-aware methods. **LongLLMLingua** (Jiang et al., 2024) performs question-aware coarse-to-fine compression by first allocating compression budgets across documents and then pruning tokens according to their query-conditioned importance. **EXIT** (Hwang et al., 2024) performs context-aware extractive compression by jointly encoding retrieved context and selecting query-relevant sentences while preserving their contextual dependencies. **COMI** (Tang et al., 2026c) performs coarse-to-fine compression using marginal information gain, dynamically allocating compression budgets across groups and merging tokens according to query relevance and semantic redundancy. **RAM** (Tang et al., 2026b) adaptively retains highly query-relevant segments as explicit text while compressing less relevant segments into query-guided summary vectors.

B.3 Implementation Details

Our main experiments are conducted with LLaMA-3.1-8B-Instruct and Qwen3-8B as backbone models. We keep the backbone parameters frozen and optimize only the compressor LoRA with $r = 64$ and $\alpha = 128$, the decoder LoRA with $r = 32$ and $\alpha = 64$, together with the scoring head and alignment layer. Training instances are constructed from HotpotQA, 2WikiMultihopQA, and NaturalQuestions. The necessity and sufficiency terms are equally weighted with $\lambda = 0.5$, yielding approximately 40K training examples. We first perform SFT for 3 epochs using AdamW with a learning rate of 1×10^{-5}. We then conduct GRPO for 1 epoch while freezing the routing module to maintain the same compression budget as in SFT. For each prompt, GRPO samples $G = 8$ responses with a temperature of 1.0 and a maximum generation length of 2,048 tokens. We use group-normalized advantages, a clipping range of 0.2, and a KL coefficient of 0.01. The reward is defined as $\max(\mathrm{EM}, \mathrm{F1})$ for responses satisfying the required output format, and zero otherwise. The reasoning similarity reward is weighted by $\beta = 0.1$. GRPO uses a constant learning rate of 1×10^{-5}. All experiments are conducted on 8 NVIDIA A800 GPUs.

C Additional Results

C.1 Analysis on Extra Compression Rates

Following the same experimental setting as §6.4, we additionally report the Exact Match (EM) results on NaturalQuestions across compression ratios from $4\times$ to $32\times$. As shown in Figure 6, CCoR consistently maintains the strongest performance across different compression budgets, demonstrating substantial improvements over all baselines. For instance, at the $4\times$ and $8\times$ ratios, CCoR achieves EM scores of 60.7% and 59.1%, respectively, outperforming the strongest baseline (RAM) by 9.6 and 10.1 absolute percentage points. The performance gap becomes particularly critical under more aggressive compression ratios. At the extreme $32\times$ compression ratio, where conventional methods like COMI experience severe performance degradation (plummeting to 6.0% EM), CCoR sustains a robust EM score of 46.0%. This establishes a clear 6.2 percentage point advantage over the second-best method, RAM (39.8%), and significantly exceeds LongLLMLingua (28.2%). This indicates that the advantage of CCoR is not limited to a specific compression ratio, but robustly generalizes to a broad range of context budgets. We attribute this robustness to CCoR's reasoning-aware compression, which prioritizes evidence by its contribution to downstream reasoning rather than surface-level relevance. As the compression budget becomes tighter, conventional methods are more likely to discard intermediate evidence required for reasoning. In contrast, CCoR preserves reasoning-critical chunks while compressing less critical content into latent representations, enabling more stable reasoning under aggressive compression. These results further validate the robustness of CCoR across different compression ratios.

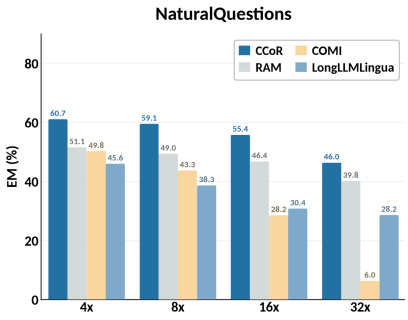

Figure 6: Additional results on NaturalQA under different compression ratios.

Table 5: Efficiency comparison under different compression constraints. Compression and prefill latency are reported in seconds (s).

Method	4× Compression Constraint			8× Compression Constraint		
	EM	Compress (s)	Prefill (s)	EM	Compress (s)	Prefill (s)
LongLLMLingua	45.61	1.27	0.10	38.27	0.94	0.06
RAM	51.15	0.14	0.13	48.96	0.14	0.07
COMI	49.83	0.14	0.12	43.31	0.14	0.06
CCoR SFT	57.14	0.09	0.11	55.93	0.09	0.06
CCoR RL	60.68	0.20	0.11	59.10	0.20	0.06

C.2 Analysis on Efficiency

As formally defined in §2, the primary goal of context compression is to mitigate input-side redundancy and alleviate prefill processing overhead, rather than focusing on output decoding latency. Accordingly, we evaluate the efficiency–effectiveness trade-off by strictly measuring input-side computational costs, reporting the compression and prefill latencies on NaturalQuestions under $4\times$ and $8\times$ constraints.

As shown in Table 5, CCoR achieves substantially higher Exact Match (EM) scores while maintaining latencies comparable to, or better than, efficient baselines. For instance, under the $4\times$ constraint, CCoR SFT achieves a 57.14% EM with merely 0.09s of compression time and 0.11s of prefill latency. This operates significantly faster than LongLLMLingua (1.27s compress) and edges out RAM (0.14s compress), all while delivering vastly superior downstream accuracy. CCoR-RL achieves the strongest performance overall (60.68% EM at $4\times$ and 59.10% EM at $8\times$) while keeping prefill latency identical to CCoR SFT (0.11s and 0.06s, respectively). It introduces a moderate additional compression cost (0.20s) due to its dual-LoRA design. This design is adopted to preserve representation stability during reinforcement learning: the SFT-adapted representation space is retained for reliable importance estimation, while a separate LoRA is optimized to improve latent compression under the reasoning-preservation reward. This decoupling mitigates representation drift during RL Li et al. (2024a); Ma et al. (2025) and enables the compressed soft tokens to complement potentially imperfect explicit chunk selection, yielding maximum reasoning preservation with only marginal additional input-side latency.

C.3 Case Study

To qualitatively illustrate the limitation of query-centric semantic relevance and the advantage of our reasoning-aware mechanism, we present representative cases from three benchmarks: 2WikiMultihopQA, HotpotQA, and NaturalQuestions. In these examples, we compare the top selected chunks from a semantic-relevance baseline (RAM) against those from our CCoR framework. To clearly distinguish different types of text matching, we use **Orange** to mark **Surface Match** (lexically query-aligned expressions) and **Green** to mark **Reasoning Evidence** (the actual information required to derive the correct answer).

From these cases, we discover that chunks closely matching the query at the surface level are often not the evidence needed for downstream multi-step reasoning. In Case 1 (2WikiMultihopQA), which requires comparing the ages of two entities, RAM is distracted by the high lexical overlap of the name "**Johann Heinrich**" and retrieves entirely irrelevant individuals. In contrast, CCoR precisely identifies the birth dates of both target individuals (**April 18, 1945** and **April 29, 1758**), providing the necessary logical bridge for the comparison. Similarly, in Case 2 (HotpotQA), RAM retrieves chunks with dense keyword occurrences (e.g., "**German**", "**film**") but completely misses the crucial demographic information. Conversely, CCoR accurately extracts the specific biographical sentences confirming Kurt Gerron's "**German Jewish**" descent.

Overall, these findings illustrate that semantic similarity primarily captures what appears lexically relevant to the query, which can easily mislead the generation model. Effective context compression should therefore transcend surface-level alignment and prioritize preserving the fundamental evidence according to its actual contribution to the downstream reasoning path.

Case 1 | 2WikiMultihopQA

Query: Who is younger, Richard Bausch or Johann Heinrich Lips?

Chain-of-Thought: Step 1: Richard Bausch was born on April 18, 1945. Step 2: Johann Heinrich Lips was born on April 29, 1758.

Gold Answer: Richard Bausch

RAM: Semantic-Relevance Selection

Top-1 Chunk. . . . Johann Heinrich Hübschmann (1 July 1848–20 January 1908) was a German philologist. Johann Heinrich Schönfeld Johann Heinrich Schönfeld (1609–. . .

Top-2 Chunk. . . . Dutch chemist. Johann Heinrich Troll Johann Heinrich Troll (1756–1824) was a Swiss painter. Miloš Zličić Miloš Zličić (born 29 December 1999) is a Serbian . . .

Prediction: Johann Heinrich Lips ×

CCoR: Reasoning-aware Selection

Top-1 Chunk. . . . Richard Bausch (born April 18, 1945) is an American novelist and short story writer, and Professor in the Writing Program at Chapman University . . .

Top-2 Chunk. . . . Johann Heinrich Lips (29 April 1758, Kloten – 5 May 1817, Zürich) was a Swiss . . .

Prediction: Richard Bausch ✓

Case 2 | HotpotQA

Query: Which actor and film director is from German Jewish descent, Kurt Gerron or Dell Henderson?

Chain-of-Thought: Step 1: Kurt Gerron was a German Jewish actor and film director. Step 2: Dell Henderson was a Canadian-American actor, director, and writer, with no mention of German Jewish descent.

Gold Answer: Kurt Gerron

RAM: Semantic-Relevance Selection

Top-1 Chunk. The White Demon (German: Der weiße Dämon) is a 1932 German drama film directed by Kurt Gerron and starring Hans Albers, Gerda Maurus and Peter Lorre.

Top-2 Chunk. A Crazy Idea (German: Ein toller Einfall) is a 1932 German comedy film directed by Kurt Gerron and starring Willy Fritsch, Dorothea Wieck and Rosy Barsony. . . .

Top-3 Chunk. . . . is a 1929 German silent mountain film co-directed by Arnold Fanck and Georg Wilhelm Pabst and starring Leni Riefenstahl, . . .

Prediction: Dell Henderson ×

CCoR: Reasoning-aware Selection

Top-1 Chunk. Kurt Gerron (11 May 1897 – 28 October 1944) was a German Jewish actor and film director. My Wife, the Impostor . . .

Top-2 Chunk. . . . Dell Henderson (July 5, 1877 – December 2, 1956) was a Canadian-American actor, director, and writer. He began his long and prolific film career in the early days of silent film. . . .

Prediction: Kurt Gerron ✓

Case 3 | NaturalQuestions

Query: Who turned out to be the mother on How I Met Your Mother?

Chain-of-Thought: Step 1: The character known as "The Mother" in How I Met Your Mother is Tracy McConnell. Step 2: Tracy McConnell is played by Cristin Milioti.

Gold Answer: Tracy McConnell

RAM: Semantic-Relevance Selection

Top-1 Chunk. 9. The Mother is played by Cristin Milioti. How I Met Your Mother by their friendship when they both lived in New York City. . . .

Prediction: Cristin Milioti ×

CCoR: Reasoning-aware Selection

Top-1 Chunk. The Mother (How I Met Your Mother). Tracy McConnell, better known as "The Mother", is the title character from the CBS television sitcom How I Met Your Mother. . . .

Prediction: Tracy McConnell ✓