Decoupled Reasoning with Implicit Fact Tokens (DRIFT): A Dual-Model Framework for Efficient Long-Context Inference

Wenxuan Xie[1,2,3], **Xuhong Wang**[3*]
[1]Fudan University, Shanghai, China
[2]Shanghai Innovation Institute, Shanghai, China
[3]Shanghai Artificial Intelligence Laboratory, Shanghai, China
wxxie25@m.fudan.edu.cn, wangxuhong@pjlab.org.cn

Abstract

The integration of extensive, dynamic knowledge into Large Language Models (LLMs) remains a significant challenge due to the inherent entanglement of factual data and reasoning patterns. Existing solutions, ranging from non-parametric Retrieval-Augmented Generation (RAG) to parametric knowledge editing, are often constrained in practice by finite context windows, retriever noise, or the risk of catastrophic forgetting. In this paper, we propose **DRIFT**, a novel dual-model architecture designed to explicitly decouple knowledge extraction from the reasoning process. Unlike static prompt compression, DRIFT employs a lightweight knowledge model to dynamically compress document chunks into implicit fact tokens conditioned on the query. These dense representations are projected into the reasoning model's embedding space, replacing raw, redundant text while maintaining inference accuracy. Extensive experiments show that DRIFT significantly improves performance on **long-context tasks**, outperforming strong baselines among comparably sized models. Our approach provides a scalable and efficient paradigm for extending the effective context window and reasoning capabilities of LLMs. Our code is available at https://github.com/Lancelot-Xie/DRIFT.

1 Introduction

The applicability of Large Language Models (LLMs) to knowledge-intensive tasks is limited by the static nature of their pre-training data. To address this limitation, prior work has explored two complementary strategies. The first focuses on augmenting the input context, while the second emphasizes knowledge parameter editing.

Traditional input augmentation via RAG or long-context prompting is increasingly constrained by the "retriever's ceiling" and the quadratic computational costs of processing long sequences. Neural compression methods (e.g., COCOM (Rau et al., 2025), C3 (Liu and Qiu, 2025)) attempt to mitigate this by distilling text into latent representations; however, as they primarily focus on static compression, task-critical information relevant to the query is frequently lost. Conversely, internalizing knowledge through direct parametric updates, such as fine-tuning or knowledge editing, often disrupts the inherent coupling between a model's internal knowledge and its reasoning logic, while also risking catastrophic forgetting. While modular approaches like MLP Memory (Wei et al., 2025) offer a plug-and-play alternative, they remain bound to pre-indexed resources and struggle to handle instantaneous, unseen long-context inputs in real-time.

To address these challenges, we introduce **DRIFT**, a dual-model architecture that decouples context processing from core reasoning. In this framework, a lightweight **knowledge model** extracts query-relevant information from document chunks and compresses it into high-density **implicit fact tokens** within a latent space. These tokens serve as concise, knowledge-rich representations that are projected into a larger **reasoning model's** embedding space. The reasoning model can perform sophisticated inference efficiently based on this compact factual context instead of raw text, even in long-context or knowledge-intensive scenarios. By delegating the processing of redundant background knowledge to the knowledge module, this design allows the reasoning model to remain unburdened by raw context, focusing instead on "clean" and deep inference through the distilled factual representations.

Our main contributions are as follows:

- **A Decoupled Inference Paradigm for Large Language Models:** We propose a dual-model

[*]Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 38047–38067

framework that decouples knowledge extraction from reasoning. A lightweight knowledge model encodes query-relevant information into compact fact tokens, which are consumed by a larger reasoning model for inference. Compared with directly processing long contexts, DRIFT improves performance while substantially reducing inference latency, achieving **an average 7× speedup on 256k-token documents**.

- **Expanding Effective Context Window with High-Ratio Compression:** By encoding extensive textual knowledge into compact fact tokens, our framework significantly extends the model's usable context window. Specifically, our **DRIFT** model (based on Mistral 7B) achieves a **32× compression ratio** while **improving accuracy from 20.87% to 29.22%** on the LongBench v2 benchmark, demonstrating superior reasoning capabilities with substantially reduced time overhead. Furthermore, even under more aggressive compression settings (**64×** and **128×**), DRIFT remains highly competitive, indicating strong robustness to extreme compression ratios.
- **Comprehensive Empirical Analysis and Resource Contribution:** We construct a large-scale Document–QA–Evidence dataset with fine-grained supervision, comprising over **300K** instances and documents ranging from **1K** to **8K** tokens. We further conduct extensive ablation studies and quantitative evaluations, rigorously validating the framework and demonstrating its robustness.

2 Related Work

2.1 Prompt Compression

Directly feeding new knowledge as context into Large Language Models (LLMs) is constrained by limited context windows and incurs significant overhead in both memory and computation. To mitigate these issues, various prompt compression methods have been proposed, which generally fall into two paradigms: *Hard Compression* and *Soft Compression*.

Hard Compression (Token Selection). Hard compression methods, also known as token pruning or selection, aim to reduce input length by discarding tokens deemed less informative. Early approaches like Selective Context (Li et al., 2023b) utilize self-information or perplexity metrics to filter out redundancy. More advanced frameworks, such as LongLLMLingua (Jiang et al., 2024) and LLMLingua-2 (Pan et al., 2024), perform prompt compression via task-aware coarse-to-fine filtering and distillation-based token selection, respectively. Task-aware token selection has also been optimized using reinforcement learning (Shandilya et al., 2025). Despite these advancements, hard compression approaches inherently limit the model's reasoning potential. They **irreversibly discard information** and rely on rigid, locally made retention decisions that often fail to preserve the **global semantic structure** required for reliable complex reasoning.

Soft Compression (Latent Representation). Early soft compression approaches, such as AutoCompressor (Chevalier et al., 2023), Gist Tokens (Mu et al., 2023), and ICAE (Ge et al., 2024), integrate compression and reasoning within a single language model. While effective for moderate context reduction, this tightly coupled design makes it difficult for a single model to simultaneously excel at both high-fidelity compression and complex reasoning, particularly under extreme compression ratios. To address this limitation, subsequent work explores decoupling compression from reasoning. Methods such as xRAG (Cheng et al., 2024) and COCOM (Rau et al., 2025) build upon the Retrieval-Augmented Generation (RAG) paradigm by compressing retrieved documents into compact latent representations before passing them to the language model. Although effective in reducing input length, these approaches remain fundamentally constrained by the retrieval stage and inherit the upper-bound limitations of RAG systems. Beyond RAG-based compression, E2LLM (Liao et al., 2025) employs a lightweight encoder to compress long inputs into latent representations for downstream LLM reasoning, though the model weights are not publicly released, limiting reproducibility and practical adoption. Context Cascade Compression (C3) (Liu and Qiu, 2025) demonstrates strong compression fidelity but lacks task-specific adaptation for downstream reasoning. As a result, a gap remains between compressed representation understanding and effective reasoning.

However, most existing soft compression methods operate in a **static** and query-agnostic manner, producing generic compressed representations that often fail to preserve task-critical information under high compression ratios. In contrast, **DRIFT** adopts a **dynamic, query-conditioned compression strategy** that selectively encodes relevant information, enabling effective reasoning even under **extreme compression ratios** in knowledge-intensive scenarios.

2.2 Learned Memory

Another line of work introduces learned parametric memory modules, such as Memory Decoder (Cao et al., 2025) and MLP Memory (Wei et al., 2025), which store knowledge in trainable parameters and are often pretrained to emulate retrieval behavior. These methods reduce inference latency without modifying the underlying model parameters, thereby preserving general reasoning capabilities. However, these architectures are inherently static-resource bound: they rely on pre-indexing or offline training on fixed knowledge bases, rendering them incapable of handling instantaneous, long-context inputs in real-time. Moreover, as these modules are often pre-trained to emulate or compress retriever behaviors, their inference dynamics remain tethered to the limitations of the original retrieval paradigm.

2.3 Functional Decoupling in Large Language Models

Recent work has explored functional decomposition in language models, including fast/slow role separation (Qi et al., 2024), disentangling memory retrieval from reasoning during inference (Jin et al., 2025), and introducing dedicated memory modules that separate static storage from the Transformer backbone's dynamic computation (Cheng et al., 2026). In contrast, DRIFT focuses on decoupling knowledge compression from reasoning for query-conditioned long-context inference.

3 Methodology

Our proposed strategy encourages the knowledge model to perform information-adaptive compression, rather than static compression. This compels the model to first identify and abstract core informational content before encoding it into a compressed semantic representation. As a result, the model learns to prioritize semantically meaningful content over superficial token patterns, leading to improved generalization and robustness in downstream tasks.

3.1 Bucketed Compression: Beyond Fixed-Ratio Compression

Most existing context compression methods adopt a fixed-ratio strategy, where the number of compressed tokens is strictly proportional to the input length (e.g., compressing 128 input tokens into 16 output tokens for an 8:1 ratio). However, such a design implicitly assumes that informative content is evenly distributed across the input, which rarely holds in real-world tasks. The amount of query-relevant information within a given context is always unknown and varies in token length. Certain tokens (such as numbers, named entities, main verbs, and constraint-related words) carry a large amount of task-relevant information, whereas redundant modifiers and generic, content-free sentences contribute little to understanding the context. For example, in document-grounded QA or long-form reasoning, a few critical sentences may carry the majority of the answer-relevant information.

This fixed-ratio compression can thus become fragile in scenarios where the input contains sparse but crucial evidence. Moreover, training a model to uniformly compress variable-length sequences may encourage shortcut learning (e.g., positional bias, over-averaging), hindering semantic abstraction. To address this, we propose a Bucketed Compression strategy that shifts from ratio-based compression to range-based compression. Instead of computing the number of output tokens as a fixed fraction of the input, we predefine token-length buckets (e.g., 64–128, 128–256), and map each input in a bucket to a fixed-size output based on the upper bound of that range.

To make the distinction clear, we present a direct comparison of the two strategies in formulaic form, under the assumption of a compression ratio c:

$$\xi_{\text{uniform}} = \left\lceil \frac{n}{c} \right\rceil, \qquad \xi_{\text{bucket}} = \left\lceil \frac{b(n)}{c} \right\rceil$$

where $b(n)$ denotes the upper bound of the bucket containing n tokens.

3.2 DRIFT: Decoupled Reasoning with Implicit Fact Tokens

The core idea of DRIFT is to modularize and explicitly separate knowledge reading and reasoning. Specifically, a small-scale knowledge model (ψ_{kno}) is responsible for reading long documents and compressing them into query-relevant information, while a large-scale reasoning model θ_{rea} focuses on utilizing this compressed knowledge to perform complex reasoning and generate answers. The interaction between the two models is realized in the latent space, which reduces redundancy and mitigates the risk of irrelevant noise interfering with the reasoning process.

To facilitate a clear understanding of our method, the overall workflow of DRIFT is depicted in Figure 1. We first define a document as $X = (x_1, \ldots, x_n)$, where n is the number of tokens in the document. For long input contexts, a systematic document chunking strategy was employed within the DRIFT framework. We utilized the `RecursiveCharacterTextSplitter` from the LangChain framework (LangChain, 2025) for this purpose. This recursive approach was adopted to ensure optimal semantic coherence by prioritizing natural delimiters (e.g., paragraphs and sentences).

$$X \xrightarrow{\text{Split}} C = (C_1, C_2, \ldots, C_K). \tag{1}$$

Given a query Q, we append a fixed number of `<|CPS|>` tokens to each chunk C_j and process them in parallel with the knowledge model ψ_{kno}, using the last-layer hidden states of these compression tokens as the latent representation T_j.

$$\psi_{\text{kno}} : (C_j, Q) \ \rightarrow \ T_j \in \mathbb{R}^{\xi_j \times d}. \tag{2}$$

Finally, the outputs from all chunks are concatenated in the original order to yield the global sequence of implicit fact tokens, denoted T.

$$T = \text{Concat}(T_1, \ldots, T_K) = [t_1, \ldots, t_\xi] \in \mathbb{R}^{\xi \times d}, \tag{3}$$

where $\xi = \sum_{j=1}^{K} \xi_j \ll N$.

Implementation Note: During concatenation, a double newline separator (\n\n) is inserted between adjacent sub-sequences T_j and T_{j+1} to mark chunk boundaries. We define $T_i \in \mathbb{R}^d$ as implicit fact tokens, and d denotes the hidden state dimensionality. These tokens encapsulate essential information from the input and serve as continuous latent units for the reasoning model.

In the next step, a three-layer MLP projector π maps the implicit fact tokens into implicit fact embeddings $E = (e_1, e_2, \ldots, e_\xi)$, thereby aligning them with the embedding space of the reasoning model. These embeddings, together with the query embeddings $E(Q)$, are subsequently fed into the reasoning model for downstream inference.

$$\theta_{\text{rea}} : \text{Concat}(E, E(Q)) \ \rightarrow \ \text{Response} \tag{4}$$

The Response denotes the thoughts and the final answer generated by the reasoning model.

This approach not only substantially reduces GPU memory consumption but also frees the reasoning model from processing lengthy documents filled with irrelevant information.

3.3 Task Definition

To more effectively achieve the decoupling of knowledge and reasoning, we decompose the training of DRIFT into three distinct stages, each optimized for a different objective, as shown in Figure 2.

3.3.1 Latent Fact Reconstruction Pretraining (LFRP)

We redefine the pretraining objective such that the reasoning model is used only as a frozen decoder to provide a reconstruction signal, while the knowledge model is optimized to generate latent factual representations that best support document reconstruction.

For the knowledge model, this compression is static because it is query-independent. We set the static compression ratio c_{sta} to 8. After applying the bucketed compression strategy, we obtain ξ_{sta}, denoting the number of implicit fact tokens.

Given a document x consisting of n tokens, $X = (x_1, x_2, \ldots, x_n)$, the knowledge model produces latent fact tokens conditioned on a static compression instruction I_{sta}:

$$\begin{aligned} T_{\text{sta}} &= [t_1, t_2, \ldots, t_{\xi_{\text{sta}}}] \\ &= \psi_{\text{kno}}(I_{\text{sta}}, x_1, x_2, \ldots, x_n). \end{aligned} \tag{5}$$

These implicit fact tokens are projected into fact embeddings via a projector module π:

$$E_{\text{sta}} = \pi(T_{\text{sta}}) = (e_1, e_2, \ldots, e_{\xi_{\text{sta}}}). \tag{6}$$

The reasoning model, parameterized by θ_{rea}, remains frozen during this pretraining stage. It receives a reconstruction instruction I_{rec} and predicts each token conditioned on the fact embeddings and previously generated tokens:

$$\mathcal{L}(\psi_{\text{kno}}, \pi) = -\sum_{x_t \in X} \log P_{\theta_{\text{rea}}}(x_t \mid I_{\text{rec}}, E, x_{<t}), \tag{7}$$

Thus, although the loss is computed using the frozen reasoning model, gradients are only backpropagated through E_{sta} into the projector π and

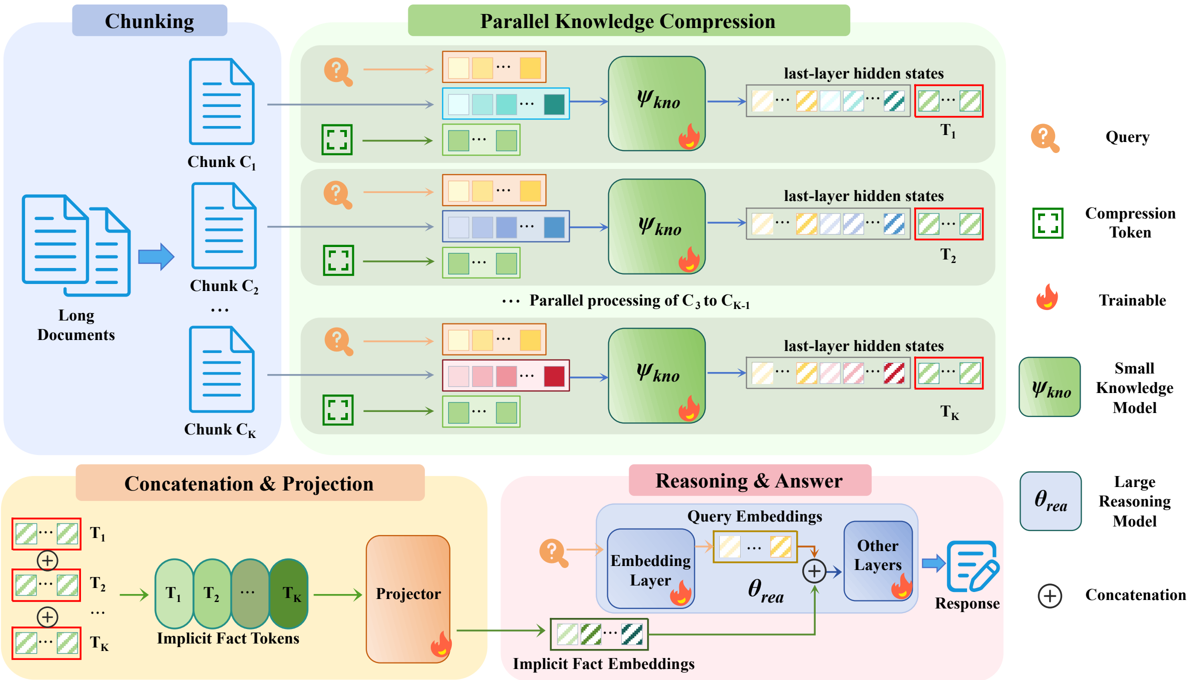

Figure 1: **The overall workflow of DRIFT.** DRIFT implements knowledge compression and decoupled reasoning in four steps. **Step 1:** The long document X is recursively partitioned into semantically coherent chunks to preserve structural integrity. **Step 2:** The small knowledge model ψ_{kno} compresses query-relevant information from each chunk in parallel into latent implicit fact tokens T_J. **Step 3:** The latent tokens are concatenated and mapped by an MLP projector π to align with the reasoning model's embedding space. **Step 4:** The large reasoning model θ_{rea} generates the final response by performing efficient inference on the concatenated embeddings.

the knowledge model ψ_{kno}. This teaches the knowledge model to produce latent factual representations that are maximally useful for reconstructing the original document.

3.3.2 Query-Aware Fine-Tuning (QAFT) with Single-Context

Through the pretraining objective described above, the knowledge model acquires the ability to encode factual knowledge into a latent space, while the reasoning model learns to understand the implicit fact embeddings. To further adapt the framework for downstream tasks, we introduce an additional training objective that incorporates query inputs. This objective is designed to encourage the knowledge model to implicitly extract and compress query-relevant knowledge into the latent space, and to enable the reasoning model to perform question answering by leveraging the information encoded in the implicit fact embeddings. We train the models on QA datasets, including reading comprehension, fact verification and open-domain question answering.

To better train toward this objective, we perform two fine-tuning tasks in sequence: first a dynamic compression task, followed by a question-answering task.

Dynamic Compression Task In the pretraining task, the knowledge model learns to perform static compression of the context. Now, we aim to train the model to acquire query-aware dynamic compression capability.

In training dataset, each question-answer pair is annotated with supporting evidence. We design a dynamic compression task using the context and question-evidence pairs: a knowledge model extracts query-specific information from the context into a latent space (implicit fact tokens), while a reasoning model reconstructs the evidence from these tokens. By leveraging the sparsity of query-relevant information within the context, dynamic compression can safely achieve a significantly higher compression ratio than static approaches. The default dynamic compression ratio is set to 32.

We use the instruction I_{dyn} to ask the knowledge model to extract question-relevant knowledge from the document and encodes it into a set of implicit fact tokens in the latent space. Conditioned on the input query Q, the knowledge model ψ_{kno} generates a sequence of question-aware implicit fact tokens T_{dyn}.

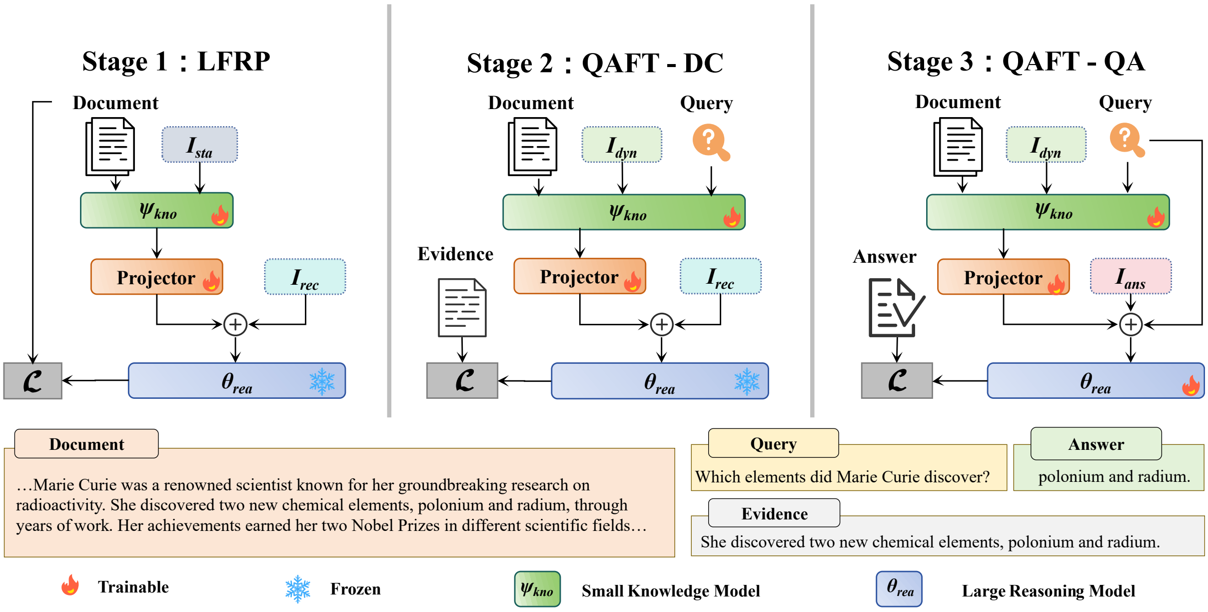

Figure 2: Three different trainging tasks for DRIFT. The instructions in the figure include the dynamic compression instruction, reconstruct instruction, answer instruction, and static compression instruction.

The reconstruction performed by the reasoning model under instruction I_{rec} remains analogous to the pre-training stage. However, we diverge by restricting the reconstruction target to the evidence instead of the complete original text. This mechanism employs the evidence as the supervisory signal, explicitly training the knowledge model to develop dynamic compression capabilities.

Given a document $X = (x_1, x_2, \ldots, x_n)$ and a question $Q = (q_1, q_2, \ldots, q_m)$, the knowledge model ψ_k performs dynamic compression on X conditioned on Q.:

$$T_{\text{dyn}} = (t_1, t_2, \ldots, t_{\xi_{\text{dyn}}}) = \psi_{\text{kno}}(I_{\text{dyn}}, X, Q) \quad (8)$$

These latent tokens are then projected into the final fact embeddings $E_{\text{dyn}} = (e_1, e_2, \ldots, e_{\xi_{\text{dyn}}})$ via the projector module π. The reasnoning model still keeps fixed and is asked to reconstruct the evidence X_{evi}:

$$\mathcal{L}(\psi_{\text{kno}}, \pi) = -\sum_{x_k \in X_{\text{evi}}} \log P_{\theta_{\text{rea}}} \left(x_k \mid I_{\text{rec}}, E_{\text{dyn}}, x_{<k}\right). \quad (9)$$

Question-answering Task This task is the only one in which the reasoning model is not frozen during training, enabling it to better exploit the compressed context for downstream tasks. We perform fine-tuning by updating the models solely based on the target answers.

The generation of E_{dyn} leverages the same mechanism as the Dynamic Compression Task, implemented by the Knowledge model and projector. Then we instruct the reasoning model with I_{ans} to generate the answer of the question based on the fact embeddings. We denote the answer sequence as $A = (a_1, a_2, \ldots, a_l)$.

We define the training objective as the standard language modeling loss. Our formulation is largely analogous to instruction tuning (Wei et al., 2022), with the key distinction that the context presented to the reasoning model is transformed from the explicit text tokens to the implicit fact embeddings produced by the knowledge model. The question embedding is represented by $E(Q)$. We minimize the following loss to optimize the model parameters:

$$\mathcal{L}(\theta_{\text{rea}}, \psi_{\text{kno}}) = -\sum_{a_j \in A} \log P_{\theta_{\text{rea}}} \left(a_j \mid I_{\text{ans}}, E_{\text{dyn}}, E(Q), a_{<j}\right). \quad (10)$$

3.4 Multi-Context Inference without Fine-Tuning

We find that DRIFT, fine-tuned solely in the single-context setting via QAFT, generalizes effectively to multi-context inference without requiring any additional multi-context training. Specifically, to process extensive contexts, we employ an overlapping chunking strategy utilizing

`RecursiveCharacterTextSplitter`. We set the chunk size to 8,192 tokens, aligning with the maximum document length used during training. These chunks are then dynamically compressed in parallel by the knowledge model.

3.5 Training Data

We use the English Wikipedia snapshot dated November 1, 2023, treating each entry as a document and its text field as the raw training content. Since longer input documents make model training and convergence more challenging, we adopt a token-level curriculum learning strategy across all three training tasks, with phases defined by input document length. Details of the construction procedures for LFRP and QAFT, as well as the curriculum learning partitions, are provided in Appendix A.

4 Experiments

4.1 Implementation Details

For the DRIFT setup, we fix `Qwen2.5-Instruct-3B` as the knowledge model and evaluate reasoning models from different families, including Mistral(Jiang et al., 2023) and Qwen2.5(Yang et al., 2024), with `Mistral-7B-Instruct-v0.2` as default. We further explore additional configurations, such as smaller knowledge models (1.5B) and larger reasoning models (14B). All models are trained using parameter-efficient fine-tuning with LoRA (Hu et al., 2022). The results for broader model combinations, along with detailed training hyperparameters, are provided in the appendix.

RQ1: To what extent do the latent representations capture and preserve the essential information of the original context?

To assess whether the knowledge model can compress long contexts with minimal information loss, we evaluate the reconstruction fidelity of the learned latent representations before downstream reasoning. Specifically, after training on the LFRP task, we conduct a compression–reconstruction experiment on a test subset with input lengths ranging from 512 to 1024 tokens, where the reasoning model reconstructs the original text conditioned solely on the compressed representations produced by the knowledge model. Reconstruction quality is measured using BLEU and ROUGE (ROUGE-1/2/L) scores.

Models		BLEU	R-1	R-2	R-L
DRIFT after LFRP	Mistral-7B	90.01	94.95	93.93	94.75
	Qwen2.5-7B	83.43	86.60	84.13	88.80
DRIFT before LFRP	Mistral-7B	0.01	1.63	0.07	1.46
	Qwen2.5-7B	0.00	6.32	0.65	4.52

Note: R-1/2/L stands for ROUGE-1/2/L scores. Scores are scaled by 100.

Table 1: Performance evaluation of the reconstruction task across different model configurations.

As illustrated in Table 1, the compression–reconstruction task achieves strong performance across all model combinations following the LFRP training phase. This indicates that after training, the knowledge model effectively compresses the input content and the reasoning model accurately interprets the compressed representations.

RQ2: To what extent can the query-conditioned compression module selectively preserve query-relevant evidence from the original context?

To further examine the compression behavior of DRIFT, we evaluate its query-conditioned compression module after training on the QAFT-DC task. The knowledge model first compresses the context conditioned on the query, and the reasoning model then reconstructs the compressed content. Reconstruction quality is measured against the ground-truth evidence annotated during data generation using BLEU and ROUGE (ROUGE-1/2/L) across different context length ranges.

Range	BLEU	R-1	R-2	R-L
1k–2k	50.30	63.50	56.23	61.44
2k–4k	52.12	63.91	57.45	60.89
4k–8k	51.74	65.64	58.49	61.75

Note: Scores are scaled by 100.

Table 2: Performance of query-conditioned compression across context length ranges with Mistral-7B.

As shown in Table 2, the model achieves consistently strong BLEU and ROUGE scores across different context length ranges. These results suggest that the latent compression process retains substantial query-relevant information that can be recovered by the inference model. In addition, noise and alternative valid spans in the AI-annotated evidence may further reduce lexical-overlap-based metrics. Therefore, these results should be viewed as supportive evidence that DRIFT preserves query-relevant content, rather than as a near-lossless reconstruction result.

RQ 3: How does DRIFT compare to representative baselines in terms of overall effectiveness in long-context reasoning scenarios?

Model	Comp. Ratio	BAMBOO (16k)				L-Eval (QA Subset)				L-Eval (Sum Subset)			LoCoMo					LongBench-v2			
		AltQA	Meet	Paper	Avg	NQ	NarQA	Crse	Avg	QMS	SPC	Avg	T1	T2	T3	T4	Avg	Short	Medium	Long	Avg
Baseline Methods based on Mistral-7B-Instruct-v0.2																					
LLMLingua-2	3×	40.00	75.00	76.77	57.89	77.98	33.18	64.53	53.94	19.37	18.86	19.14	46.81	27.41	48.96	76.93	59.35	26.11	13.95	25.00	20.68
NaiveRAG	–	34.50	79.00	75.76	55.89	72.48	21.50	63.37	47.27	19.61	18.11	18.94	45.04	24.61	41.67	73.48	56.10	17.78	26.05	25.00	22.86
xRAG	128×	25.00	68.00	57.58	43.86	56.88	12.62	50.58	35.56	18.06	12.34	15.49	20.57	6.23	34.38	32.10	24.74	31.11	25.58	25.00	27.43
ICAE	4×	21.50	19.00	19.19	20.30	36.70	5.14	15.70	15.76	17.62	13.27	15.67	17.38	3.43	23.96	15.10	13.64	13.89	14.88	10.56	13.12
COCOM	4×	30.50	60.00	47.47	42.11	61.47	9.81	53.49	36.36	9.15	7.79	8.54	9.93	1.87	27.08	10.34	9.55	27.22	23.26	34.26	27.04
	16×	25.50	49.00	41.41	35.34	53.21	8.88	55.81	34.95	7.78	5.37	6.70	11.35	1.87	25.00	10.11	9.55	27.78	23.26	34.26	27.24
	128×	15.00	40.00	36.36	26.57	40.37	5.61	39.53	25.05	8.71	4.79	6.95	12.06	1.56	30.21	8.32	8.96	29.44	21.40	34.26	27.04
DRIFT based on Qwen2.5-Instruct-3B & Mistral-7B-Instruct-v0.2																					
DRIFT (Ours)	**32×**	**41.50**	**80.00**	**77.78**	**60.15**	**69.72**	**27.10**	**61.05**	48.28	**21.59**	**19.75**	20.76	**36.88**	**23.99**	**33.33**	**80.38**	**57.73**	**32.22**	**28.84**	**25.00**	**29.22**
	64×	**36.00**	**82.00**	**82.83**	**59.15**	**70.64**	**24.77**	**58.72**	46.67	**21.95**	**19.49**	20.85	**34.75**	**16.82**	**42.71**	**74.67**	**53.31**	**26.67**	**29.77**	**19.44**	**26.44**
	128×	**36.00**	**77.00**	**78.79**	**56.89**	**71.56**	**17.29**	**60.47**	44.24	**20.69**	**19.14**	19.99	**30.14**	**17.13**	**36.46**	**72.06**	**50.71**	**32.22**	**24.65**	**19.44**	**26.24**
DRIFT based on Qwen2.5-Instruct-3B & Qwen2.5-Instruct-7B																					
DRIFT (Ours)	**32×**	**37.00**	**81.00**	**84.85**	**59.90**	**80.73**	**27.10**	**76.16**	55.96	**22.66**	**19.01**	21.02	**42.55**	**27.73**	**40.62**	**85.49**	**62.79**	**36.67**	**28.37**	**33.33**	**32.41**
Vanilla LLM																					
Mistral-7B-Instruct-v0.2	1×	40.00	75.00	76.77	57.89	79.82	31.78	65.12	53.94	19.32	18.90	19.13	46.81	27.73	48.96	76.81	59.35	25.00	16.28	23.15	20.87
Qwen2.5-Instruct-7B	1×	36.00	78.00	82.83	58.15	80.73	33.18	75.58	58.38	20.74	18.50	19.74	48.23	38.01	46.88	85.14	66.17	41.11	26.05	24.07	31.01

Table 3: Main results of DRIFT and other baseline methods on long context datasets.

Benchmarks To evaluate DRIFT across diverse long-context scenarios, we employ several representative benchmarks. **BAMBOO** (Dong et al., 2024) serves as a comprehensive suite for testing extended context capabilities through tasks like question answering and code completion. **L-Eval** (An et al., 2024) provides a balanced evaluation of single-hop and multi-hop reasoning across varying document lengths up to 256K tokens while minimizing knowledge leakage. **LongBench-v2** (Bai et al., 2025) consists of challenging multiple-choice questions requiring multi-document understanding and structured data reasoning across contexts reaching millions of words. Finally, **LoCoMo** (Maharana et al., 2024) focuses on long-term conversational memory, probing the model's ability to perform temporal reasoning and event summarization over multi-session dialogue histories.

Metrics We report two primary metrics. For datasets involving multiple-choice, closed-ended, and measurable open-ended tasks, we use `Qwen-2.5-72B-Instruct` as an **LLM-Judge** to compute accuracy. To verify reproducibility, we repeated the evaluation 10 times under the same configuration (temperature = 0), and observed identical accuracy scores across runs. More details of the LLM-as-a-Judge setup are provided in Appendix E.1. For summarization-oriented tasks, we report **ROUGE-L** scores to measure the similarity between generated responses and ground-truth references based on the Longest Common Subsequence.

Baselines We compare **DRIFT** against a diverse set of baseline methods categorized by their compression and retrieval paradigms. For **hard compression**, we select **LLMLingua-2** to represent lexical-level token pruning. For **soft compression**, we evaluate several latent-space models including **ICAE**, **COCOM**, and **xRAG**. Additionally, we implement a **NaiveRAG** baseline utilizing the **BGE-M3** embedding model for document retrieval. The **Mistral-7B-v0.2** model serves as our primary vanilla backbone to provide a performance lower bound without external enhancements.

Results and Analysis Table 3 demonstrates that DRIFT consistently outperforms existing compression-based baselines across diverse long-context benchmarks, particularly under high compression ratios. With the same Mistral backbone, DRIFT sets a new state of the art over existing compression-based methods. Notably, on task-oriented summarization benchmarks (**QMSUM** and **SPACE**) and the **LoCoMo** conversational memory benchmark—where prior compression approaches largely fail—DRIFT remains effective, highlighting its superior ability to preserve and exploit long-range contextual information.

RQ4: How does each training objective contribute to the overall performance of DRIFT?

DRIFT incorporates three distinct training objec-

Model	BAMBOO-Avg	L-Eval(QA)-Avg	L-Eval(Sum)-Avg	LoCoMo-Avg	LongBenchv2-Avg
DRIFT 3B+7B	59.90	55.96	21.02	62.79	32.41
DRIFT 1.5B+7B	59.40	53.94	21.58	59.22	33.60
DRIFT 3B+14B	60.65	56.97	21.95	62.01	34.39

Table 4: Performance of different model size combinations across multiple long-context benchmarks, reported using average scores for each benchmark.

tives: LFRP, QAFT-DC, and QAFT-QA. To justify the necessity of this multi-stage design and quantify the contribution of each component, we conduct a comprehensive ablation study across three long-context benchmarks.

Method	Bamboo	LongBenchv2	LoCoMo
DRIFT (32×)	**60.15**	**29.22**	**57.73**
w/o LFRP	$57.64^{\downarrow 2.51}$	$26.84^{\downarrow 2.38}$	$52.68^{\downarrow 5.05}$
w/o QAFT-DC	$56.64^{\downarrow 3.51}$	$25.84^{\downarrow 3.38}$	$57.36^{\downarrow 0.37}$
w/o QAFT-QA	$45.14^{\downarrow 15.01}$	$18.05^{\downarrow 11.17}$	$36.89^{\downarrow 20.84}$

Note: QAFT-DC denotes the *QAFT Dynamic Compression* objective, and QAFT-QA denotes the *QAFT Question Answering* objective.

Table 5: DRIFT ablation results (avg. accuracy).

As shown in Table 5, each training objective in DRIFT serves a unique purpose: QAFT-QA provides the fundamental reasoning backbone, while LFRP and QAFT-DC further optimize the latent space efficiency and robustness.

RQ5: Does DRIFT maintain competitive inference efficiency compared to classical baselines?

Efficiency is a critical bottleneck for long-context reasoning. We conduct an end-to-end Time-to-First-Token (TTFT) analysis to evaluate whether DRIFT maintains its performance advantages without incurring prohibitive computational costs as the input scale grows. The end-to-end Token-to-First-Token (TTFT) measures the latency from providing both the input documents and the query to the system until the first output token is generated. This metric directly captures the practical responsiveness of long-context reasoning systems under realistic inference settings.

Figure 3 shows that DRIFT maintains competitive efficiency across all baselines. Notably, the performance gap between DRIFT and the *Full-Context* baseline widens significantly as the input length increases, demonstrating DRIFT's superior scalability for ultra-long sequences.

RQ6: Does the DRIFT method remain effective across different combinations of model sizes? What would be the impact of substituting a smaller knowledge model or a larger reasoning model?

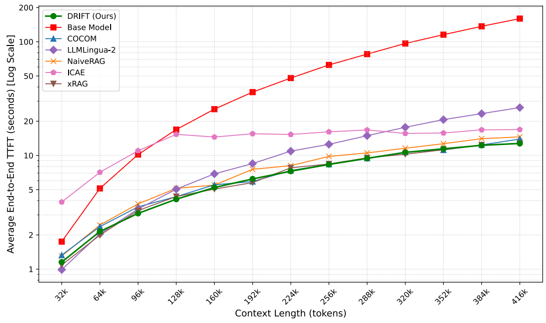

Figure 3: End-to-end TTFT as a function of input length for different baselines.

Table 4 shows that DRIFT is robust to different knowledge–reasoning model size combinations. A smaller knowledge model (1.5B vs. 3B) causes only minor and non-systematic changes, suggesting that the knowledge module mainly serves as an efficient extractor of query-relevant evidence. By contrast, scaling up the reasoning model (7B to 14B) yields consistent but moderate gains on several reasoning-intensive benchmarks. Overall, these results indicate that DRIFT's gains are architectural rather than configuration-specific: the decoupled knowledge compression–reasoning framework remains effective across different model size combinations, providing flexibility in balancing performance and computational cost.

5 Conclusion

In this paper, we introduced **DRIFT**, a dual-model architecture designed to decouple factual knowledge acquisition from general reasoning in LLMs. Input documents are segmented into chunks, compressed into high-density fact embeddings by a lightweight knowledge model, and interpreted by a larger reasoning model. Experimental results and ablation studies demonstrate the effectiveness of DRIFT and the necessity of each training task. The method also generalizes well across different compression ratios and backbone models, highlighting its robustness and practical applicability.

6 Limitations

Despite its effectiveness, our work has several limitations that suggest directions for future research. First, due to computational resource constraints, our experiments are primarily conducted on models with up to 14B parameters; the performance and scaling behavior of DRIFT on larger-scale models remain to be explored. Second, the use of latent compression introduces challenges for interpretability, as the implicit fact tokens are not as human-readable as explicit text snippets in traditional RAG systems. This creates a potential risk in that the transferred information is harder to inspect, trace, and diagnose, making it more difficult to explicitly understand what knowledge is preserved or omitted during compression. Finally, our current framework is mainly optimized through Supervised Fine-Tuning (SFT). We anticipate that incorporating Reinforcement Learning (RL) or other feedback-driven optimization strategies could further improve the model's decision-making in knowledge selection and lead to additional performance gains.

7 Ethical Considerations

This work is a methods paper and does not involve human-subject data collection or private user data. Our experiments are conducted on publicly available corpora and standard benchmarks. Nevertheless, because DRIFT compresses long contexts into latent representations, its outputs may be less interpretable than text-based retrieval systems and should therefore be used with caution in high-stakes settings. In practical deployments, appropriate data authorization, privacy protection, and human oversight remain important.

Acknowledgements

ChatGPT was used solely for language polishing during manuscript preparation, including minor paraphrasing and wording refinement. All technical content, claims, and final wording were verified by the authors.

References

Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. 2024. L-eval: Instituting standardized evaluation for long context language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14388–14411, Bangkok, Thailand. Association for Computational Linguistics.

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

Jiaqi Cao, Jiarui Wang, Rubin Wei, Qipeng Guo, Kai Chen, Bowen Zhou, and Zhouhan Lin. 2025. Memory decoder: A pretrained, plug-and-play memory for large language models. *CoRR*, abs/2508.09874.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. Evaluating large language models trained on code. *Preprint*, arXiv:2107.03374.

Xin Cheng, Xun Wang, Xingxing Zhang, Tao Ge, Si-Qing Chen, Furu Wei, Huishuai Zhang, and Dongyan Zhao. 2024. xrag: Extreme context compression for retrieval-augmented generation with one token. In *Advances in Neural Information Processing Systems*, volume 37, pages 109487–109516. Curran Associates, Inc.

Xin Cheng, Wangding Zeng, Damai Dai, Qinyu Chen, Bingxuan Wang, Zhenda Xie, Kezhao Huang, Xingkai Yu, Zhewen Hao, Yukun Li, and 1 others. 2026. Conditional memory via scalable lookup: A new axis of sparsity for large language models. *arXiv preprint arXiv:2601.07372*.

Alexis Chevalier, Alexander Wettig, Anirudh Ajith, and Danqi Chen. 2023. Adapting language models to compress contexts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3829–3846, Singapore. Association for Computational Linguistics.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Zican Dong, Tianyi Tang, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. 2024. BAMBOO: A comprehensive benchmark for evaluating long text modeling capacities of large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2086–2099, Torino, Italia. ELRA and ICCL.

Tao Ge, Hu Jing, Lei Wang, Xun Wang, Si-Qing Chen, and Furu Wei. 2024. In-context autoencoder for context compression in a large language model. In *International Conference on Learning Representations*, volume 2024, pages 2591–2607.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2024. 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)*, pages 1658–1677, Bangkok, Thailand. Association for Computational Linguistics.

Mingyu Jin, Weidi Luo, Sitao Cheng, Xinyi Wang, Wenyue Hua, Ruixiang Tang, William Yang Wang, and Yongfeng Zhang. 2025. Disentangling memory and reasoning ability in large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1681–1701, Vienna, Austria. Association for Computational Linguistics.

Yuri Kuratov, Aydar Bulatov, Petr Anokhin, Ivan Rodkin, Dmitry Igorevich Sorokin, Artyom Sorokin, and Mikhail Burtsev. 2024. BABILong: Testing the limits of LLMs with long context reasoning-in-a-haystack. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Yanlin Lai, Mitt Huang, Hangyu Guo, Xiangfeng Wang, Haodong Li, Shaoxiong Zhan, Liang Zhao, Chengyuan Yao, Yinmin Zhang, Qi Han, Chun Yuan, Zheng Ge, Xiangyu Zhang, and Daxin Jiang. 2026. R-align: Enhancing generative reward models through rationale-centric meta-judging. *CoRR*, abs/2602.06763.

LangChain. 2025. langchain-text-splitters. PyPI package. Version 0.3.8; text splitting utilities.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023a. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore. Association for Computational Linguistics.

Yucheng Li, Bo Dong, Frank Guerin, and Chenghua Lin. 2023b. Compressing context to enhance inference efficiency of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6342–6353, Singapore. Association for Computational Linguistics.

Zihan Liao, Jun Wang, Hang Yu, Lingxiao Wei, Jianguo Li, Jun Wang, and Wei Zhang. 2025. E2LLM: Encoder elongated large language models for long-context understanding and reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 19201–19230, Suzhou, China. Association for Computational Linguistics.

Fanfan Liu and Haibo Qiu. 2025. Context cascade compression: Exploring the upper limits of text compression. *arXiv preprint arXiv:2511.15244*.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. 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)*, pages 13851–13870, Bangkok, Thailand. Association for Computational Linguistics.

Jesse Mu, Xiang Li, and Noah Goodman. 2023. Learning to compress prompts with gist tokens. In *Advances in Neural Information Processing Systems*, volume 36, pages 19327–19352. Curran Associates, Inc.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Dongmei Zhang. 2024. LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 963–981, Bangkok, Thailand. Association for Computational Linguistics.

Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hannaneh Hajishirzi. 2025. Generalizing verifiable instruction following. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Biqing Qi, Xinquan Chen, Junqi Gao, Dong Li, Jianxing Liu, Ligang Wu, and Bowen Zhou. 2024. Interactive continual learning: Fast and slow thinking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12892.

David Rau, Shuai Wang, Hervé Déjean, Stéphane Clinchant, and Jaap Kamps. 2025. Context embeddings for efficient answer generation in retrieval-augmented generation. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data

Mining, WSDM '25, page 493–502, New York, NY, USA. Association for Computing Machinery.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2024. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

Shivam Shandilya, Menglin Xia, Supriyo Ghosh, Huiqiang Jiang, Jue Zhang, Qianhui Wu, Victor Rühle, and Saravan Rajmohan. 2025. TACO-RL: Task aware prompt compression optimization with reinforcement learning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 1582–1597, Vienna, Austria. Association for Computational Linguistics.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Rubin Wei, Jiaqi Cao, Jiarui Wang, Jushi Kai, Qipeng Guo, Bowen Zhou, and Zhouhan Lin. 2025. MLP memory: A retriever-pretrained memory for large language models. *CoRR*, abs/2508.01832.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024. Qwen2.5 technical report. *CoRR*, abs/2412.15115.

Jingwei Yi, Yueqi Xie, Bin Zhu, Emre Kiciman, Guangzhong Sun, Xing Xie, and Fangzhao Wu. 2025. Benchmarking and defending against indirect prompt injection attacks on large language models. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, KDD '25, page 1809–1820, New York, NY, USA. Association for Computing Machinery.

A Data Generation Details

We compute token lengths for Wikipedia documents and sample text segments for each length bucket. Specifically, for the LFRP task, we sample 200,000 segments per bucket, ensuring full coverage including the 4k8k range. For the QA-FT task, we sample 100,000 segments per bucket, with the exception of the 4k–8k range where data remains insufficient. Each stage is then split into training, validation, and test sets with an 8:1:1 ratio.

A.1 LFRP Data

For the unlabeled data used in pre-training, we deliberately refrain from any data cleaning. Since documents or knowledge bases in real-world scenarios are often noisy and heterogeneous in format, we retain the raw text to improve the robustness of the pre-trained model. For each stage, we use 160,000 samples to train the model.

A.2 QAFT Data

Based on raw Wikipedia documents corresponding to each length bucket, we employed Qwen2.5-72B-Instruct to generate QA pairs, which were subsequently used for fine-tuning in each training stage. In general, each document corresponds to one QA pair; however, for the 4k–8k stage, the available raw documents were insufficient, so some documents were reused. To avoid positional bias in the generated QA data, we segment each document and randomly sample one slice as the input for QA generation. The workflow of data generation is shown in Figure 4.

Data Sampling Strategy To ensure that relevant information is uniformly distributed across the generated dataset, we adopt the following sampling strategy. Each Wikipedia document is first divided into multiple slices of equal length. For every document, we then randomly select one slice as the context for QA generation. This prevents the model from always encountering answers concentrated in specific regions (e.g., the beginning of documents) and ensures that relevant content appears at random positions. As a result, the generated QA pairs cover diverse locations within documents, leading to a more balanced and robust training signal. In addition to generating the question–answer pairs, the model is also required to provide the supporting evidence from the original text corresponding to each answer.

Question Type Diversification To better train the reasoning model to utilize information compressed by the knowledge model, we randomly sample one of three formats—multiple-choice, true/false, or short-answer questions—for QA generation. This exposes the model to diverse reasoning scenarios, thereby strengthening its ability to integrate knowledge-derived information for effective problem solving, and enhancing its generalization across different task settings.

Data Filtering To ensure the quality of the automatically generated QA pairs, we employ Qwen2.5-72B-Instruct as the judger model to filter out low-quality instances. Specifically, the filtering process is guided by five criteria: (i) **Relevance**: the question must be grounded in the given context; (ii) **Correctness**: the provided answer should be factually consistent with the context; (iii) **Clarity**: the question and answer must be well-formed and unambiguous; (iv) **Fidelity**: the evidence must accurately reflect content from the original document without introducing external information; (v) **Sufficiency**: the evidence must provide enough information to answer the question. Only QA pairs that satisfy all conditions are retained for subsequent training.

A.3 Training Strategy: Token-Level Curriculum Learning

During model training, we observed that longer input documents lead to increased training difficulty. In particular, the convergence of both the knowledge and reasoning models becomes more challenging as the number of tokens grows. Therefore, we apply a **token-level curriculum learning strategy** across all three training tasks, where training stages are defined according to the token length of the input documents. Table 6 summarizes the token-level stage configuration for each of the three training tasks.

Task	Stage 1	Stage 2	Stage 3	Stage 4
LFRP	64-128	128-256	256-512	512-1k
QAFT Tasks				
Dynamic Comp.	1k-2k	2k-4k	4k-8k	–
Question Ans.	1k-2k	2k-4k	4k-8k	–

Table 6: Curriculum learning stages. Note that QAFT tasks conclude at Stage 3.

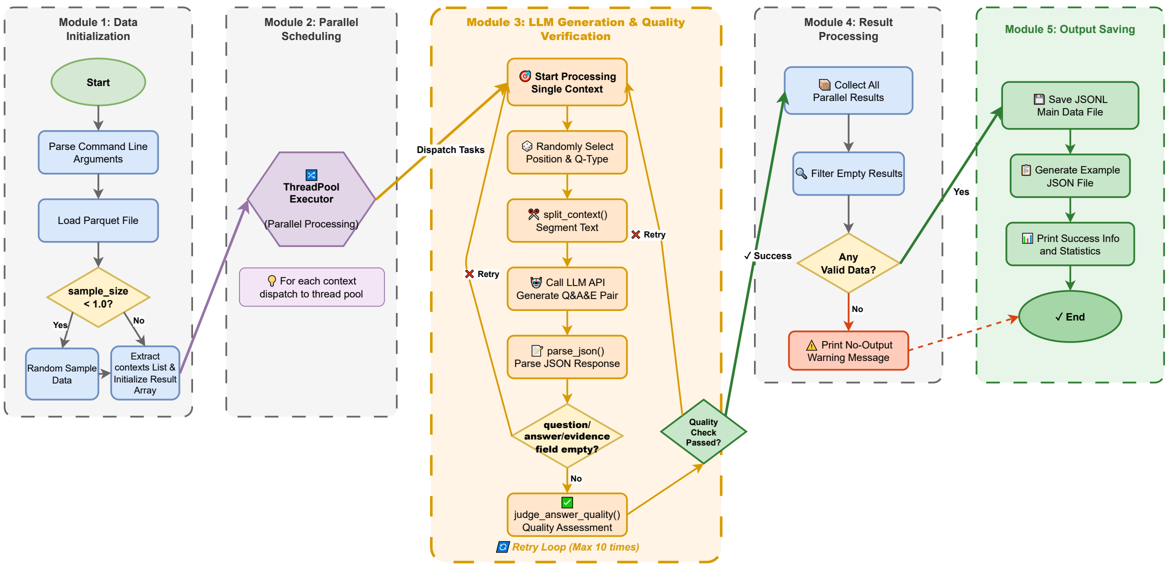

Figure 4: An Automated Pipeline for Contextual Question-Answering Data Synthesis

A.4 Construction Prompt

Two distinct prompts serve data construction: one for generation and one for fine-grained evaluation.

Prompt 1 *" Please generate a question that can be answered based on the provided context. The question should be highly relevant to the context, and the answer must be directly inferable from the given information. Avoid asking questions that cannot be answered using the context. The question should be of the type: {question_type}.*
Your response should consist of three parts:
1. Question – the generated question. (a string)
2. Answer – the answer, including how it is reasoned out from the relevant information in the context. (a string)
3. Evidence – the specific part(s) of the original text that support the answer. (a string)
Attention: Evidence must be quoted directly from the original text and must include all the information needed to answer the question. If some parts of the evidence involve unclear references (e.g., ambiguous subjects), include the related sentences that clarify them, so that the evidence alone is sufficient for answering the question. Ensure that every sentence remains complete, without the use of ellipses.
Your output format should be:
json
{{
"question": "<the generated question (include options if the question type is multiple choice)>",
"answer": "<the corresponding answer, including how it is inferred from the relevant information in the context>",
"evidence": "<the specific part(s) taken directly from the original text that support the answer>"
}}
Context: {context}
Your output: ",

Prompt 2 *"You are a judge evaluating the quality of question-answer pairs. Your task is to determine whether the given answer can be reasonably inferred from the provided evidence.*
Please evaluate based on the following criteria:
1. Can the answer be directly supported by the evidence?
2. Is the evidence sufficient to answer the question?
3. Is the answer logically consistent with the evidence?
4. Are there any contradictions between the answer and evidence?
Question: {question}
Evidence: {evidence}
Answer: {answer}
Please respond with only "true" if the answer can be reasonably inferred from the evidence, or "false" if it cannot.
Your judgment:",

B Dataset Statistics

Dataset Statistics and Configuration: Table 7 summarizes the statistics of our cleaned raw dataset. The dataset comprises approximately 2.13 million samples with a total of 1.61 billion tokens, covering a wide range of sequence lengths from 64 to 4,096. This diverse distribution ensures the model's robustness across various context windows. Our final datasets were constructed by extracting and processing samples from this original dataset. For the training pipeline, we strategically allocated the data across different phases: the LFRP stage utilized 640,000 samples to establish solid feature representations, while the QAFT stage employed 240,000 specifically constructed samples to refine the model's task-specific performance.

Table 7: Statistics of the dataset across different token length ranges.

Token Range	Split	Total Tokens	Samples
64 – 128	Train	30,511,507	320,000
	Val	3,813,711	40,000
	Test	3,813,120	40,000
	Total	**38,138,338**	**400,000**
128 – 256	Train	61,175,314	320,000
	Val	7,645,806	40,000
	Test	7,644,977	40,000
	Total	**76,466,097**	**400,000**
256 – 512	Train	122,436,827	320,000
	Val	15,301,762	40,000
	Test	15,303,893	40,000
	Total	**153,042,482**	**400,000**
512 – 1,024	Train	244,714,180	320,000
	Val	30,584,627	40,000
	Test	30,586,768	40,000
	Total	**305,885,575**	**400,000**
1,024 – 2,048	Train	464,065,287	303,724
	Val	58,032,940	37,964
	Test	58,040,040	37,968
	Total	**580,138,267**	**379,656**
2,048 – 4,096	Train	361,195,148	118,272
	Val	45,147,828	14,784
	Test	45,127,513	14,784
	Total	**451,470,489**	**147,840**
4,096 – 8,192	Train	246,548,120	40,392
	Val	30,841,105	5,048
	Test	30,832,237	5,052
	Total	**308,221,462**	**50,492**

C Training Details

C.1 Hyperparameter Settings

The specific hyperparameter configurations for our model architectures and training environment are summarized here. We conduct the training using Low-Rank Adaptation (LoRA) to ensure parameter efficiency. Specifically, we set the LoRA rank to r=16 and the scaling factor to α=32, incorporating a dropout rate of 0.05 to mitigate overfitting. The optimization is performed with a learning rate of 0.0001 and a total effective batch size of 128.

C.2 Training Loss Curves Across Three Stages

To evaluate the optimization stability and convergence of DRIFT, we illustrate the training loss trajectories across its three sequential stages below.

C.2.1 Stage 1: Latent Fact Reconstruction Pretraining (LFRP)

The training loss for the LFRP stage is presented in Figure 5. The curve shows a steady decline and eventual plateau, indicating that the knowledge model successfully learned to reconstruct factual content into the latent space with high fidelity.

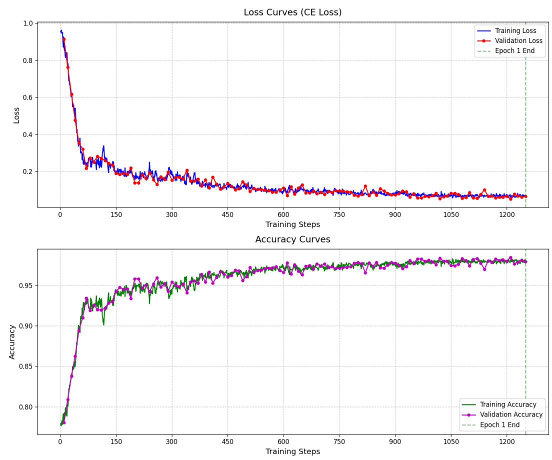

Figure 5: Training loss trajectory of Stage 1

C.2.2 Stage 2: Query-Aware Fine-Tuning (QAFT) with Single-Context Dynamic Compression

During this stage, we fine-tune the model on the Dynamic Compression Task. The training objective focuses on the model's ability to compress and project query-relevant knowledge into the reasoning model's embedding space. The loss reflects the efficiency of information bottlenecking under query guidance.

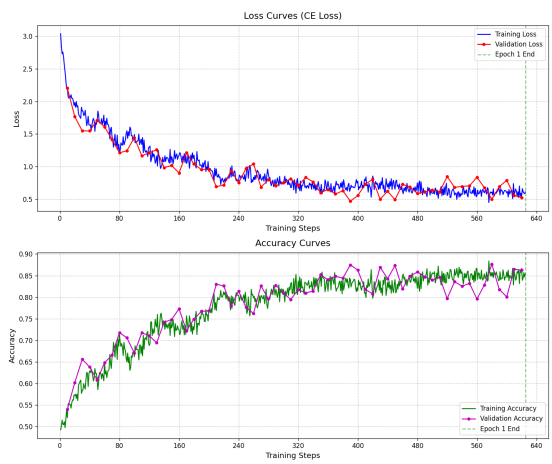

Figure 6: Training loss trajectory of Stage 2

C.2.3 Stage 3: Query-Aware Fine-Tuning (QAFT) with Single-Context Question Answering

The final stage involves end-to-end optimization for the Question-Answering (QA) task. We report the cross-entropy loss during this phase, which demonstrates how the reasoning model effectively utilizes the distilled latent facts to generate accurate and context-grounded answers.

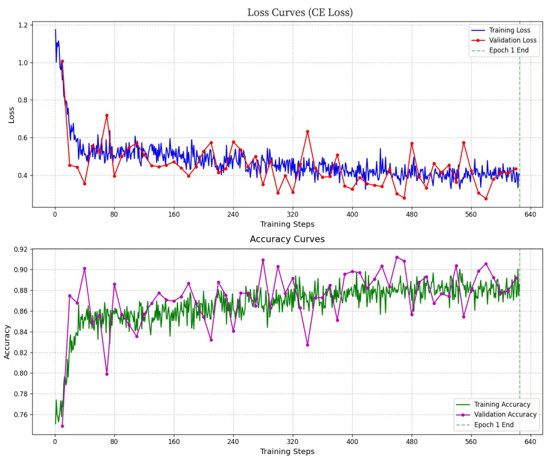

Figure 7: Training loss trajectory of Stage 3

D Additional Experimental Results

RQ7: Does DRIFT outperform the reasoning model with direct input across varied context lengths?

To compare the performance of DRIFT against the reasoning model with direct input across various context lengths, we evaluated a sampled subset from the BABILong (Kuratov et al., 2024) benchmark. The resulting comparison is illustrated in the bar chart below:

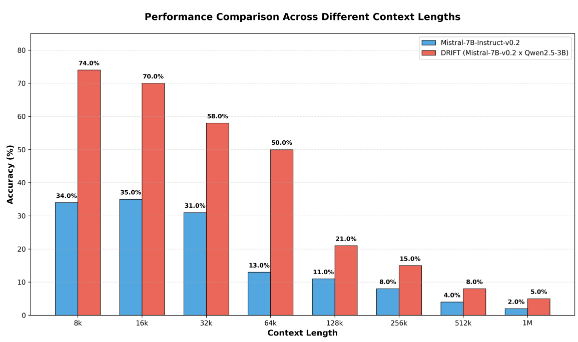

Figure 8: Comparison of accuracy between Mistral-7B-Instruct-v0.2 and DRIFT on a BABILong subset across context lengths from 8k to 1M tokens.

Experimental results indicate that DRIFT consistently outperforms the vanilla reasoning model across all tested scales, maintaining a substantial performance margin even as the context length extends to 1M tokens. Notably, at the 64k context length, DRIFT achieves nearly a $4\times$ accuracy improvement (50.0% vs. 13.0%) compared to the direct inference approach.

RQ8: Does fine-tuning degrade the reasoning model's general-purpose capabilities?

To assess whether the reasoning-oriented fine-tuning (QAFT-QA) affects the model's broad utility, we compare the general-purpose capabilities of the reasoning model before and after our proposed training. The results indicate that the degradation in generic competencies remains minimal. This suggests that our fine-tuning strategy effectively enhances reasoning performance while preserving overall versatility.

Logical Reasoning To test this capability, we utilized the BBH (Big-Bench Hard) dataset (Suzgun et al., 2023), which consists of 23 challenging tasks from the BIG-bench suite that require sophisticated multi-step reasoning where previous language models often underperformed.

Math Reasoning To test this capability, we evaluate models on the standard GSM8K test set (1,319 problems) (Cobbe et al., 2021), which is drawn from a benchmark of over eight thousand high-quality grade school math word problems requiring multi-step arithmetic reasoning and logical deduction.

Scientific Knowledge To test this capability, we utilized the GPQA Diamond dataset (Rein et al., 2024), which is a subset of the Graduate-Level Google-Proof Q&A benchmark. It contains 198 high-quality questions in biology, physics, and chemistry that have been meticulously vetted by experts to ensure they are exceptionally challenging even for highly skilled non-experts with access to the internet.

Instruction Following To test this capability, we evaluate on the IFBench benchmark (Pyatkin et al., 2025), which measures precise instruction-following generalization using a set of verifiable constraints that models must satisfy in their generated outputs.

Code Generation To test this capability, we utilized the HumanEval dataset (Chen et al., 2021), comprising 164 manually crafted Python programming problems used to evaluate the functional cor-

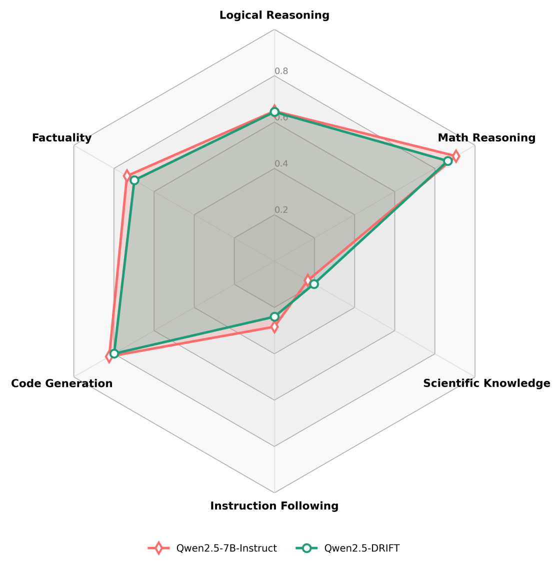

Figure 9: Radar chart illustrating general-purpose capabilities before and after reasoning-oriented fine-tuning.

rectness of code synthesized from natural language function docstrings.

Factuality To test this capability, we utilized the HaluEval dataset (Li et al., 2023a), a large-scale benchmark designed to assess hallucination levels in large language models by testing their ability to recognize and avoid generating factually incorrect information.

RQ9: How does the model align its latent reasoning with explicit evidence during training?

A critical question in our framework is whether the reasoning model merely replicates explicit textual evidence or instead develops distinct and efficient reasoning strategies within the latent space. This distinction is crucial for determining whether the implicit context functions as a complementary modality rather than a redundant compression.

To examine this behavior, we introduce a diagnostic metric, **Reasoning Consistency** (M_{ED}), which is monitored during the question-answering task in the **QAFT stage**. Specifically, M_{ED} measures the Kullback–Leibler (KL) divergence between the output distributions of the reasoning model when conditioned on compressed latent representations versus explicit textual evidence:

$$M_{\text{ED}} = D_{\text{KL}}\Big(P_{\theta_{\text{rea}}}(\cdot \mid I_{\text{ans}}, E_{\text{dyn}}, \mathrm{E}(Q)) \Big\| P_{\theta_{\text{rea}}}(\cdot \mid I_{\text{ans}}, X_{\text{evi}}, Q)\Big) \quad (11)$$

Importantly, M_{ED} is used solely as a non-intrusive diagnostic probe to analyze reasoning behavior and does not participate in gradient backpropagation.

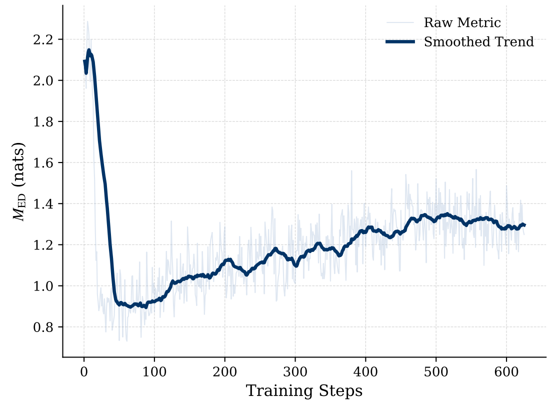

Figure 10: Evolution of the reasoning consistency metric M_{ED} during traiwoqning.

The trajectory of M_{ED} in Figure 10 exhibits a clear *Grounding-then-Specializing* learning pattern with two phases. In **Stage I (Rapid Grounding)**, M_{ED} decreases sharply, indicating that the reasoning model initially aligns its latent representations closely with explicit textual evidence to ensure faithful reasoning. In **Stage II (Emergent Specialization)**, M_{ED} gradually increases and stabilizes, suggesting that the model moves beyond strict textual alignment and develops more specialized and efficient reasoning strategies in the latent space. Overall, this behavior shows that the implicit context is not a mere compressed copy of the input, but a complementary modality that supports task-oriented reasoning beyond surface-level text.

RQ10: Is DRIFT More Robust to Indirect Prompt Injection Attacks?

DRIFT's architecture may provide an additional defensive benefit against indirect prompt injection. To examine this possibility, we conduct an additional experiment on **BIPIA** (Yi et al., 2025), a benchmark for indirect prompt injection attacks. We report **Attack Success Rate (ASR)**, which measures the percentage of cases in which the injected malicious context successfully causes the model to follow the attacker's instruction. Lower ASR indicates stronger robustness.

We evaluate DRIFT and its corresponding backbone models on two BIPIA subsets, **EmailQA** and **TableQA**. As shown in Table 8, DRIFT substantially reduces ASR across both backbones and both tasks. For example, on Mistral-7B, DRIFT reduces ASR from 23.60% to 0.00% on EmailQA and from 27.32% to 0.26% on TableQA. Similarly, on Qwen2.5-7B, DRIFT lowers ASR from 14.13% to 0.01% on EmailQA and from 13.29% to 0.09% on TableQA.

Model	EmailQA	TableQA
DRIFT (Mistral-7B)	0.00%	0.26%
Vanilla Mistral-7B	23.60%	27.32%
DRIFT (Qwen2.5-7B)	0.01%	0.09%
Vanilla Qwen2.5-7B	14.13%	13.29%

Table 8: Robustness comparison on BIPIA. We report attack success rate (ASR, lower is better) on two indirect prompt injection benchmark subsets. DRIFT consistently achieves substantially lower ASR than its corresponding backbone models.

These results suggest that DRIFT may improve robustness against indirect prompt injection by reducing the direct influence of injected raw context on the final reasoning process. Since DRIFT first compresses the input context into latent knowledge representations before answer generation, malicious instructions embedded in the retrieved or provided context may be less likely to be directly propagated to the reasoning model. However, we emphasize that this experiment should be viewed as a preliminary robustness analysis rather than a comprehensive security evaluation. A broader study across more attack settings and models remains an important direction for future work.

E Other Details

E.1 Additional Details on LLM-as-a-Judge Evaluation

For accuracy-based evaluation, we use `Qwen-2.5-72B-Instruct` as the judge model with temperature set to 0 for deterministic evaluation. The judge is prompted to make a binary decision on whether the model prediction matches the ground-truth answer. A unified evaluation prompt is used across tasks, while task-specific decision rules are specified in the instructions. For example, in multiple-choice settings, the judge is instructed to verify only whether the predicted option matches the reference option.

For invalid or unparseable judge outputs, we adopt a simple retry mechanism and re-invoke the judge up to 10 times under the same deterministic setting. If no valid decision can be obtained after all retries, the prediction is conservatively marked as incorrect. In practice, however, we did not encounter such cases in our experiments.

To verify reproducibility, we repeated the evaluation 10 times under the same configuration and observed identical accuracy scores across runs, indicating stable judge behavior in our setting. In addition, we manually inspected a sampled subset of judge decisions and found them to be broadly consistent with human judgment.

We note that the current LLM-as-a-Judge evaluation mainly focuses on whether the final predicted answer matches the ground-truth answer, and does not explicitly assess the quality of the intermediate reasoning process or knowledge transfer. Related work has also discussed the importance of going beyond outcome-only evaluation in rationale-aware judgment settings (Lai et al., 2026). We leave more process-aware evaluation protocols as an important direction for future work.

E.2 Details of End-to-End TTFT

Table 9 reports the end-to-end TTFT of DRIFT and all baselines across different context lengths.

E.3 Instructions for DRIFT

I_{sta} *"Given a text passage, condense its core concepts into a set of words. The number of these compressed words is {num}. The placeholder of compressed word is '{COMPRESSION_TOKEN}'. The text you need to condense is: <context>context</context> The compressed words are: "*

I_{rec} *"Background: <background> {compressed_information} </background>. Please restate the background information above in your own words to convey the same meaning: "*

I_{dyn} *"Given several documents and a question, you need to extract the information from the Documents that is relevant to the Question, and condense the core concepts of this knowledge into a set of words. Please note that you are only responsible for extracting information relevant to answering the question. You are not required to reason out the answer yourself. You are not allowed to fabricate information. You may only extract and compress relevant information contained in the documents. Please ensure the completeness and understandability of the compressed knowledge. The number of these compressed words is {num}." + f"The placeholder of compressed word is '{COMPRESSION_TOKEN}' The documents are: <Documents> {document}</Documents> The question is: <Question>{question}</Question>The compressed words of useful information are: . "*

I_{ans} *"You will be provided with a background consisting of {num} different paragraphs. Background: <background> {compressed_information} </background>. Please answer the following question based on the background. <Question>{question}</Question>{answer_prefix} "*

Context Length	DRIFT (Ours, 128x)	DRIFT (Ours, 32x)	Base Model (1x)	COCOM (128x)	LLMLingua-2 (3x)	NaiveRAG (-)	ICAE (4x)	xRAG (128x)
32k	1.0876s	1.1537s	1.7461s	1.3288s	0.9899s	1.3046s	3.8931s	1.0878s
64k	1.9499s	2.1438s	5.1303s	2.3652s	2.0445s	2.4411s	7.1120s	1.9925s
96k	2.8347s	3.0952s	10.1852s	3.5299s	3.3788s	3.7484s	11.0069s	3.2746s
128k	3.7697s	4.1215s	16.8847s	4.3426s	5.0518s	5.1354s	15.3403s	4.3528s
160k	4.7408s	5.2230s	25.5340s	5.5164s	6.8872s	5.4895s	14.5195s	5.0580s
192k	5.6937s	6.2129s	36.0540s	5.8981s	8.4839s	7.5512s	15.5197s	5.7713s
224k	6.6531s	7.2707s	47.9456s	7.3762s	10.9141s	8.1346s	15.2952s	7.7858s
256k	7.6741s	8.3384s	62.5092s	8.3666s	12.5249s	9.8109s	16.1304s	8.4149s
288k	8.5211s	9.4246s	77.8434s	9.4565s	14.9139s	10.5237s	16.7616s	9.5193s
320k	9.6025s	10.6386s	96.5573s	10.6911s	17.6892s	11.5716s	15.6166s	10.2454s
352k	10.3629s	11.4425s	115.3715s	11.1676s	20.6509s	12.6846s	15.7478s	11.2215s
384k	11.1033s	12.2773s	136.6740s	12.3757s	23.3289s	14.0489s	16.7980s	12.3579s
416k	11.6874s	12.7512s	159.7921s	13.9401s	26.3622s	14.5645s	16.8822s	12.7685s

Table 9: End-to-end TTFT for different baselines across various context lengths.

E.4 Reconstruction Cases of Query-Conditioned Compression

To further illustrate the behavior of query-conditioned compression, we present two qualitative cases from the QAFT-DC task. For each case, we compare the evidence reconstructed by the inference model from the compressed latent representation against the reference evidence. These examples help interpret the quantitative results in the main text and show that lexical-overlap-based metrics may underestimate reconstruction quality when the model preserves the core evidence but expresses it in a different or more expanded form.

Case 1 **Question.** Which of the following statements is true regarding the relationship between rugby and apartheid in South Africa from 1948 to 1994? A) South Africa was expelled from the International Rugby Board (IRB) during this period. B) The British Lions toured South Africa in 1986. C) South Africa was excluded from the first two Rugby World Cups. D) The South African Rugby Board (SARB) did not have any say in international tours.

Reconstructed Evidence. From 1948 to 1994, apartheid in South Africa led to increasing international isolation. In the realm of sports, rugby was a focal point of protest. While South Africa remained a member of the International Rugby Board, they faced numerous boycotts. Most notably, due to their racial policies, South Africa was excluded from the first two Rugby World Cups, held in 1987 and 1991. They only returned to the international stage as hosts and champions in 1995 after the fall of apartheid.

Reference Evidence. From 1948 to 1994, apartheid in South Africa led to increasing international isolation. In the realm of sports, rugby was a focal point of protest. While South Africa remained a member of the International Rugby Board, they faced numerous boycotts. Most notably, due to their racial policies, South Africa was excluded from the first two Rugby World Cups, held in 1987 and 1991. They only returned to the international stage as hosts and champions in 1995 after the fall of apartheid.

Metrics. BLEU: 1.0000, ROUGE-1: 1.0000, ROUGE-L: 1.0000.

Case 2 **Question.** What significant event happened to Anthrax during the recording of *Persistence of Time*, and what was the financial impact?

Reconstructed Evidence. Released in August 1990, *Persistence of Time* was written and recorded during a period of turmoil for Anthrax, as Ian had recently been divorced from his first wife, and the middle of the recording session was interrupted by a fire that caused the band to lose more than $100,000 worth of gear and their rehearsal studio. The band members also claimed it was during this period that there were talks of potentially working with a different singer, a situation that Belladonna was unaware of at the time.

Reference Evidence. Released in August 1990, *Persistence of Time* was written and recorded during a period of turmoil for Anthrax, as Ian had recently been divorced from his first wife, and the middle of the recording session was interrupted by a fire that caused the band to lose more than $100,000 worth of gear and their rehearsal studio.

Metrics. BLEU: 0.6373, ROUGE-1: 0.7862, ROUGE-L: 0.7862.

E.5 Text Reconstruction Case

A representative case of the source text and its corresponding reconstruction is presented in Figures 11 and 12, illustrating the model's ability to preserve semantic coherence under highly compressed representations.

Initial Content

Arthur Edgar Winston (12 March 1887 – 4 November 1965) was a British historian, linguist, and cultural theorist best known for his pioneering work on the "discursive margins" of 19th and early 20th-century Europe. His interdisciplinary approach to cultural history, emphasizing the recovery and analysis of non-canonical texts, profoundly influenced the development of microhistory and the linguistic turn in historical studies during the mid-20th century. Winston is most celebrated for his seminal work The Silent Text: Forgotten Narratives of the Everyday (1938), which redefined how historians interpret marginal voices in historical discourse.

Born in Bristol, England, to a family of modest means—his father was a railway clerk and his mother a schoolteacher—Winston displayed an early aptitude for languages and classical literature. He attended Bristol Grammar School on scholarship before winning a place at Balliol College, Oxford, in 1905, where he studied Classics and Modern History. At Oxford, he came under the influence of historian Lord Acton and philologist J.R. Mayor, whose emphasis on textual precision and moral historiography shaped Winston's intellectual trajectory. He graduated with first-class honours in 1909 and remained at Oxford for postgraduate research, focusing on the vernacular pamphlets of the Chartist movement.

Winston's academic career was briefly interrupted by the outbreak of the First World War. Commissioned as a lieutenant in the Royal Engineers, he served in France and Belgium, where his linguistic skills were employed in intelligence and translation. The war had a transformative effect on his worldview; he later described the trenches as "a laboratory of human expression under extremity," where he began collecting soldiers' letters, graffiti, and field notes—materials that would later inform his methodological innovations. Injured during the Battle of the Somme in 1916, he was discharged and returned to academia.

In 1920, Winston was appointed lecturer in Historical Linguistics at the University of Manchester, where he began developing his theory of "peripheral discourse." He argued that official historical records—government documents, chronicles, and elite memoirs—often obscured the lived experiences of ordinary people. Instead, he advocated for the systematic study of "residual texts": private correspondence, shopkeepers' ledgers, parish records, and oral testimonies. His 1927 monograph Voices from the Margin: Language and Class in Victorian England was among the first scholarly works to treat informal writing as a legitimate object of historical analysis.

Winston's magnum opus, The Silent Text, published in 1938, synthesized two decades of archival research across Britain, France, and Germany. The book examined how marginalized communities—industrial workers, rural women, colonial migrants—used language to assert identity and resist dominant narratives. He introduced the concept of "linguistic latency," the idea that meaning in subaltern texts is often encoded through metaphor, silence, or grammatical deviation, requiring specialized interpretive frameworks. The work received both acclaim and criticism: while praised for its methodological rigor, some contemporaries dismissed it as overly speculative or lacking in political clarity.

During the Second World War, Winston worked with the BBC's European Service, analyzing propaganda and monitoring public sentiment in occupied territories. After the war, he returned to Oxford as Professor of Cultural History, a newly established chair reflecting the growing recognition of his field. He mentored a generation of scholars, including E.P. Thompson and Raymond Williams, who acknowledged his influence on the development of British cultural studies.

Winston retired in 1957 but continued writing and lecturing. He published Echoes of the Unwritten in 1962, a reflective work on the ethics of historical interpretation. He died in Oxford on 4 November 1965, survived by his wife, Margaret (née Carr), whom he married in 1913, and their two children. Though not a public intellectual in the traditional sense, Winston's legacy endures in the methodologies of digital humanities, oral history, and postcolonial studies. In 2005, the International Society for Cultural History established the annual Winston Prize for outstanding contributions to the study of everyday discourse.

Figure 11: Case study: original passage.

Reconstruction Content

Arthur Edgar Winston (12 March 1887– 4 November 1965) was a British historian, linguist, and cultural theorist best known for his pioneering work on the "discursive margins" of 19th and early 20th century Europe. His interdisciplinary approach to cultural history, emphasizing the recovery and analysis of non-canonical texts, profoundly influenced the development of microhistory and the linguistic turn in historical studies during the late 20th century. Winston is most celebrated for his seminal work The Silent Text: Forgotten Narratives of the 1930s (which redefined how historians interpret modern voices in historical discourse).

Born in Bristol, England to a modest family—his father was a railway clerk and his mother—a schoolteacher showed early aptitude for languages and classics. He attended Bristol Grammar School on scholarship before winning a place at Balliol College, Oxford, in 1905, where he studied Classics and Modern History. At Oxford, he came under the influence of historian Lord Acton and philosopher J.R. Pole, whose emphasis on textual precision and moral historiography shaped Winston's intellectual trajectory. He graduated with first-class honours in 1909 and remained at Oxford for postgraduate research, focusing on the vernacular pamphlets of the Chartist movement.

Winston's academic career was briefly interrupted by the outbreak of the First World War. Commissioned as a lieutenant in the Royal Engineers, he served in France and Belgium, where his skills in linguistic intelligence were employed in the war. A transformative effect had on his worldview; later he described the trenches as "a laboratory of human expression under extremity," where he began collecting soldiers' letters, graffiti, and field notes—materials that would later inform his methodological innovations. Injured during the Battle of the Somme in 1916, he was discharged and returned to academia.

In 1920, Winston was appointed lecturer in Historical Linguistics at the University of Manchester, where he began developing his theory of "peripheral discourse." He argued that official historical records—government documents, chronicles, and elite memoirs—often obscured the lived experiences of ordinary people. Instead, he advocated for the systematic study of "resurgent texts": private correspondence, shopkeepers' ledgers, parish records, and oral testimonies. His 1927 monograph Voices from the Margin: Language and Class in Victorian England was among the first scholarly works to treat informal writing as a legitimate object of historical analysis.

Winston's magnum opus, The Silent Text, published in 1938, synthesized two decades of archival research across Britain, France, and Germany. The book examined how the marginalized communities—industrial workers, rural women, colonial migrants—used language to assert identity and resist dominant narratives. He introduced the concept of "linguistic latency," the idea that meaning in modern texts is often encoded through metaphor, silence, or grammatical deviation, requiring specialized interpretive frameworks. The work received both acclaim and criticism: while praised for its methodological rigor, some contemporaries dismissed it as overly speculative or lacking in political clarity.

During the Second World War, Winston worked with the BBC's European Service, analyzing propaganda and monitoring public opinion in occupied territories. After the war, he returned to Oxford as Professor of Historical Linguistics, a newly established chair reflecting the growing recognition of his field. He mentored a generation of scholars, including E.P. Thompson, Raymond Williams, and who acknowledged his influence on the development of British cultural studies.

Winston retired in 1957 but continued writing and lecturing. He published Echoes of the Unwritten in 1962, a reflective work on the ethics of historical interpretation. He died in Oxford on 4 November 1965, survived by his wife, Margaret (née Carr), whom he married in 1913, and their two children. Though not a public intellectual in the traditional sense, Winston's legacy endures in the methodologies of digital humanities, oral history, and postcolonial studies. In 2005, the International Society for Cultural Studies established the annual Winston Prize for outstanding contributions to the study of everyday discourse.

Figure 12: Case study: reconstructed passage.