HAWK: Highlighting Entity-aware Knowledge for Alleviating Information Sparsity in Long Contexts

Seonmin Koo[1*], **Jinsung Kim**[1*], **Chanjun Park**[2†], **Heuiseok Lim**[1†]
[1]Department of Computer Science and Engineering, Korea University
[2]School of Software, Soongsil University
{fhdahd,jin62304,limhseok}@korea.ac.kr, chanjun.park@ssu.ac.kr

Abstract

As the textual data given as the context of various tasks lengthens, having necessary information scattered throughout makes it more difficult for large language models (LLMs) to capture relevant details. This challenge is particularly prominent in tasks such as question answering (QA), where key information is often not evenly distributed within the context. This problem of *information sparsity* has led to the attempts of various approaches, such as direct context adjustment and retrieval-based methods. However, these approaches typically leverage compressed contexts, which increases the risk that key information may be contained in the dropped portions. Therefore, research from the perspective of addressing the *information sparsity* while not losing key details in contexts is required. To address this issue, we propose **H**ighlighting entity-**AW**are **K**nowledge (**HAWK**) framework. **HAWK** consists of three main steps: i) entity extraction, ii) entity-aware subcontext selection, and iii) triplet construction. The core mechanism of **HAWK** is to highlight key information in a context and structuralize it in an entity-aware manner, facilitating knowledge-enhanced generation. Through extensive experiments and comprehensive analysis, **HAWK** confirms significant improvements in QA tasks with long contexts, achieving up to a 27.6-point F1 score increase and at least an average win rate of 76.75% over existing methods.

1 Introduction

The increasing demand for processing longer textual data in the real world has led to extending the length of context windows that large language models (LLMs) can understand (Achiam et al., 2023; Jiang et al., 2023a). Despite the growing interest in the context lengths, longer contexts still pose an obstacle to adequate grounding and generation (Li et al., 2023a; Liu et al., 2024b). In particular, addressing the salient information scattered throughout a long context is a challenge that needs to be solved (Jiang et al., 2023c; Wu et al., 2024). This challenge is especially prominent in tasks such as question answering (QA), where key information is often unevenly distributed within the context.

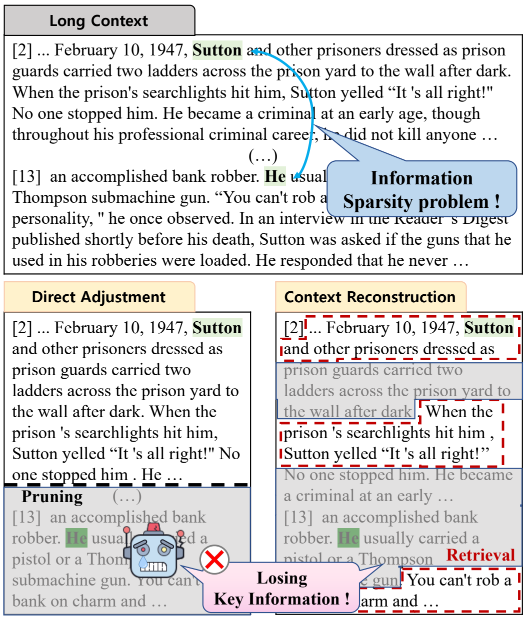

Figure 1: *Information sparsity* problem that occurs in long context scenarios and the potential risk of existing methods. The numbers in brackets indicate the paragraph indices.

We term this problem, where information critical for understanding contexts is scattered, the *information sparsity* problem. Many previous studies have actively proposed methods such as direct context length adjustment and retrieval-based approaches to address this problem in long context scenarios[1]. For example, to lower the understanding difficulty by directly reducing the length of a given context, methods such as pruning the context to a certain

[*] Equally contributed.
[†] Corresponding author.

[1]For a taxonomy of existing methods, see Appendix A.1.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 13161–13184
November 4-9, 2025 ©2025 Association for Computational Linguistics

length or sliding through a relatively small window have been studied (Jiang et al., 2023b; Li et al., 2023b). In addition, summarization or retrieval-based approaches in contexts have been attempted to reduce the amount of information LLMs must consider (Robertson et al., 2009; Gao et al., 2023; Vu et al., 2023; Karpukhin et al., 2020).

However, since these methods truncate or compress only a part of the given context, there is a risk that key information is included in the dropped part (Vodrahalli et al., 2024). Figure 1 illustrates the problem of *information sparsity* in a long context and the potential limitations of existing methods. For example, the clues about 'Sutton' mentioned in the second paragraph are quite a distance from those in the thirteenth paragraph, and various distractions may exist in between. The 'pruning' method (bottom left) that cuts a given context to a certain length may not consider important information included in a range outside the fixed length, and the same risk exists when reconstructing the original context with only some content through 'retrieval' (bottom right). In other words, coarsely dropping a large amount of input may cause information loss and hurt the task performance (Shi et al., 2023). From this perspective, research is needed on effective methods to address the *information sparsity* problem in long context scenarios while not losing salient details. Therefore, we conduct this study with the following research question:

RQ) How can the information sparsity of long contexts be alleviated without losing salient information?

To address this issue, we are inspired by cognitive science and education studies that state that structuring and highlighting knowledge enhances human understanding and reasoning abilities (Gentner, 1983; Kintsch, 1998; Caccamise and Snyder, 2005). Following these studies, we aim to highlight salient information in long contexts and structuralize it as knowledge.

Therefore, we propose **H**ighlighting entity-**AW**are **K**nowledge (**HAWK**) framework to densify salient details in long context scenarios. The core mechanism of **HAWK** is to highlight key information in the context and structure them in an entity-aware manner, thereby enabling the model to perform knowledge-enhanced generation without information loss in long contexts. **HAWK** consists of three main steps: i) entity extraction, ii) entity-aware subcontext selection, and iii) triplet construction. Moreover, a choice technique for high-quality key information is introduced through multi-agents-based entity filtering to improve the quality of the knowledge structuring process. Through extensive experiments and detailed analysis, **HAWK** has proven its effectiveness by showing up to 27.6% improvement in F1 score in long context QA scenarios and at least an average win rate of 76.75% in LLM evaluation.

2 Related Work

2.1 Strategies for Long Context Processing

Transformer-based models struggle with long contexts due to the quadratic cost of self-attention (Lin et al., 2022). To address this, three main strategies have emerged: positional extrapolation and interpolation (Chen et al., 2023b,a; Su et al., 2024), context window segmentation and sliding (Hao et al., 2022; Ratner et al., 2022; Xiao et al., 2023), and prompt compression (Jiang et al., 2023b,c; Li et al., 2023b).

Positional extrapolation and interpolation extend positional encodings to support longer sequences (Press et al., 2021). Context window segmentation and sliding split inputs into manageable segments while preserving coherence (Hao et al., 2022; Ratner et al., 2022). Prompt compression methods such as LLMLingua (Jiang et al., 2023b) condense inputs by removing redundancy, improving efficiency. These approaches improve long-sequence processing, though truncation and compression can risk excluding essential information.

2.2 Information Sparsity and Evidence Highlighting

As LMs adopt longer context windows, they increasingly face the challenge of information sparsity, where key information is distributed across distant positions. Retrieval-Augmented Generation (Lewis et al., 2020) and summarization-based reconstruction (Laban et al., 2024) attempt to address this by condensing relevant content. Chain-of-Thought (CoT) prompting (Kojima et al., 2022) further leverages the model's reasoning abilities through pre-trained patterns. Chain-of-Note (Yu et al., 2024) addresses the measurement of uncertainty at the semantic level, rather than focusing solely on lexical variation, by leveraging the concept of semantic entropy.

However, these methods often struggle to retain

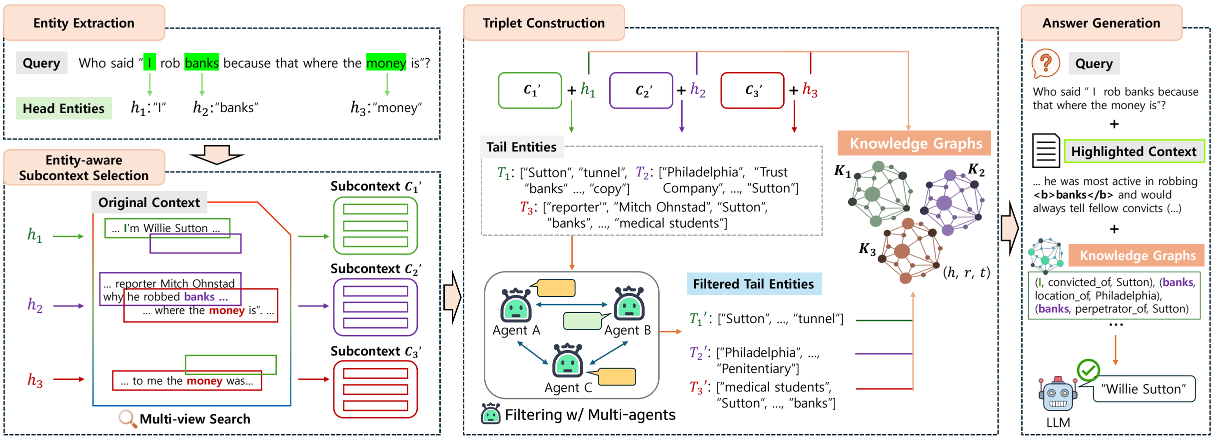

Figure 2: Overview of the proposed framework, **HAWK**.

both completeness and structural coherence of the context. To this end, Lv et al. (2024) propose entity highlighting as a viable alternative, offering cues that improve reasoning while preserving original context structure. Yet, challenges remain in modeling complex relationships when salient information is sparsely located.

Drawing inspiration from information extraction methodologies, we propose a framework that maintains full context while applying targeted highlighting inspired by information extraction, enabling effective reasoning through focused evidence clues.

3 Method

3.1 Inspirations from Cognitive Science

This study draws inspiration from cognitive science and educational research; it posits that the structuralization and emphasis of knowledge significantly improve human comprehension and reasoning abilities. Human cognition is fundamentally dependent on structured representations, which facilitate analogical reasoning as outlined by Gentner (1983) and Kintsch (1998). Moreover, structured knowledge has been demonstrated to positively impact learners' memory retention and cognitive frameworks, as evidenced by the work of Caccamise and Snyder (2005).

Kintsch et al. (1977) emphasizes that the ease of comprehension is directly related to the number of propositions, not simply the length of sentences. In other words, this comprehension is possible by structuring given texts into basic meaning units. It does not represent all surface features of language but 'semantic relations' that are important for humans to understand, remember, and think (Kintsch and Keenan, 1973; Goetz et al., 1981).

In this study, we confirm that highlighting key information from a given long context and structuring them into triplets (or knowledge graphs), a representative form of knowledge representation, supports LLMs' understanding and inference abilities. To address the problem of *information sparsity* due to the increased context length, we believe that one direction we should take is to utilize structured knowledge as clues for inference. Although attempts to highlight specific information in the context have been made (Lv et al., 2024), they lacked a critical role as clues for model understanding because they did not structuralize the appropriate relations between entities and turn them into structured knowledge. We highlight entities that are key to the context and structuralize them entity-awarely so that the model can perform knowledge-enhanced generation without information loss in the long context.

3.2 Denotations

The basic denotations in the long context-based QA situation are described. The original context is denoted C, the query is denoted q, and the corresponding ground-truth answer is denoted y. Also, when C and q are given, the response generated by model M is $\hat{y}$.

The set of triplets, which is a structured knowledge in an entity-aware manner by utilizing the proposed framework **HAWK**, is $TP = \{(h_1, r_{1,2}, t_2), \ldots\}$, where TP consists of head entities $H = \{h_1, h_2, h_3, \ldots h_k\}$ and tail entities $T = \{t_1, t_2, t_3, \ldots h_j\}$ and the relation $r_{k,j}$ between them.

3.3 HAWK Framework

An overview of **HAWK** framework is illustrated in Figure 2. The **HAWK** framework goes through multiple main steps to build entity-aware knowledge to enhance model generation: i) entity extraction, ii) entity-aware subcontext selection, and iii) triplet construction. In other words, this framework extracts and constructs concentrated knowledge centered on entities from long contexts where helpful information is scattered. Salient information is structured in the form of triplets (or knowledge graphs) to construct high-quality knowledge, and entity-aware tasks are performed during this process. The informativeness of the constructed knowledge is guaranteed through an approach based on entities from queries and contexts rather than random phrases.

3.3.1 Entity Extraction

Research on entities has conventionally been widely conducted in the field of information extraction (IE). Still, recently, it has also shown its effectiveness in various application tasks such as QA and conversation systems (Nan et al., 2021; Zhang et al., 2022; Jang et al., 2024). Also, entities are significant in that they are usefully utilized as components of contextual clues (Cucerzan and Yarowsky, 1999).

A set of head entities H is extracted from a query q corresponding to a given context C. The number of individual head entities (h) included in H is obtained variably depending on the length or complexity of q, and the number of h is at least 1. That is, $H = \{h_1, h_2, \cdots, h_{|H|}\}$ ($|H| \geq 1$). Extracting query-based entities serves as the basis for appropriately finding and structuring key information that is located differently depending on the query, even in the same context.

3.3.2 Entity-aware Subcontext Selection

In the entity-aware subcontext selection phase, we introduce a method to consider the original context through multiple viewpoints based on entities. The existing retrieval method, which extracts subcontext by embedding the entire query at once, may not accurately capture the grounds in the context corresponding to each entity in the query. In contrast, **HAWK** considers multiple local viewpoints from the global full context in a way that is aware of the head entities. In other words, we focus on detailed viewpoints by individually selecting a corresponding subcontext for each head entity, and the overlapping portions between local subcontexts have the advantage of implicitly bridging the interrelationships between the head entities.

Considering the set of head entities H extracted, a set of all subcontexts C' is constructed from the global context C. In detail, when C is a set of as many chunk texts c as $|C|$, C'_k, a subset of chunk texts that are lexically or semantically similar to entity h_k ($\in H$), is constructed. That is, since each subcontext C'_k corresponding to head entity h_k is constructed as many as the number of components in H, $C' = \{C'_1, ..., C'_k, ..., C'_{|C'|}\}$.

To formalize, let $S(\cdot)$ be the selection fitness function that takes as input an individual head entity h and a text chunk c, and the set of subsets, C' is computed as follows:

$$C'_k = \{c \mid \underset{c \in C}{S(h_k, c)} = \text{True}\} \tag{1}$$

$$C' = \bigcup_{h_k \in H} C'_k \tag{2}$$

For example, if h_1 ($\in H$), fits with chunks c_1, c_{15}, c_{31} in the original context C, then the corresponding subcontext C'_1 is $\{c_1, c_{15}, c_{31}\}$. In this way, the entity-aware subcontext selection step extracts meaningful subsets in a way that is aware of the key head entities, which becomes the foundation for entity-aware knowledge construction.

3.3.3 Triplet Construction

Triplet, one of the most typical forms of knowledge representation, has been considered significant in knowledge graph research and plays a valuable role in representing world knowledge (Zhang et al., 2018; Rosso et al., 2020; Liu et al., 2024a). Knowledge structuralized using this can serve as clues for cognitive scientific reasoning by alleviating *information sparsity* in long context scenarios (Gentner, 1983; Kintsch, 1998).

Algorithm 1 (in the Appendix A.2) outlines the overall procedure of triplet (or knowledge graph) construction. In this step, the additional components that must be obtained to construct a triplet (h, r, t) are 1) tail entities and 2) predicates between head and tail entities.

Tail Entity Set Acquisition and Filtering. First, considering the head entity h_k, a set of tail entities T is extracted from the corresponding subcontext C'_k. After that, a multi-agents-based filtering process is performed to improve the quality of the extracted tail entities. Figure 3 illustrates this filtering

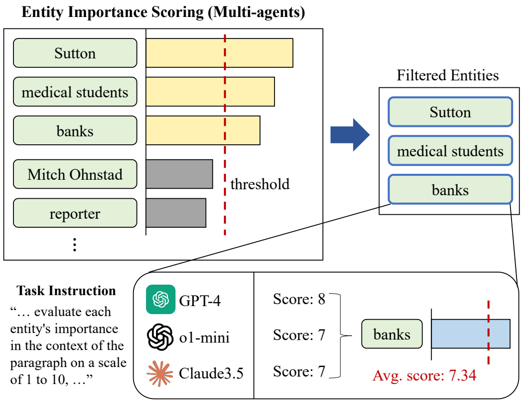

Figure 3: Entity filtering process with multi-agents

process. For the set of tail entities T extracted from each subcontext C'_k, an importance score threshold θ is calculated by majority voting among multi-agents. In this process, a set of high-quality tail entities is obtained, exceeding θ, $T_{filtered} = \{t_1, t_2, \ldots, t_n\}$.

Predicate Generation. Afterward, a predicate that adequately describes the relation between the head h and the filtered tail entity t ($\in T_{filtered}$) is generated through LLM M. The extracted relation r is added to the set of triplets TP to build knowledge graphs. This effectively structures the sparse information spread within the long context.

3.3.4 Answer Generation

Moreover, to bridge the dynamics between the original context and the triplets structured in an entity-aware manner, the head entities from the query are highlighted in the context C with special tokens. Finally, the QA model takes as input the query q, the highlighted context $C_{highlighted}$, and the knowledge graphs TP constructed via the **HAWK** framework and generates the response $\hat{y}$. Through this knowledge-enhanced and -highlighted approach, an appropriate response can be generated by grounding the sparsely scattered salient information in the long context without information loss.

4 Experimental Setup

In Appendix A, additional experimental settings, including hyperparameters, dataset statistics, model details, and prompt templates, can be found. Also, the taxonomy of baseline methods for comparison with **HAWK** is described.

Datasets. This study adopted the Natural Questions (NQ) dataset (Kwiatkowski et al., 2019) as the raw source data, which is widely used in many QA studies. For experiments on long context scenarios, a subset of examples with context lengths of 16K or more was curated. Additionally, to further investigate the generalizability of our method in multi-hop QA scenarios, we adopt the MuSiQue dataset (Trivedi et al., 2022).

Metrics. The evaluation is carried out in two ways: through existing automated metrics, including F1 score, and an LLM-based evaluation. First, automated metrics, which primarily evaluate lexical overlap with the ground-truth answer, are adopted, including F1, Rouge-L, BLEU (n=1, 2, 4) (Papineni et al., 2002) scores, and BERTScore (Zhang et al.). Additionally, to evaluate the generation results from various perspectives, we employed the LLM-as-a-Judge approach (Zheng et al., 2023), which calculates the win-tie-lose rates by evaluating the generation results in a pair-wise manner.

5 Results and Analysis

Additional results and qualitative analysis examples not covered in the main body can be found in Appendix B.

5.1 Automated Evaluation

Insight 1: highlighted and structured knowledge of salient information is practical in long-context situations. Table 1 shows the performance of various models and methods in the QA with long contexts. **HAWK** framework consistently outperforms the other approaches. Notably, the LLaMA3.1 model demonstrates the most substantial improvement across all results, achieving a 27.6% increase in F1 score compared to the vanilla method. The other models also show significant improvements in F1 scores: 14.21% (ChatGPT), 27.47% (Claude3.5), and 26% (Mistral). Furthermore, the Rouge-L score reveals improvements ranging from 13.43% (ChatGPT) to 43.16% (Claude3.5) over the vanilla method.

On the other hand, other baseline methods tend to show slight performance degradation compared to the vanilla method, which only provides task instructions. For instance, pruning methods that directly trim the given context to a fixed length show a decrease of up to 5.85% (Claude3.5) in F1 score across all models. In other words, while other methods did not show significant improvements in performance, **HAWK** framework demonstrated sub-

Backbone	Method		F1	Rouge-L	BLEU (n=1/2/4)			BERTScore
ChatGPT	Vanilla		56.59	37.93	19.23	14.63	9.46	81.78
	Direct Adjustment	Pruning	54.77	33.23	16.39	10.64	5.38	80.03
		Windowing	57.20	33.54	17.46	11.31	5.86	79.49
		Summarization	58.71	34.61	15.27	8.85	2.85	80.06
	Context Reconstruction	RALM (Vu et al., 2023)	62.01	39.39	20.67	13.20	7.15	81.50
		DPR (Karpukhin et al., 2020)	60.77	37.85	20.20	13.61	7.68	81.49
	Reasoning-enhanced	CoT (Kojima et al., 2022)	46.28	29.39	16.43	11.67	6.79	78.46
	HAWK (Ours)		**70.80**	**51.36**	**25.58**	**20.37**	**13.38**	**86.73**
Claude3.5	Vanilla		37.11	10.73	8.46	5.24	2.77	73.99
	Direct Adjustment	Pruning	31.26	8.05	6.18	3.48	1.27	72.07
		Windowing	33.06	7.76	6.55	3.78	1.44	71.14
		Summarization	33.57	7.10	6.14	3.41	1.34	71.96
	Context Reconstruction	RALM	35.62	8.71	7.86	5.05	2.53	71.47
		DPR	39.37	9.19	8.59	5.55	2.58	72.72
	Reasoning-enhanced	CoT	13.70	7.17	4.07	2.54	1.08	69.49
	HAWK (Ours)		**64.58**	**53.89**	**21.90**	**16.29**	**7.02**	**74.33**
LLaMA3.1	Vanilla		46.39	29.76	15.11	11.98	8.42	77.29
	Direct Adjustment	Pruning	43.22	19.80	10.51	6.31	2.55	74.83
		Windowing	42.57	20.06	10.25	6.49	3.08	73.95
		Summarization	43.05	19.28	5.61	3.17	1.06	74.61
	Context Reconstruction	RALM	43.50	20.37	10.61	6.87	3.43	73.82
		DPR	44.53	25.09	12.99	9.28	5.54	75.64
	Reasoning-enhanced	CoT	40.93	24.92	12.48	**27.59**	**18.92**	86.39
	HAWK (Ours)		**73.99**	**53.47**	**19.46**	15.98	10.45	**87.31**
Mistral	Vanilla		40.27	25.60	9.85	6.83	3.73	76.29
	Direct Adjustment	Pruning	36.55	18.82	8.39	5.17	2.07	73.70
		Windowing	35.57	16.30	8.07	5.14	2.29	72.46
		Summarization	36.32	18.63	8.48	4.71	1.37	74.01
	Context Reconstruction	RALM	38.05	16.95	8.93	5.85	2.79	72.52
		DPR	41.18	20.91	11.56	8.02	4.69	74.18
	Reasoning-enhanced	CoT	32.35	18.83	8.78	6.03	3.03	75.18
	HAWK (Ours)		**66.27**	**46.52**	**27.50**	**20.05**	**10.03**	**84.97**

Table 1: Overall results on QA task. **Bold** text indicates the best performances.

stantial performance gains in long-context-based QA. This suggests the effectiveness of highlighting and structuring knowledge about salient information in long-context scenarios.

5.2 LLM Evaluation

Insight 2: highlighted and structured knowledge plays a crucial role in ensuring semantically accurate answers. In addition to the existing automated metrics, such as lexical overlapping-based F1 score, we ensure the reliability of the results through LLM-based evaluation. Figure 4 shows the results of comparing and evaluating the outputs of various models using the 'LLM-as-a-Judge' (Zheng et al., 2023) approach. Specifically, it presents the win-tie-lose rates between the outputs generated by the vanilla method, representative baseline approaches[2], and the proposed HAWK framework. The evaluation is conducted pairwise, where the results of two methods are compared simultaneously, and the superior output is selected based on quality.

The HAWK framework exhibited average win rates of 85.25% (ChatGPT), 76.75% (Claude 3.5), 82.5% (LLaMA 3.1), and 85% (Mistral) across the four models. In particular, HAWK achieved the highest performance with a 90% win rate compared to the results by the vanilla method, which only provided task instructions, on LLaMA3.1.

5.3 Effectiveness of Multi-agent-based Filtering

Insight 3: Multi-agent-based filtering facilitates effective utilization of high-quality entities. Figure 5 shows the change in the number of tail entities after the filtering process based on the multi-agents. According to the reduction rates of each model, the number of filtered tail entities decreases by a minimum of 27% (ChatGPT) to a maximum of 51.89%

[2]For the baseline method descriptions, please refer to Appendix A.1.

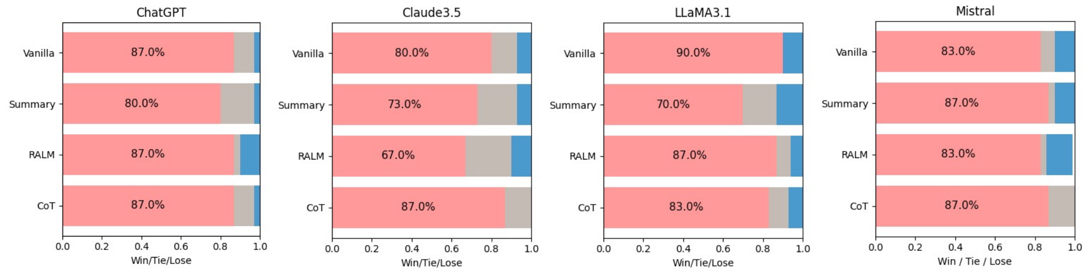

Figure 4: LLM-as-a-Judge (pair-wise) results between the **HAWK** framework and other methods. Red, gray, and blue portions indicate the win rate, the tie rate, and the lose rate, respectively.

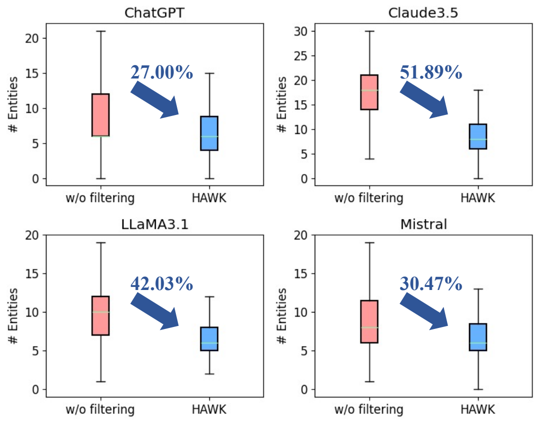

Figure 5: Distribution changes in the number of tail entities before and after the filtering process for the adopted models. The blue arrows (→) and numbers (e.g., 30.47%) indicate the average rate of decrease in the number of tail entities.

(Claude3.5) compared to the initially extracted tail entities.

Furthermore, Figure 6 illustrates the performance changes in QA tasks with and without the filtering process. Through the filtering process, each model shows performance improvements of 3.23%, 1.56%, 0.88%, and 3.53%, respectively. This implies that entities that act as noise in the knowledge construction process are effectively sifted out through the process using multi-agents.

Even though tail entities were removed by a minimum of 27% and a maximum of about 42.03% for the models, the QA performance increased by up to 3.23% (ChatGPT). This indicates the effectiveness of utilizing a multi-agents-based approach in filtering tail entities corresponding to head entities from a query.

5.4 Variation of Subcontext Selection

Insight 4: LLMs-based subcontext selection can actively leverage the knowledge of LLMs. Table 2 shows the performances of the Mistral model for the variants of the method used in the subcontext selection step, which is one of the knowledge construction steps in the **HAWK** framework[3].

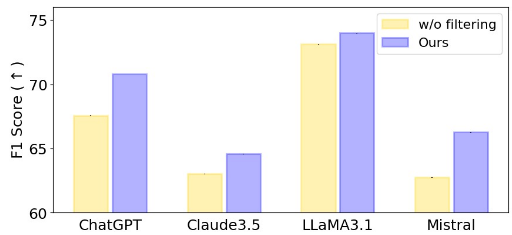

Figure 6: QA performance comparison of models with and without multi-agents-based filtering process in the triplet construction step

Method	F1	Rouge-L	BLEU (n=1/2/4)			BS
HAWK (LLM)	**66.27**	**46.52**	**27.05**	**20.05**	**10.03**	**84.97**
w/ Random	61.43	38.43	22.72	15.59	7.80	82.06
w/ String Match	65.92	45.96	19.11	13.56	6.21	84.66
w/ BM25	65.92	41.74	19.81	14.17	6.26	83.74

Table 2: Performance comparison of Mistral according to the variation of the subcontext selection method. The metric 'BS' indicates BERTScore.

In **HAWK**, the same LLM employed during the query answer generation phase is also utilized for subcontext selection. Additional experiments were conducted to compare the effectiveness of LLM-based selection with alternative methods that replace it, including random choice, string matching, and retrieval-based selections.

According to the results, **HAWK** framework with LLMs-based subcontext selection tends to achieve significant improvements across automated evaluation metrics. Specifically, when compared to random selection, which generally showed the poorest performance, **HAWK** demonstrated improvements of

[3]Due to space constraints, we focus on the Mistral, which has the smallest parameter size among the open-source models in our experiments.

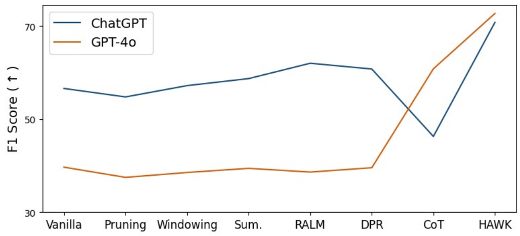

Figure 7: Results on model variation within the GPT family. ‘Sum.’ indicates the summarization method.

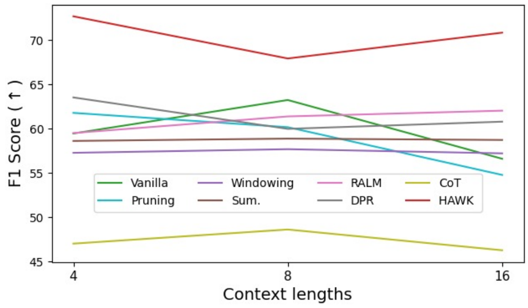

Figure 8: Changes in QA results by context length.

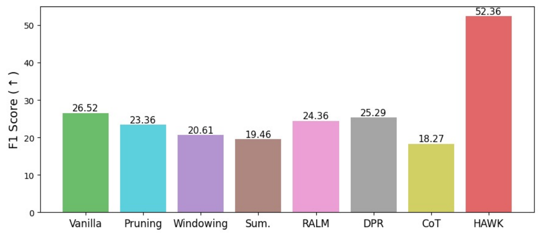

Figure 9: Experimental results on a representative multi-hop QA benchmark, MuSiQue.

4.85% and 8.09% in F1 and Rouge-L scores, respectively. It outperformed other methods that rely on lexical matching or term frequency. In contrast to the other methods, the advantage of utilizing LLMs lies in actively leveraging the model’s inherent knowledge, allowing consideration of various factors. Additionally, this method can achieve effective outcomes through meticulously crafted prompt configurations, eliminating the need for independent module setups.

5.5 Variation within the Model Family

Insight 5: highlighted and structured knowledge is still effective across in-family variations. Figure 7 demonstrates the results regarding model variants within the GPT family (ChatGPT and GPT-4o). The results show that the baseline methods have inconsistencies in which the score gap between model families is large in either direction. Performance degradation was observed, particularly in the case of the CoT method, which is known to be effective in enhancing the reasoning ability of LLMs, contrary to other methods. Compared to the vanilla method’s F1 score of 56.59% in ChatGPT, the CoT method resulted in a lower F1 score of 46.28%. However, **HAWK** robustly demonstrates effectiveness across model variants, making it a more consistent performance improvement than CoT.

5.6 Generalization Feasibility under Length Variation and Multi-hop Scenario

Insight 6: highlighted and structured knowledge can achieve improvements beyond single-hop QA. As shown in Figure 8, **HAWK** consistently demonstrates robust improvements across varying context length settings, achieving significantly higher performance compared to all baseline methods at every context length. According to the results, other baseline methods exhibit inconsistencies in their performance gap relative to the vanilla method (depicted by the green line). Notably, at a context length of 8k, all baseline methods—excluding **HAWK**—achieve performance scores at least 1.85 points lower than the vanilla method (when compared with the RALM method).

Furthermore, Figure 9 illustrates the generalization potential of the **HAWK** framework in a multi-hop QA setting. **HAWK** achieves a substantial performance gain of 25.84 points over the vanilla method, whereas all other baseline methods exhibit at least a 1-point drop in performance relative to the vanilla baseline. In particular, when compared against the zero-shot CoT approach—which shows the most significant performance degradation—**HAWK** achieves an F1 score improvement of 34.09 points.

These results suggest that the **HAWK** framework consistently enhances knowledge-grounded generation across a variety of scenarios beyond single-hop QA settings, resulting in generalization potentials.

6 Conclusion

Despite the increasing length of the context window of LLMs, challenges such as the *‘lost in the middle’* phenomenon (Liu et al., 2024b) continue to hinder the effective capture and retention of critical information within lengthened contexts. This study focuses on alleviating the *information sparsity* problem, which is challenging when utilizing long contexts. Many existing studies have adopted approaches that coarsely truncate or compress the given context; however, these methods cannot completely eliminate the risk of omitting crucial information.

Therefore, we propose **HAWK**, a knowledge-densified framework that highlights and structures key information in an entity-aware manner while preventing the loss of important contents. Extensive experiments demonstrate that our knowledge augmentation approaches can enhance problem-solving ability in long-context QA scenarios.

Limitations

While **HAWK** enhances the capability to process long contexts through entity-aware knowledge highlighting techniques in LLMs, it relatively lacks the ability to directly modify the parametric knowledge embedded within LLMs. Addressing the challenge of information sparsity in NLP tasks through direct knowledge updates involving extensive contexts remains a priority for future research. This future work will aim to identify effective and cost-efficient methods for updating knowledge within LLMs, and to utilize this strategy to simplify the resolution of complex, long context-based NLP tasks with knowledge-enhanced LLMs.

Ethics Statement

We discuss the main ethical considerations of the framework we proposed: (1) Privacy. the datasets adopted to experiment with our framework provide factual information sourced from the web or Wikipedia, and our verification results do not contain privacy issues. (2) Potential problems. Although we take conscientious steps to ensure the quality of our framework and resources, there can still be potential problems with the quality of the generated results, which can lead to incorrect predictions in applications that leverage factual information and long contexts. (3) Model deployment. Our approach employs the pre-trained large language models (LLMs) for the downstream tasks, which have the risk of reflecting the bias of the training data. It is a well-known threat in tasks using pre-trained language models (PLMs) and LLMs, and we should be careful about social impact when using this method since our approach aims to handle factual information and long contexts.

Acknowledgement

This work was supported by the Commercialization Promotion Agency for R&D Outcomes(COMPA) grant funded by the Korea government(Ministry of Science and ICT)(2710086166). This work was supported by Institute of Information & communications Technology Planning & Evaluation(IITP) under the Leading Generative AI Human Resources Development(IITP-2025-R2408111) grant funded by the Korea government(MSIT). This work was supported by Institute for Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. RS-2022-II220369, (Part 4) Development of AI Technology to support Expert Decision-making that can Explain the Reasons/Grounds for Judgment Results based on Expert Knowledge). This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) under the artificial intelligence star fellowship support program to nurture the best talents (IITP-2025-RS-2025-02304828) grant funded by the Korea government(MSIT).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI Anthropic. 2024. Claude 3.5 sonnet model card addendum. *Claude-3.5 Model Card*, 3:6.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection.

Donna Caccamise and Lynn Snyder. 2005. Theory and pedagogical practices of text comprehension. *Topics in language disorders*, 25(1):5–20.

Guanzheng Chen, Xin Li, Zaiqiao Meng, Shangsong Liang, and Lidong Bing. 2023a. Clex: Continuous length extrapolation for large language models. *arXiv preprint arXiv:2310.16450*.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. 2023b. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*.

Silviu Cucerzan and David Yarowsky. 1999. Language independent named entity recognition combining morphological and contextual evidence. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

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

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Dedre Gentner. 1983. Structure-mapping: A theoretical framework for analogy. *Cognitive science*, 7(2):155–170.

Ernest T Goetz, Richard C Anderson, and Diane L Schallert. 1981. The representation of sentences in memory. *Journal of verbal Learning and verbal Behavior*, 20(4):369–385.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. 2025. From rag to memory: Non-parametric continual learning for large language models. *arXiv preprint arXiv:2502.14802*.

Yaru Hao, Yutao Sun, Li Dong, Zhixiong Han, Yuxian Gu, and Furu Wei. 2022. Structured prompting: Scaling in-context learning to 1,000 examples. *arXiv preprint arXiv:2212.06713*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Yoonna Jang, Suhyune Son, Jeongwoo Lee, Junyoung Son, Yuna Hur, Jungwoo Lim, Hyeonseok Moon, Kisu Yang, and Heuiseok Lim. 2024. Post-hoc utterance refining method by entity mining for faithful knowledge grounded conversations. *arXiv preprint arXiv:2406.10809*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023a. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2023b. Llmlingua: Compressing prompts for accelerated inference of large language models. *arXiv preprint arXiv:2310.05736*.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2023c. Longllmlingua: Accelerating and enhancing llms in long context scenarios via prompt compression. *arXiv preprint arXiv:2310.06839*.

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

Walter Kintsch. 1998. *Comprehension. A Paradigm for Cognition*. Cambridge University Press, Cambridge, UK.

Walter Kintsch and Janice Keenan. 1973. Reading rate and retention as a function of the number of propositions in the base structure of sentences. *Cognitive psychology*, 5(3):257–274.

Walter Kintsch et al. 1977. The representation of meaning in memory. *General Linguistics*, 17(2):124–127.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

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

Philippe Laban, Alexander R Fabbri, Caiming Xiong, and Chien-Sheng Wu. 2024. Summary of a haystack: A challenge to long-context llms and rag systems. *arXiv preprint arXiv:2407.01370*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Jiaqi Li, Mengmeng Wang, Zilong Zheng, and Muhan Zhang. 2023a. Loogle: Can long-context language models understand long contexts? *arXiv preprint arXiv:2311.04939*.

Yucheng Li, Bo Dong, Chenghua Lin, and Frank Guerin. 2023b. Compressing context to enhance inference efficiency of large language models. *arXiv preprint arXiv:2310.06201*.

Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. 2022. A survey of transformers. *AI open*, 3:111–132.

Jingping Liu, Mingchuan Zhang, Weichen Li, Chao Wang, Shuang Li, Haiyun Jiang, Sihang Jiang, Yanghua Xiao, and Yunwen Chen. 2024a. Beyond entities: A large-scale multi-modal knowledge graph with triplet fact grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18653–18661.

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

Qitan Lv, Jie Wang, Hanzhu Chen, Bin Li, Yongdong Zhang, and Feng Wu. 2024. Coarse-to-fine highlighting: Reducing knowledge hallucination in large language models. *arXiv preprint arXiv:2410.15116*.

Feng Nan, Ramesh Nallapati, Zhiguo Wang, Cicero Nogueira dos Santos, Henghui Zhu, Dejiao Zhang, Kathleen McKeown, and Bing Xiang. 2021. Entity-level factual consistency of abstractive text summarization. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2727–2733, Online. Association for Computational Linguistics.

OpenAI-Blog. 2022. Chatgpt: Optimizing language models for dialogue.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Ofir Press, Noah A Smith, and Mike Lewis. 2021. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*.

Nir Ratner, Yoav Levine, Yonatan Belinkov, Ori Ram, Inbal Magar, Omri Abend, Ehud Karpas, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2022. Parallel context windows for large language models. *arXiv preprint arXiv:2212.10947*.

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

Paolo Rosso, Dingqi Yang, and Philippe Cudré-Mauroux. 2020. Beyond triplets: hyper-relational knowledge graph embedding for link prediction. In *Proceedings of the web conference 2020*, pages 1885–1896.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Kiran Vodrahalli, Santiago Ontanon, Nilesh Tripuraneni, Kelvin Xu, Sanil Jain, Rakesh Shivanna, Jeffrey Hui, Nishanth Dikkala, Mehran Kazemi, Bahare Fatemi, et al. 2024. Michelangelo: Long context evaluations beyond haystacks via latent structure queries. *arXiv preprint arXiv:2409.12640*.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, et al. 2023. Freshllms: Refreshing large language models with search engine augmentation. *arXiv preprint arXiv:2310.03214*.

Siye Wu, Jian Xie, Jiangjie Chen, Tinghui Zhu, Kai Zhang, and Yanghua Xiao. 2024. How easily do irrelevant inputs skew the responses of large language models? *arXiv preprint arXiv:2404.03302*.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2023. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*.

Wenhao Yu, Hongming Zhang, Xiaoman Pan, Peixin Cao, Kaixin Ma, Jian Li, Hongwei Wang, and Dong Yu. 2024. Chain-of-note: Enhancing robustness in retrieval-augmented language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14672–14685, Miami, Florida, USA. Association for Computational Linguistics.

Jiahao Zhang, Haiyang Zhang, Dongmei Zhang, Yong Liu, and Shen Huang. 2023. End-to-end beam retrieval for multi-hop question answering. *arXiv preprint arXiv:2308.08973*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Yuyu Zhang, Hanjun Dai, Zornitsa Kozareva, Alexander Smola, and Le Song. 2018. Variational reasoning for question answering with knowledge graph. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Zhihan Zhang, Wenhao Yu, Chenguang Zhu, and Meng Jiang. 2022. A unified encoder-decoder framework with entity memory. *arXiv preprint arXiv:2210.03273*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

A Experimental Details

This section provides additional descriptions of the various experiments in this study.

A.1 Taxonomy of Baseline Methods

Method		Description	Limitation
Direct Adjustment	Pruning	To ensure the context length does not exceed the limit, the context is truncated to a predefined length.	The dropped portion may contain important information.
	Windowing	Contexts are slid based on a specific window size.	Same as Pruning
Context Reconstruction	Summarization	Reconstruct the context into a summarized version using either an abstractive or extractive approach.	Important information may be omitted during the summarizing process.
	Retrieval-based	This method utilizes retrieval technology. Representative methods based on term frequency include BM25 and TF-IDF. Additionally, it encompasses techniques that rely on similarity between embeddings, such as DPR.	Portions containing important information may be missing from the candidate chunks of the retrieved document.
Reasoning Enhancement		Approach to leveraging the model's intrinsic knowledge is through methods such as Chain-of-Thought (CoT), which are representative examples.	The key information within the context may not be identified, which can hinder its utilization in inference. This may lead to the generation of incorrect reasoning paths, posing a potential risk of hallucinations in the final predictions.

Table 3: Taxonomy of existing approaches for long context (baseline selection)

A.2 Algorithm for Triplet Construction Task

Algorithm 1 presents the pseudo-code for the triplet construction algorithm, which is designed for completing the knowledge graph (KG) as described in Section 3.3.3.

Algorithm 1 Triplet Construction Algorithm

Input: Set of head entities $\mathcal{H}$, corresponding subcontext set $\mathcal{C}'$, multi-agent list $Agent = \{Agent_1, Agent_2, Agent_3\}$, large language model $\mathcal{M}$, importance score threshold θ.

Initialize set of triplets $\mathcal{TP} \leftarrow \emptyset$
for $h_k \in \mathcal{H}$ **do**
  Prepare subcontext $C'_k \in \mathcal{C}'$ corresponding to h_k
  Use $\mathcal{M}$ to extract set of candidate tail entities $\mathcal{T} = \{t_1, t_2, \dots, t_n\}$ from C'_k
  Initialize set of filtered tail entities $\mathcal{T}_{filtered} \leftarrow \emptyset$
  for $t_i \in \mathcal{T}$ **do**
    Compute importance scores s_1, s_2, s_3 for t_i using $Agent_1$, $Agent_2$, and $Agent_3$
    Compute average score $\bar{s}_i = \frac{s_1+s_2+s_3}{|Agent|}$
    if $\bar{s}_i \geq \theta$ **then**
      Add t_i to $\mathcal{T}_{filtered}$
    end if
  end for
  for $t_j \in \mathcal{T}_{filtered}$ **do**
    Use $\mathcal{M}$ to extract relation $r_{k,j}$ between h_k and t_j
    Add triplet $(h_k, r_{k,j}, t_j)$ to $\mathcal{TP}$
  end for
end for

Output: Set of triples $\mathcal{TP} = \{\cdots, (h_k, r_{k,j}, t_j), \cdots\}$.

A.3 Models

Question Answering. Only models with context windows of over 16K tokens were selected for the QA experiments with long contexts. This is because the ability to process relatively long contexts is a fundamental prerequisite for valid experiments. For ChatGPT (OpenAI-Blog, 2022), the gpt-3.5-turbo-16k version is used, and the claude-3-5-haiku-20241022 version is utilized for Claude 3.5 (Anthropic, 2024). Additionally, the LLaMA3.1-8B (Dubey et al., 2024) and Mistral-7B (Jiang et al., 2023a) models are employed.

Filtering with Multi-agents. HAWK framework improves the efficacy of the filtering process for extracted initial tail entities by leveraging a multi-agents-based approach (§ 3.3.3). The models adopted in filtering process are o1-mini (o1-mini-2024-09-12) (Jaech et al., 2024), GPT-4o (gpt-4o-2024-08-06) (Hurst et al., 2024), and Claude3.5 (claude-3-5-haiku-20241022).

A.4 Hyperparameters

For a model generation, the temperature was set to 0, and the max generation length was set to 1000. Other parameter settings are set to default values recommended by OpenAI, Meta, Anthropic, and Mistral AI.

A.5 Tools and Implementation Details

All experiments were conducted in a zero-shot setting. This is because when dealing with long contexts, providing exemplars for another long context as input can further increase the risk of exceeding the context window of the model being tested (e.g., ChatGPT 16K).

Among the baseline methods, the windowing method (Table 3) calculates the average value of the evaluation results obtained by sliding a window with token lengths of $4k$. The BM25 (Robertson et al., 2009) method was utilized for the sparse retrieval-based approach. In addition, the dense passage retrieval (DPR) method (Karpukhin et al., 2020) was adopted for dense retrieval, and the similarity between the embeddings of all text chunks in the context and given query was calculated for this.

For a fair subcontext selection experiment, the value of k, the number of text chunks selected in all methods, was set by a statistical approach (§ 3.3.2). In other words, it is set based on the average number of extracted chunk texts when the string-based exact match method is applied to form a subcontext by using the head entities. $k = 4$ was used in this study. Based on this, the number of text chunks searched through BM25, DPR techniques, etc., was also set to $k = 4$.

A.6 Dataset Statistics

Split	# Examples
Train	307,373
Validation	7,830
Test	7,842

Table 4: Statistics of NQ datset

The benchmark for the QA task experiments in this study is the Natural Questions (NQ) dataset (Kwiatkowski et al., 2019), which has been widely used until recently. The dataset consists of examples as context and qa pairs. Table 4 shows the number of examples for each dataset split. In this study, examples with a context length of 16K or more were curated (§ 4).

Also, NQ dataset has a Creative Commons Share-Alike 3.0 license. This license requires that reusers give credit to the creator. It allows reusers to distribute, remix, adapt, and build upon the material in any medium or format, even for commercial purposes. If others remix, adapt, or build upon the material, they must license the modified material under identical terms.

Task Instruction
```
Generate an answer (A) to the question (Q) based on the given context. Please respond short answers.

Context: {{context}}
Question: {{question}}


Output Format Instruction
Please respond according to the format below.
Answer:
```

Table 5: Prompt template for the vanilla method. In addition, among the baseline methods, direct adjustment and context reconstruction types only involve modifications to the context, so the same prompt template is used.

A.7 Prompt Templates

Table 5 and 6 are the default prompt setup for QA tasks and the prompt template when utilizing the zero-shot CoT technique (Kojima et al., 2022), respectively. In the case of a vanilla prompt, only the task instruction and current input are provided, and the CoT method adds an intermediate step that generates the reasoning paths required for the final answer predictions through a trigger sentence called "Let's think step-by-step.".

Additionally, Table 7-11 show prompt templates for performing each step of **HAWK**, our framework that utilizes multiple steps of knowledge construction based on information extraction.

Table 12 shows a prompt template for the pair-wise comparison method of the LLM-as-a-Judge approach, an evaluation method introduced to judge various aspects of the results of each model's generation beyond the metrics that simply measure lexical overlapping.

Phase 1
Task Instruction
```
Generate an answer (A) to the question (Q) based on the given context. Please respond short answers.

Context: {{context}}
Question: {{question}}


Please respond according to the format below.
A:

Let's think step-by-step.
```
Phase 2
Task Instruction
```
- Reasoning path: {{reasoning_path}}

Context: {{context}}
Question: {{question}}


Generate short answers considering the reasoning results.
Answer:
```

Table 6: Prompt template for the zero-shot chain-of-thought (CoT) method.

Task Instruction
```
Your role is to perform a Named Entity Recognition (NER) task on the given input (Input). Please
extract important entities for the given input (Input). You can extract a minimum of 1 and a
maximum of 3 important entities. Each entity must consist of one or two words.

Input: {{question}}


Please respond according to the format below.
# e1:
# e2:
# e3:
...
```

Table 7: Prompt template for the entity extraction task in the proposed **HAWK** framework. In this step, key entities are extracted from a given query.

Task Instruction
```
You are an advanced text-processing AI. I will provide you with a context and a list of key entities.
Your task is to identify and extract the **top 4** sentences from the given context that includes
or are most relevant to the provided entities. Output the exact sentences from the context without
any additional explanation or formatting.

Here's the input:

Context: {{context}}

Key Entities: {{key_entities}}
```

Table 8: Prompt template example for the entity-aware subcontext selection task in the proposed **HAWK** framework.

Phase 1: Entity Extraction for Subcontext
Task Instruction
```
Your role is to perform a Named Entity Recognition (NER) task on the given input (a paragraph).
Considering the given subject, please extract relevant entities for the given input (Input).

Subject: {{subject}}
Input: {{input_paragraph}}


Please respond according to the format below.
# e1:
# e2:
# e3:
# e4:
...
```

Phase 2: Multi-agents-based Entity Filtering
(Please refer to Table 10.)

Phase 3: Relation Extraction
Task Instruction
```
Your role is to perform a Relation Extraction (RE) task for two given entities (e1, e2) in the
given input (Input).

Input: {{subcontext}}

e1: {{entity1}}, e2: {{entity2}}
```

Output Format Instruction
```
You must answer only the relation type. Please respond using the triplet format below.

Triplet: (e1, Relation, e2)
```

Table 9: Prompt template example for the triplet construction step in the proposed **HAWK** framework.

Task Instruction
```
You are an advanced natural language processing model. I will provide you with a paragraph and
a list of entities mentioned within it. Your task is to evaluate each entity's importance in
the context of the paragraph on a scale of 1 to 10, where 1 means 'not important' and 10 means
'extremely important.' Here is the paragraph and the list of entities:
```

Input
```
Paragraph: {{paragraph}}

Entities: {{entities}}
```

Output Format Instruction
```
Please return the entities with their corresponding scores in the following format:

{Entity1: Score [1-10],
Entity2: Score [1-10], ...}

Below is an example of an ideal output format.
{"Rotten Tomatoes": 3,
"Brawl in Cell Block 99": 4,
"Vince Vaughn": 3,
"Metacritic": 8,
"Richard Roeper": 7,
"Chicago Sun-Times": 6}
```

Table 10: Prompt template example for the entity filtering task in the proposed **HAWK** framework. The same prompt is provided to multiple agents.

Task Instruction

```
Generate an answer to the question by referencing triplets for context. Please respond short answers.

Context: {{context}}
Triplet (e1, r, e2): {{triple}}
Question: {{question}}
```

Output Format Instruction

```
Please respond according to the format below.

Answer:
```

Table 11: Prompt template example for the answer generation task in the proposed **HAWK** framework.

Task Instruction

```
Please act as an impartial judge and evaluate the quality of the responses provided by two AI assistants to the user question displayed below. You should choose the assistant that follows the user's instructions and answers the user's question better. Your evaluation should consider factors such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of their responses. Begin your evaluation by comparing the two responses and provide a short explanation. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Do not favor certain names of the assistants. Be as objective as possible. After providing your explanation, output your final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]" if assistant B is better, and "[[C]]" for a tie.
```

Input

```
[User Question]
{{question}}

[The Start of Assistant A's Answer]
{{answer_a}}
[The End of Assistant A's Answer]

[The Start of Assistant B's Answer]
{{answer_b}}
[The End of Assistant B's Answer]
```

Table 12: Prompt template for LLM-as-a-Judge method using pair-wise comparison.

B Analysis Details

B.1 Detailed Results of Filtering Method with Multi-agents

Method	w/o filtering	HAWK	# of filtered entities	Rate of decrease (%)
ChatGPT	9.44	6.89	2.55	27.00
Claude3.5	17.66	8.50	9.16	51.89
LLaMA3.1	11.39	6.61	4.79	42.03
Mistral	9.34	6.49	2.85	30.47

Table 13: Number of tail entities before and after applying the filtering process.

B.2 Additional Results on Subcontext Selection Method Variation

Table 14 shows ChatGPT, Claude3.5, LLaMA3.1, and Mistral results (Table 2 has shown the results of Mistral.). Although each model shows slightly different tendencies depending on the selection methods and metrics, improved results compared to random selection are displayed regardless of the method used. This suggests the validity of the composition of carefully selected subcontexts.

Backbone	Method	F1	Rouge-L	BLEU (n=1/2/4)			BERTScore
ChatGPT	HAWK (LLM)	**70.80**	**51.36**	25.58	20.37	13.38	**86.73**
	w/ Random	68.63	51.03	23.53	17.87	10.72	86.33
	w/ String Match	68.16	48.57	23.43	17.29	8.86	85.30
	w/ BM25	68.01	49.02	**28.41**	**22.45**	**15.40**	85.72
Claude3.5	HAWK (LLM)	**64.58**	53.89	21.90	16.29	7.02	74.33
	w/ Random	63.11	5.77	8.90	5.70	2.42	73.95
	w/ String Match	63.25	**54.74**	**26.97**	**19.17**	7.08	**74.79**
	w/ BM25	63.37	54.01	22.05	16.81	**9.06**	74.27
LLaMA3.1	HAWK (LLM)	**73.99**	53.47	19.46	15.98	10.45	**87.31**
	w/ String Match	73.41	**54.21**	18.77	14.89	7.96	87.29
	w/ BM25	71.77	51.27	**21.14**	**18.03**	**12.33**	86.33
Mistral	HAWK (LLM)	**66.27**	**46.52**	**27.05**	**20.05**	**10.03**	**84.97**
	w/ Random	61.43	38.43	22.72	15.59	7.80	82.06
	w/ String Match	65.92	45.96	19.11	13.56	6.21	84.66
	w/ BM25	65.92	41.74	19.81	14.17	6.26	83.74

Table 14: Performance comparison of ChatGPT, Claude3.5, LLaMA3.1, and Mistral with subcontext selection method variations.

B.3 Expansion of Stronger Baselines

Table 15 and 16 present additional baseline results on the NQ and MuSiQue datasets, respectively. We compare Self-RAG (Asai et al., 2024), HippoRAG (Gutiérrez et al., 2025), GraphRAG (Edge et al., 2024), Beam Retrieval (Zhang et al., 2023) with our proposed framework. These results further underscore the effectiveness and generalizability of our proposed HAWK framework, which consistently outperforms recent strong baselines.

Method	F1	Rouge-L	BLEU-1	BLEU-2	BERTScore
Vanilla	56.59	37.93	19.23	14.63	81.78
Self-RAG	42.38	5.70	7.52	5.40	69.35
HippoRAG	36.10	1.29	17.61	4.84	62.70
GraphRAG	44.58	25.06	6.69	3.64	75.09
Beam Retrieval	63.30	43.12	26.80	19.81	83.40
HAWK	**70.80**	**51.36**	**25.58**	**20.37**	**86.73**

Table 15: Additional baseline results on the NQ dataset.

Method	F1	Rouge-L	BLEU-1	BLEU-2	BERTScore
Vanilla	26.52	15.77	3.79	2.36	54.74
Self-RAG	17.26	3.59	1.61	0.84	61.00
HippoRAG	28.73	5.73	1.67	0.95	63.07
GraphRAG	31.48	16.57	5.19	3.32	67.97
Beam Retrieval	25.00	7.10	1.97	0.92	63.77
HAWK	**52.36**	**24.76**	**22.03**	**18.00**	**76.44**

Table 16: Additional baseline results on the MuSiQue dataset.

B.4 Qualitative Results

Table 17 and 18 show several examples of qualitative results. Compared to baseline methods, **HAWK** demonstrates an ability to accurately generate the ground truth (GT) answers. Not only does it effectively utilize sparse information within the context for reasoning, but it also follows well with the instruction—commonly provided in the prompt—to generate concise outputs.

Input Context
(...) After McLaughlin's death in 1914, the company was run briefly by his brother, Samuel McLaughlin. P.D. Saylor and Associates bought the business from the McLaughlin family in 1923 and formed Canada Dry Ginger Ale, Inc., a public company. The "Dry" in the brand's name refers to not being sweet, as in a dry wine. When John J. McLaughlin, who first formulated "Canada Dry Pale Ginger Ale", originally made his new soft drink, it was far less sweet than other ginger ales then available ; as a result, he labelled it "dry". Nylon Studios produced the song used in the Rabbit's "Jack's Farm" commercial featuring Canada Dry Ginger Ale. A Cantonese version of the ad was also produced. Brands with limited availability in the United States include : In 1890, Canadian pharmacist and chemist John J. McLaughlin of Enniskillen, Ontario, after working in a soda factory in Brooklyn, New York, opened a carbonated water plant in Toronto. McLaughlin was the eldest son of Robert McLaughlin, founder of McLaughlin Carriage and McLaughlin Motor Car. In 1904, McLaughlin created "Canada Dry Pale Ginger Ale". Three years later, the drink was appointed to the Viceregal Household of the Governor General of Canada, and the label featuring a beaver atop a map of Canada was replaced with the present Crown and shield. Canada Dry is a brand of soft drinks owned since 2008 by the American Dr Pepper Snapple Group. For over a century, Canada Dry has been known for its ginger ale, though the company also manufactures a number of other soft drinks and mixers. Although Canada Dry originated in Canada, it is now produced in many countries around the globe, including the United States, Iran, Mexico, Colombia, Peru, the Middle East, Europe and Japan. Limited availability flavors are produced in Pennsauken Township, New Jersey, by Pepsi - Cola / National Brand Beverages and are distributed in southern New Jersey, Delaware, southeastern Pennsylvania, and eastern Maryland. At one time, the flavors all had uniquely designed labels ; but now all use the standard Canada Dry crest logo. Norton Simon took an interest in the company in 1964, and it merged with Simon's other holdings, the McCall Corporation and Hunt Foods, to form Norton Simon Inc. Dr Pepper bought Canada Dry from Norton Simon in 1982. In 1984, Dr Pepper was acquired by Forstmann Little Company, and Canada Dry was sold to R.J. Reynolds' Del Monte Foods unit to pay off acquisition debt. RJR Nabisco sold its soft drink business to Cadbury Schweppes in 1986. Today, Canada Dry is owned by Dr Pepper Snapple Group, which was spun off from Cadbury Schweppes in 2008. Canada Dry's popularity as a mixer began during Prohibition, when its flavor helped mask the taste of homemade liquor. In the 1930s, Canada Dry expanded worldwide. From the 1950s onward, the company introduced a larger number of products. (...)

Question: Who makes Canada dry ginger ale with limited availability flavors, Pepsi or Coke?

GT Answer
A: Pepsi

Vanilla Prediction
A: Dr Pepper Snapple Group. [X]

Pruning Prediction
A: Neither Pepsi nor Coke makes Canada Dry Ginger Ale. It is produced by Canada Dry Ginger Ale, Inc., which is a separate company. [X]

Windowing Prediction
A: None of the above. Canada Dry Ginger Ale is made by Canada Dry Ginger Ale, Inc. [X]

Summarization Prediction
A: Dr Pepper Snapple Group (now Keurig Dr Pepper) makes Canada Dry Ginger Ale. [X]

RALM Prediction
A: Neither Pepsi nor Coke makes Canada Dry ginger ale. Canada Dry is its own separate company. [X]

DPR Prediction
A: Neither Pepsi nor Coke make Canada Dry ginger ale. It is owned by the Dr Pepper Snapple Group. [X]

CoT Prediction
A: Neither Pepsi nor Coke make Canada Dry ginger ale. It is owned by the Dr Pepper Snapple Group. [X]

HAWK (LLM) Prediction
A: Pepsi

Table 17: Example of qualitative results of ChatGPT. Blue indicates the correct answer and the basis for the correct answer.

Input Context
(...) The Neutrality Acts were passed by the United States Congress in the 1930s, in response to the growing turmoil in Europe and Asia that eventually led to World War II. (. . .) Following the sinking of the U.S. destroyer Reuben James on October 31, many of the provisions of the Neutrality Acts were repealed on November 17, 1941 : merchant vessels were allowed to be armed and to carry any cargoes to belligerent nations. (. . .) In September 1939, after Germany had invaded Poland, Great Britain and France declared war on Germany. Roosevelt invoked the provisions of the Neutrality Act but came before Congress and lamented that the Neutrality Acts may give passive aid to an aggressor country. Congress was divided. Nye wanted to broaden the embargo, and other isolationists like Vandenberg and Hiram Johnson vowed to fight from hell to breakfast" Roosevelt's desire to loosen the embargo. (. . .) Roosevelt prevailed over the isolationists, and on November 4 the Neutrality Act of 1939 was passed, allowing for arms trade with belligerent nations (Great Britain and France) on a cash - and - carry basis, thus in effect ending the arms embargo. (...)

Question: When did the US declared neutrality in WW2?

GT Answer
1939

Vanilla Prediction
A: The US did not formally declare neutrality in World War II. Instead, the Neutrality Acts of the 1930s attempted to keep the US out of foreign conflicts. However, these acts were gradually weakened and ultimately repealed in November 1941, shortly before the US entered the war after the Pearl Harbor attack on December 7, 1941. [X]

Pruning Prediction
A: The US did not officially declare neutrality in World War II. Instead, the Neutrality Acts of the 1930s attempted to keep the US out of foreign conflicts. However, these acts were gradually weakened and ultimately repealed in November 1941, shortly before the US entered the war after the Pearl Harbor attack on December 7, 1941. [X]

Windowing Prediction
A: The US initially maintained neutrality through the Neutrality Acts in the 1930s, but gradually moved away from strict neutrality. By November 17, 1941, most Neutrality Act provisions were repealed after German submarine attacks, effectively ending US neutrality before the formal declaration of war after Pearl Harbor on December 8, 1941. [X]

Summarization Prediction
A: The US did not formally declare neutrality in World War II. Instead, the Neutrality Acts of the 1930s attempted to keep the US out of foreign conflicts. These Acts were gradually repealed, with the final provisions being removed in 1941 after the Pearl Harbor attack, which led to the US entering the war. [X]

RALM Prediction
A: The United States did not declare neutrality in World War II. Instead, the Neutrality Acts of the 1930s were gradually weakened and ultimately repealed in November 1941, shortly before the US officially entered the war after the Pearl Harbor attack on December 7, 1941. [X]

DPR Prediction
A: The context does not specify an exact date of US neutrality declaration. However, the passage indicates the US maintained a neutrality policy prior to the Lend-Lease Act of March 1941, which effectively ended strict neutrality. The US formally entered World War II after the Pearl Harbor attack on December 7, 1941, when it declared war on Japan on December 8, 1941. [X]

CoT Prediction
A: The US initially declared neutrality in the mid-1930s through the Neutrality Acts of 1935, 1936, and 1937, which restricted US involvement in international conflicts. However, this neutrality effectively ended with the Lend-Lease Act of March 1941, which allowed the US to provide war materials to allied nations. []

HAWK (LLM) Prediction
A: 1939

Table 18: Example of qualitative results of Claude3.5. Blue indicates the correct answer and the basis for the correct answer.

Table 19 shows an example of step-by-step generation in the **HAWK** framework, applied to the Claude3.5 model. Through each step, demonstrating examples of extracting the important head entities to answer a given question, corresponding subcontext-based tail entities extraction, and efficient key entity extraction via multi-agent-based entity filtering. Additionally, examples of constructed triplets are provided. By progressing through these steps, **HAWK** mitigates information sparsity in long contexts and enables appropriate responses to query questions.

Input Context (...) The Neutrality Acts were passed by the United States Congress in the 1930s, in response to the growing turmoil in Europe and Asia that eventually led to World War II. (. . .) Following the sinking of the U.S. destroyer Reuben James on October 31, many of the provisions of the Neutrality Acts were repealed on November 17, 1941 : merchant vessels were allowed to be armed and to carry any cargoes to belligerent nations. (. . .) In September 1939, after Germany had invaded Poland, Great Britain and France declared war on Germany. Roosevelt invoked the provisions of the Neutrality Act but came before Congress and lamented that the Neutrality Acts may give passive aid to an aggressor country. Congress was divided. Nye wanted to broaden the embargo, and other isolationists like Vandenberg and Hiram Johnson vowed to fight from hell to breakfast" Roosevelt's desire to loosen the embargo. (. . .) Roosevelt prevailed over the isolationists, and on November 4 the Neutrality Act of 1939 was passed, allowing for arms trade with belligerent nations (Great Britain and France) on a cash - and - carry basis, thus in effect ending the arms embargo. (...)
Question: When did the US declared neutrality in WW2?
GT Answer 1939
Query-based Entity Extraction h_1: US, h_2: World War 2, h_3: neutrality **Entity Extraction for Subcontext** "US": [(LOCATION, United States), (ORGANIZATION, United States Congress), (LOCATION, Europe), (LOCATION, Asia), (EVENT, World War II), (LOCATION, Britain), (LOCATION, France), (LOCATION, Nazi Germany), (ORGANIZATION, Pearl Harbor), (PERSON, Japanese)] "World War 2": [(Neutrality Acts, Law/Policy), (United States Congress, Organization), (Europe, Location), (Asia, Location), (World War II, Historical Event)] "neutrality": [(LAW, Neutrality Acts), (LOCATION, United States), (ORGANIZATION, Congress), (DATE, 1930s), (LOCATION, Europe), (LOCATION, Asia), (EVENT, World War II), (LAW, Neutrality Act of 1939), (LAW, Neutrality Act of 1937), (LOCATION, Britain), (LOCATION, France), (LOCATION, Nazi Germany)] **Multi-agents-based Entity Filtering** "US": [(LOCATION, United States), (ORGANIZATION, United States Congress), (LOCATION, Europe), (EVENT, World War II), (LOCATION, Britain), (LOCATION, France), (LOCATION, Nazi Germany), (ORGANIZATION, Pearl Harbor)] "World War 2": [(Neutrality Acts, Law/Policy), (United States Congress, Organization), (Europe, Location), (Asia, Location), (World War II, Historical Event)], "neutrality": [(LAW, Neutrality Acts), (LOCATION, United States), (DATE, 1930s), (EVENT, World War II), (LAW, Neutrality Act of 1939), (LAW, Neutrality Act of 1937), (LOCATION, Nazi Germany)] **Constructed Triplets** (US, same_entity, United States), (US, located_near, Europe), (US, INVOLVED_IN, World War II), (US, support, Britain), (...), (World War 2, located_in, Europe), (World War 2, located_in, Asia), (World War 2, alias, World War II), (neutrality, is_a, Neutrality Acts), (neutrality, ORIGIN, United States), (neutrality, legislative_action, Congress), (neutrality, time_period, 1930s), (...) **HAWK (LLM) Prediction** A: 1939

Table 19: An example of step-by-step generation in the **HAWK** framework, applied to the Claude3.5 model.

Table 20 shows the LLM-as-a-Judge evaluation results for Claude3.5 prediction results. In the experiment, the LLM is provided with predictions from **HAWK** and other baseline methods and is tasked with determining which response was better, along with a justification for its choice. In order to reduce position bias, the actual experiment was performed twice by switching A and B. **HAWK**'s method is evaluated better because it provides more concise and accurate answers to queries.

Input Context
Question: What type of government is oceania in 1984?

HAWK (LLM) Prediction
A: Totalitarian government,

Vanilla Prediction
A: Totalitarian oligarchy under the ideology of English Socialism (Ingsoc), ruled by the Inner Party with Big Brother as the symbolic leader. The government maintains control through constant surveillance, manipulation of language, perpetual war, and suppression of individual thought.,

LLM Judgment
"I evaluated the responses provided by the two AI assistants based on the user's question about who makes Canada Dry Ginger Ale, Pepsi, or Coke.A correctly states that the Dr Pepper Snapple Group is the owner of Canada Dry Ginger Ale, which is the most relevant and accurate response to the user's question. This information is crucial in understanding the current ownership of the brand.B, on the other hand, mentions that Canada Dry was created by John J. McLaughlin and is currently owned by Keurig Dr Pepper. While the historical context provided is interesting, it does not directly address the user's question about whether Pepsi or Coke makes Canada Dry., Assistant A has provided a more appropriate and precise response to the user's question, making it the better answer overall.[[A]]"

Summarization Prediction
A: Totalitarian regime under English Socialism (Ingsoc)", "key_characteristics": ["Ruled by Inner Party", "Symbolically led by Big Brother", "Controlled through propaganda", "Uses Newspeak to limit rebellious thoughts", "Maintains power through strict social policies"]

LLM Judgment
"Assistant A provides a concise and direct response to the user question, clearly stating that Dr Pepper Snapple Group makes Canada Dry Ginger Ale. On the other hand, Assistant B not only answers the question but also provides additional context about the ownership of Dr Pepper Snapple Group by Keurig Dr Pepper and the limited availability flavors produced under Pepsi-Cola/National Brand Beverages. While the additional information is informative, it goes beyond what was asked in the user question, which could be seen as a minor deviation from the main focus. , both responses are accurate and relevant to the user question. However, Assistant A's response is more precise, on point, and directly addresses the question without adding extra details. Therefore, **[[A]]** is better."

RALM Prediction
A: Totalitarian socialist government", "key_characteristics": ["Ruled by the Inner Party", "Ideology of Ingsoc (English Socialism)", "Symbolic leadership of Big Brother", "Extreme social stratification", "Perpetual surveillance and control", "Manipulation through propaganda and perpetual war"], "political_structure": "ruling_class": "Inner Party", "middle_class": "Outer Party", "lower_class": "Proles", "leadership_symbol": "Big Brother" , "governance_methods": ["Thought control", "Language manipulation (Newspeak)", "Constant historical revisionism", "Suppression of individual freedom"]

LLM Judgment
"I evaluated the responses provided by the two AI assistants based on the user's question about who makes Canada Dry Ginger Ale, Pepsi, or Coke.A correctly states that the Dr Pepper Snapple Group is the owner of Canada Dry Ginger Ale, which is the most relevant and accurate response to the user's question. This information is crucial in understanding the current ownership of the brand.B, on the other hand, mentions that Canada Dry was created by John J. McLaughlin and is currently owned by Keurig Dr Pepper. While the historical context provided is interesting, it does not directly address the user's question about whether Pepsi or Coke makes Canada Dry., Assistant A has provided a more appropriate and precise response to the user's question, making it the better answer overall.[[A]]"

CoT Prediction
A: Totalitarian socialist government", "key_characteristics": ["Ruled by the Inner Party", "Ideology of Ingsoc (English Socialism)", "Symbolic leadership of Big Brother", "Extreme social stratification", "Perpetual surveillance and control", "Manipulation through propaganda and perpetual war"], "political_structure": "ruling_class": "Inner Party", "middle_class": "Outer Party", "lower_class": "Proles", "leadership_symbol": "Big Brother" , "governance_methods": ["Thought control", "Language manipulation (Newspeak)", "Constant historical revisionism", "Suppression of individual freedom"]

LLM Judgment
"When comparing the responses provided by Assistant A and Assistant B, Assistant A gives a more concise and direct answer to the user's question. Assistant A simply states that the company that makes Canada Dry Ginger Ale is the Dr Pepper Snapple Group. On the other hand, Assistant B, while providing valuable context and details about the ownership history of Canada Dry, takes a longer route to answer by mentioning that neither Pepsi nor Coca-Cola owns the brand, but rather the Dr Pepper Snapple Group. , based on the user's question and the need for a clear and straightforward response, Assistant A's answer is better as it directly addresses the query without unnecessary elaboration. , the evaluation leads to the following conclusion:[[A]]"

Table 20: LLM-as-a-Judge evaluation results for Claude3.5 prediction results. In order to reduce position bias, the actual experiment was performed twice by switching A and B.