CSTree-SRI: Introspection-Driven Cognitive Semantic Tree for Multi-Turn Question Answering over Extra-Long Contexts

Zhaowen Wang[1] Xiang Wei[1] Kangshao Du[1] Yiting Zhang[1]
Libo Qin[1] Yingjie Xia[2,3] Li Kuang[1*]
[1]School of Computer Science and Engineering, Central South University
[2]Micro-Electronics Research Institute, Hangzhou Dianzi University
[3]College of Computer Science and Technology, Zhejiang University
{wizxech, 8205220302, 8210220928, 8209220418}@csu.edu.cn
m15840540712@163.com, xiayingjie@zju.edu.cn, kuangli@csu.edu.cn

Abstract

Large Language Models (LLMs) have achieved remarkable success in natural language processing (NLP), particularly in single-turn question answering (QA) on short-text. However, their performance significantly declines when applied to multi-turn QA over extra-long context (ELC), as they struggle to capture the logical correlations across multiple chunks of ELC and maintain the coherence of multi-turn Questions. To address the challenges, we propose the CSTree-SRI framework(**C**ognitive **S**emantic **Tree** through **S**ummarization, **R**etrieval, and **I**ntrospection). CSTree-SRI dynamically constructs the CSTree to preserve logical coherence within ELC through hierarchical synthesis and introspective validation. Then a logic-driven traversal strategy on CSTree is designed to provide efficient information retrieval for question answering. Additionally, we construct a suite of multi-turn QA datasets and an evaluation benchmark tailored for ELC tasks, and comprehensive experiments demonstrate the framework's superiority in addressing the challenges of multi-turn QA over ELC.

1 Introduction

The rapid proliferation of digital information has intensified the demand for understanding extra-long context (ELC) in multi-turn question answering (MTQA) with LLM. ELC involves both single documents (e.g., legal contracts) and cross-document synthesis (e.g., academic literature reviews) that exceed the context window of LLM (Bai et al., 2024b). MTQA over ELC scenarios further complicates the problem. Users often engage in iterative questioning, such as consulting legal clauses or exploring academic topics. As shown in Fig. 1, such kind of tasks require capturing the logical correlation among multiple chunks of ELC, the coherence among multi-turn QA, as well as the alignment between questions and partially overlapping retrieved segments (Zhu et al., 2023), thus placing higher demands on LLMs' ability to precisely and efficiently extract key information in ELC.

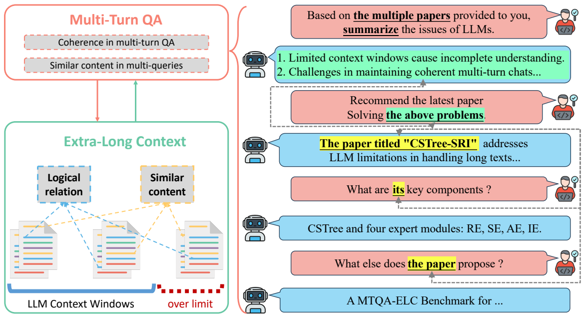

Figure 1: **An Example of MTQA over ELC.** The 1st question of summarizing multiple papers involves the correlation among multiple chunks of ELC, the 2nd question of recommending the latest one involves the coherence among multiple questions, and the following two questions of the paper details involve partially overlapping retrieved segments. We propose a cognitive semantic tree to capture logical relationships and coherence across MTQA over ELC.

There are mainly two kinds of approaches in processing ELC (Huang et al., 2023): (1) modifying LLM's architecture to extend the context window, e.g., optimizing attention mechanisms (Chen et al., 2023b), introducing recurrence (Borgeaud et al., 2022), or modifying positional encoding (Su et al., 2024); (2) employing external tools (e.g., RAG and Agents) to assist LLM in efficient retrieval and information processing (Topsakal and Akinci, 2023). These approaches primarily focus on single-turn tasks, lacking effective mechanisms for maintaining coherence across multi-turn interactions.

Research on multi-turn conversation abilities has largely been confined to short-text domains, where evaluation benchmarks have been well-established. Traditional methods on MTQA simply concatenate historical turns, where context is utilized inefficiently (Zhang et al., 2018a), and noise may be introduced. Moreover, when the context window is

*Corresponding author.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 26502–26525
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

exceeded, truncation mechanisms may discard critical information, adversely affecting the model's reasoning and comprehension.

In summary, current research on MTQA over ELC exhibits three key limitations: **(1) logical fragmentation:** Existing context window extension methods address length constraints but fail to preserve inter-document, across-MTQA relationships (Liao et al., 2024). **(2) noise accumulation:** Concatenating multi-turn inputs causes noise accumulation from redundant information, and increases computational costs (Zhang et al., 2018b). **(3) evaluation gaps:** Existing benchmarks focus on short-text MTQA (Kwan et al., 2024; Wang et al., 2023; Bai et al., 2024a), lacking datasets and metrics for evaluating MTQA reasoning performance over ELC.

To address these challenges, we propose CSTree-SRI(**C**ognitive **S**emantic **Tree** through **S**ummarization, **R**etrieval, and **I**ntrospection), a framework for multi-turn QA over extra-long context (ELC), which dynamically constructs a hierarchical Cognitive Semantic Tree (CSTree) to organize ELC into document/paragraph/sentence nodes, preserving logical coherence for efficient retrieval. CSTree-SRI integrates four expert modules: (1) Retrieval Expert (RE), for relevant segments filtering; (2) Summary Expert (SE), which generates hierarchical summaries; (3) Introspection Expert (IE), which dynamically makes decisions on retrieval and response optimization; and (4) Answer Expert (AE), produces final responses. **To address challenge 1**, CSTree-SRI first dynamically builds the CSTree through hierarchical synthesis and introspective validation by a collaboration of RE, SE, and IE. **To address challenge 2**, it then introduces a logic-driven hierarchical traversal strategy on CSTree to retrieve relevant information for the next question by RE and IE. Subsequently, the framework iteratively optimizes responses through collaboration between AE and IE, ensuring both relevance and grounding in the retrieved information. **To address challenge 3**, we construct an MTQA-ELC benchmark and assess LLM performance in extra-long context QA tasks. Our contributions transcend prior work in three dimensions:

(1) Framework Innovation: CSTree-SRI is the first attempt to construct and utilize the introspection-driven CSTree through the collaboration of multiple expert modules for understanding ELC in MTQA precisely and efficiently.

(2) Benchmark Rigor: We introduce the first MTQA-ELC benchmark, containing over 500 articles spanning 391k words, nearly 4k groups of correlated questions, and new metrics for reasoning time, accuracy, and LLM-human gaps.

(3) Empirical Superiority: On tasks with 256k+ tokens, CSTree-SRI improves multi-turn QA performance by an average of 21.48%, reduces inference time by 41.11% (ETScore) compared to RAG/Agent solutions while improving answer accuracy by 44.17%.

2 Related Work

2.1 Long-Text Processing in LLMs

Current challenges in enhancing the long-text processing capabilities of LLMs include (Huang et al., 2023): quadratic complexity in attention computation, the lack of context memory mechanisms, and limitations on the maximum length of training samples. Existing approaches can be broadly categorized into two classes:

Architectural Optimization. Existing approaches to enhance Transformer-based LLMs' long-text processing capabilities focus on the following architectural optimizations: (1) Attention mechanism refinement improves computational efficiency through blockwise processing or hierarchical attention (Qiu et al., 2019; Chen et al., 2023b; Yang et al., 2016), yet often sacrifices global contextual awareness; (2) Recurrent memory augmentation integrates external memory databases to preserve long-term dependencies (Borgeaud et al., 2022; Tworkowski et al., 2024), but struggles with precise memory retrieval; (3) Positional encoding extension employs rotary operations or NTK-aware scaling to expand context windows (Su et al., 2024; Chen et al., 2023a; Peng and Quesnelle, 2023), but they require additional adjustments and optimizations, potentially increasing training difficulty.

Unlike existing work focused on specific Transformer optimizations, we propose CSTree-SRI that enhances LLMs' multi-turn QA over ELC beyond architectural-level improvements.

External Tool Augmentation. These approaches employ LLMs as black-box processors combined with external mechanisms: (1) Multi-agent collaboration frameworks delegate long-text processing through role specialization and interaction protocols (Zhao et al., 2024), though coordination overhead increases complexity; (2) Attention modification techniques like LongHeads adapt attention patterns for extended contexts without architectural changes (Lu et al., 2024), but lack dynamic reasoning adaptation; (3) Retrieval-Augmented Generation (RAG) enhances inputs through external knowledge bases (Gao et al., 2023), with recent improvements incorporating LLM-guided retrieval evaluation (Li et al., 2023) and reflection mechanisms (Asai et al., 2023). While effectively circumventing context window limitations, such methods often underutilize LLMs' native reasoning capacities for complex textual analysis.

Our work combines chunked retrieval and reflective analysis, leveraging multi-module experts and the Cognitive Semantic Tree to extract and maintain logical information in ELC. This enables efficient information filtering and organization, offering a new pathway for MTQA over ELC.

2.2 Benchmarks for MTQA on Long-Context

Evaluating long-context models is challenging due to the inherent difficulty of collecting and analyzing long texts (Li et al., 2024). Bai et al. (2024b) introduced the LongBench benchmark, comprising six major task categories and 21 tasks, covering key long-text application scenarios. An et al. (2024) proposed the L-Eval benchmark, which includes long documents from domains such as law, finance, academic papers, novels, and conferences, along with various tasks. However, these benchmarks primarily evaluate single-turn QA tasks and lack assessments for MTQA tasks. Zheng et al. (2023) developed MT-Bench, a dataset of 80 multi-turn questions, but each dialogue consists of only two turns. Kwan et al. (2024) increased the number of turns, proposing MT-Eval, which includes multiple task types within a single dialogue to evaluate LLMs' comprehensive multi-turn dialogue capabilities. However, these benchmarks involve relatively short texts and do not address ELC.

In summary, these works lack evaluations of LLMs' reliability and efficiency in MTQA over ELC, and the distinction between evaluating models and augmenting long-text processing with external tools remains underexplored. In contrast, our work evaluates mainstream long-text LLMs and external tools (e.g., RAG, Agents) in MTQA over ELC, addressing gaps in existing research.

3 CSTree-SRI

The input to CSTree-SRI consists of a text sequence $X = \{x_1, x_2, \ldots, x_l\}$, which can be a single long document or a collection of documents, and a sequence of logically dependent queries $Q = \{q_1, q_2, \ldots, q_m\}$ across multiple rounds. The framework aims to generate answers for these queries based on the input X. To handle this, CSTree-SRI initially segments the input text X into chunks of a predefined size $sz = 512$, resulting in $X = \{C_1, C_2, \ldots, C_t\}$ where $t = \lceil l/sz \rceil$. These chunks act as the fundamental processing units, enabling effective multi-turn QA (MTQA) over extra-long context (ELC) by maintaining and leveraging historical information throughout the queries.

3.1 Framework Components

The CSTree-SRI framework comprises a Cognitive Semantic Tree (CSTree) and four expert modules.

The CSTree is a three-layer tree structure where the nodes are classified into document-level nodes, paragraph-level nodes, and sentence-level nodes. Each node contains a summary or raw text, with edges between the nodes of various layers formed due to their common logical relationships.

The four Expert modules include: (1) A Retrieval Expert (RE) that filters out relevant text segments to reduce noise. (2) A Summary Expert (SE) generates concise summaries after each QA turn to maintain logical consistency. (3) An Answer Expert (AE) that produces final responses. (4) An Introspection Expert (IE) that dynamically refines retrieval precision. The IE module will conduct introspection from two aspects: retrieval precision and response accuracy, with specific introspection questions detailed in Table 1.

Specifically, for each query q_i, CSTree-SRI performs two core operations: **(1) Dynamic Structure Construction** of the tree through collabora-

Type	Specific Question
Relative	Are the retrieved text chunks relevant to the current query q_i?
NodeRetr	For the summarized information of a node, is further retrieval necessary?
ExtraRetr	Is the retrieved information from the CSTree sufficient, or is further retrieval from the original text needed?
Support	Can the retrieved information support the AE's answer?
Useful	Does the AE's answer effectively address q_i?

Table 1: Introspection Questions for the IE Module (See Appendix C.1 for Detailed Prompts)

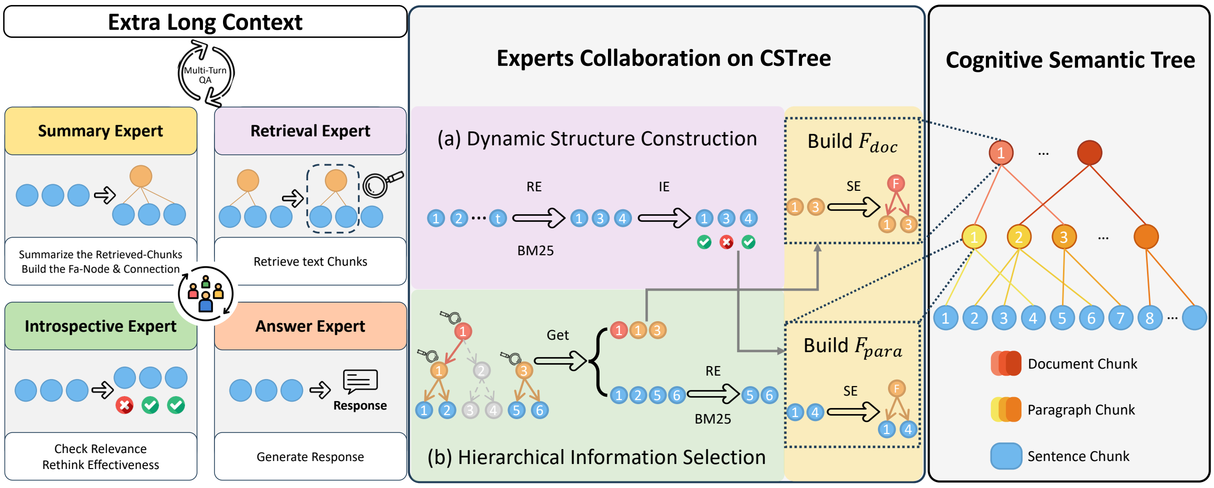

Figure 2: Expert collaborative interaction process of CSTree-SRI. The different shades of the same color in CSTree represent the step-by-step construction of the CSTree across different QA rounds.

tion among the RE, IE, and SE modules, and **(2) Hierarchical Information Selection** on the tree via collaboration between the RE and IE modules. After retrieving relevant information blocks, responses are refined through iterative optimization between the AE and IE modules to ensure enhanced answer precision. Appendix C.2 contains specific prompts for each module. The following sections describe how these modules interact collaboratively with the CSTree during each QA round q_i.

3.2 Dynamic Structure Construction

Inspired by the hierarchical structure of human reading notes (paragraph-chapter-book), we propose a dynamic CSTree construction that mimics cognitive processes through introspection-driven hierarchical synthesis. The RE, IE, and SE modules collaborate to implement the "structured note-taking" approach. They retrieve context segments, validate logical coherence through introspection, and synthesize summaries at paragraph and document levels. This process transforms ELC into navigable information structures. Below, we detail the technical implementation of constructing paragraph-level and document-level nodes.

Para-Level Node Construction. As shown in Fig. 2(a), our framework combines flat retrieval with introspective validation for paragraph-level construction, known as the flat information retrieval strategy. The RE first retrieves candidate text chunks X using the BM25 algorithm. Meanwhile, the IE assesses whether the retrieved chunks truly represent the key information C_{key} relevant to the query q_i, effectively addressing the "keyword bias" commonly found in traditional sparse retrieval methods. This two-stage filtering process—merging statistical relevance with semantic introspection—ensures that only logically coherent fragments proceed to the synthesis phase. The SE then dynamically creates a hierarchical parent node F_{para} by abstracting the relationships among the C_{key} nodes, thereby establishing explicit edges to maintain content associations and provenance.

Doc-Level Node Construction. The framework constructs C_{doc} through a ratio-controlled triggering mechanism. When C_{para} with logical relationships are identified during CSTree traversal, a predefined 1:3 doc-to-para ratio threshold governs the construction process. This ensures that the number of C_{doc} never exceeds one-third of the C_{para}, preventing structural redundancy. The proportional constraint activates the SE module only when sufficient C_{para} nodes exist. This activation asks the SE module to generate the doc-level parent node F_{doc} by summarizing relationships across paragraphs.

This hierarchical summarization structure enables dynamic CSTree evolution through progressive QA interactions. Our "structured note-taking" approach preserves critical relationships within the ELC, enhancing QA accuracy. Additionally, the CSTree improves reasoning efficiency by maintaining the logical coherence of the text, which accelerates information retrieval. These advantages are validated in our ablation studies.

3.3 Hierarchical Information Selection

Unlike conventional tree traversal methods with fixed depth-first or breadth-first strategies, our approach introduces a logic-driven hierarchical traversal strategy where the IE module evaluates node relevance at each hierarchy level. The RE and IE modules collaborate to strategically navigate the

CSTree, balancing retrieval depth with computational efficiency to address ELC challenges. This logic-driven approach mirrors human top-down comprehension, starting with high-level summaries and drilling down to details as needed. After retrieval, CSTree-SRI uses a sufficiency validation mechanism to ensure the retrieved information meets query requirements. The hierarchical information selection process is detailed below.

Logic-Driven Hierarchical Traversal Strategy As shown in Fig. 2(b), we have implemented a dynamic hierarchical traversal strategy that adjusts exploration depth through semantic introspection. The process begins with the IE module analyzing the summary information of each non-leaf node, and a dynamic continuation probability $\phi(C) = IE(q_i, C_l, NodeRetr)$ is calculated for each node chunk C. The hierarchical retrieval automatically terminates at level l when $\phi(C_l) < \tau$, implementing principled depth control that prevents over-retrieval while maintaining query relevance.

For the retrieval results across the entire CSTree, paragraph-level nodes C_{para} will have their corresponding document-level parent nodes constructed as outlined in Section 3.2. For sentence-level nodes C_{sen}, to prevent excessive information retrieval, the BM25 algorithm is employed to efficiently filter the top-K most relevant nodes, which are then used as the retrieved text chunks from the CSTree. The entire hierarchical information screening process can be formalized as follows:

$$\phi(C) \geq \tau \Rightarrow Select(Child)$$

$$Select(C_{doc}) \Rightarrow Select(C_{para}) \Rightarrow Select(C_{sen})$$

$$Chunk_{tree} = BM25(q_i, C_{sen}, topk)$$

Here, $A \Rightarrow B$ indicates that operation B is performed based on the result of operation A; $Select(\cdot)$ represents the selection operation, where the child nodes $Child$ of the selected node become the target of the next layer of retrieval; $Chunk_{tree}$ refers to the text chunks retrieved from the CSTree; and $BM25(\cdot)$ denotes the retrieval operation using the BM25 algorithm.

Sufficiency Validation Mechanism After completing the CSTree retrieval, the IE module introspects the *ExtraRetr* to evaluate whether the retrieved text chunks are sufficient to answer the query q_i. If necessary, additional relevant text chunks are retrieved from the extra-long context using the flat information retrieval strategy described in Section 3.2. Finally, all retrieved text chunks are consolidated and provided to the AE module to generate the final response.

3.4 Iterative Response Optimization

This step represents the final stage of the framework, synthesizing text chunks retrieved through the processes detailed in Sections 3.2 and 3.3. Through iterative collaboration between the IE and AE modules, the response to query q_i is refined.

The IE module evaluates the AE's output across two critical dimensions: <Support>, which ensures that the response is grounded in the retrieved text chunks, and <Useful>, which assesses the response's relevance to q_i. This dual-focused evaluation facilitates iterative optimization, ultimately leading to the final answer, as formalized below.

$$Resp = AE(q_i, Chunk_{tree} + Chunk_{flat})$$

$$IE(q_i, Resp, Support\&Useful) \Rightarrow Iter(Resp)$$

$$Resp^* = Iter(Resp)$$

Here, $AE(\cdot)$ denotes the generation of a response by the AE, and $Chunk_{flat}$ represents the text chunks retrieved using the flat information retrieval method. $IE(Support\&Useful)$ indicates the IE module performing the <Support> and <Useful> introspection. $Iter(\cdot)$ represents the iterative process of generating and refining responses through the AE and IE. $Resp^*$ refers to the updated response generated in a new iteration.

4 MTQA-ELC Benchmark

Current benchmarks for evaluating LLMs primarily focus on language modeling and generation tasks. However, these benchmarks may not fully capture the models' abilities to handle complex, multi-turn question-answering tasks, particularly with extra-long contexts. To address this gap, we have developed a benchmark specifically designed to assess LLM performance in information retrieval, key information extraction, and logical reasoning—skills

Benchmark	#words in Text Max.	#words in Text Avg.	#Turns Max.	#Turns Avg.
LongBench(QA task)	18409	8640	1	1
L-Eval(QA task)	26918	9133	1	1
MT-Bench	330	68	2	2
MT-Eval	2574	760	12	7
MT-Bench-101	817	202	**323**	67
MTQA-ELC (Ours)	**217273**	**217264**	100	**100**

Table 2: Data Statistics. Detailed data sources are provided in Appendix A.1.

that are crucial for real-world applications involving long-text processing.

4.1 Data Construction

Table 2 shows the key statistics of MTQA-ELC. Our dataset consists of reading comprehension passages from major exams such as the NMET, CET, PGEE, and TPO. Each passage is carefully divided into paragraphs, with unique identifiers added at the beginning and end of each segment to indicate the article and paragraph. This structure enables explicit tracking of relationships between paragraphs when multiple segments from different articles are concatenated into an ELC. These identifiers allow benchmarks to evaluate LLMs' ability to process and integrate information across paragraphs.

To further assess the integration and reasoning capabilities of models, we generated multi-turn question sets based on texts of varying lengths (32k, 64k, 128k, 256k). For fair evaluation, we randomize paragraph order, shuffle options, and compare model performance with human test-taker scores, as detailed in Appendix A.2.

4.2 Task Set

Reading comprehension tasks assess various cognitive skills. To evaluate LLMs' capabilities in multi-turn QA over ELC, we categorize tasks based on required abilities, including paragraph retrieval, information integration, detail/main idea comprehension, and logical reasoning.

Tasks are divided into four types: Detail Understanding (DU), Semantic & Reference (SR), Main Idea (MI), and Inference & Judgment (IJ). The former two tasks focus on single-paragraph retrieval and understanding, while the latter two require integrating information across multiple paragraphs to grasp the main idea or perform complex reasoning. For all tasks, the input consists of an ELC and a set of questions with multiple choices, and the output is the correct choice. Appendix D contains examples of various evaluation tasks.

4.3 Metrics

Our Benchmark evaluates LLMs using three key metrics: Accuracy (ACC), Effective Time Score (ETScore), and Human-Adjusted Overall Score. Accuracy is a commonly used metric, while ETScore and Human-Adjusted Overall Score are newly proposed metrics in our Benchmark.

ETScore measures LLMs' reasoning time and their ability to answer correctly within a specific time frame, addressing the limitation of traditional accuracy metrics in capturing time efficiency. The Human-Adjusted Overall Score compares LLM performance to human test-takers, highlighting strengths and weaknesses relative to people.

Accuracy is calculated as:

$$ACC = \sum_{i=1}^{N} \frac{Check(O_i, A_i)}{N} \times 100\% \tag{1}$$

Here, N is the total number of questions, and $Check(O_i, A_i)$ verifies if the model's output O_i matches the correct answer A_i.

ETScore's calculation formulas are as follows:

$$AvgTime = \frac{\sum_{i=1}^{M} (EndTime_i - StartTime_i)}{M} \tag{2}$$

$$ETScore = Acc \times \frac{K}{1 + \beta \times AvgTime} \tag{3}$$

Here, M is the number of test papers, $AvgTime$ is the average reasoning time per question, β controls time sensitivity, and K scales the score. We set $\beta = 0.002$ and $K = 100$, with higher $ETScore$ indicating better performance.

Human-Adjusted Overall Score accounts for task difficulty by incorporating test-taker accuracy:

$$Overall = \frac{\sum_{i=1}^{N} W_i}{N} \tag{4}$$

$$W_i = \frac{\sum_{j=1}^{Q_i} f(p_{ij}, a_{ij}, k_i)}{Q_i} \times 100 \tag{5}$$

$$f(p, a, k) = e^{0.5 \cdot k} + ae^{k(0.5-p)a} \tag{6}$$

$$a_{ij} = \begin{cases} 1, & Resp_{ij} = Answer_{ij} \\ -1, & Resp_{ij} \neq Answer_{ij} \end{cases} \tag{7}$$

Here, Q_i is the number of questions in test paper i, p_{ij} is the human accuracy for question j, and a_{ij} indicates correctness (1 for correct, -1 for incorrect). Hyperparameter k adjusts sensitivity: higher difficulty (p_{ij} low) increases rewards for correct answers and softens penalties for mistakes, while low-difficulty errors incur heavier penalties.

5 Experiments

In this section, we evaluate the performance of the CSTree-SRI framework on both single-turn and multi-turn QA tasks. For single-turn QA, we use the LongBench benchmarks. For multi-turn QA, we conduct experiments on the MTQA-ELC to assess the capabilities of various long-text LLMs over ELC. We also compare CSTree-SRI with mainstream RAG and Agent methods, demonstrating its superior performance in MTQA. Additionally, ablation studies validate the contributions of individual modules within the CSTree-SRI framework.

Model/Framework	Single-Doc QA NQA	Qspr.	MulFi	Avg.	Multi-Doc QA HQA	WMQA	Musq.	Avg.
Llama-2-7B-chat †	18.7	19.2	36.8	24.9	25.4	32.8	9.4	22.6
-LongHeads w/NTK init †	16.87	**30.32**	38.59	**28.59**	**36.04**	26.72	10.21	24.32
-LongLora	17.36	28.97	38.37	28.30	34.81	32.57	12.72	26.70
-CSTree-SRI	**19.42**	23.34	**41.25**	28.00	35.73	**35.21**	**21.23**	**30.72**

Table 3: The results of different methods based on the Llama-2-7B-chat model on LongBench. † means the data are sourced from the LongBench and LongHeads papers

5.1 Experiments Setting

All evaluations were conducted with `float16` precision on 4 Nvidia V100-32G GPUs. Configuration details for each benchmark are described below.

LongBench We evaluated six English datasets from LongBench: NarrativeQA, Qasper, MultiFieldQA, HotpotQA, 2WikiMQA, and Musique, spanning single- and multi-document QA tasks. The CSTree-SRI framework used Llama-2-7B-chat as the AE with gpt-3.5-turbo for SE/IE modules. Baseline included: 1) vanilla Llama-2-7B-chat, 2) LongLoRA (Chen et al., 2023b, attention-optimized fine-tuning), and 3) LongHeads (Lu et al., 2024, attention head selection).

MTQA-ELC We conducted 100-round multi-turn QA sessions. Vanilla LLMs processed texts by concatenating the first and last halves of their context windows due to inherent context window limitations. We evaluated three open-source LLMs with 128K context windows (GLM-4-9B-Chat, Llama-3.1-8B-Instruct, Qwen2.5-7B-Instruct), all locally deployed. Additionally, we tested three API-accessed models: gpt-4o-mini (128K), DeepSeek-chat (64K), and gpt-3.5-turbo (16K). External tools compared included RAG (using jina-embeddings-v2-base-en with cosine similarity), LongAgent (with gpt-4o-mini), and CSTree-SRI (with gpt-4o-mini for SE/IE), all using Llama-3.1-8B-Instruct as the QA module. Appendix C.3 contains prompts for evaluation tasks.

5.2 Single-Turn QA Evaluation

Table 3 compares our method with LongHeads and LongLora on single-turn QA tasks within LongBench. Our method achieves performance that is comparable to the baseline in single-document QA. However, in multi-document QA, CSTree-SRI significantly outperforms the others in average scores, demonstrating its effectiveness in handling more complex long-text QA tasks. This improvement is due to our framework's enhanced ability to capture the logical relationships within lengthy and intricate texts.

To further validate the generalizability of our method across different models, we conducted additional experiments on the L-Eval benchmark. The results demonstrate consistent performance improvements, as detailed in Appendix B.1.

5.3 Multi-Turn QA Evaluation

Table 4 presents the performance comparison of Long-Context LLMs and external tool-enhanced methods on MTQA tasks across different context lengths. For Long-Context LLM with a 128K native context window, both ACC and ETS decline significantly when handling texts beyond this limit (128K, 256K) compared to shorter contexts (32K, 64K), highlighting their constraints in extra-long context processing.

Comparing the overall performance of CSTree-SRI (with Qwen-2.5-7B-Instruct as AE) to gpt-4o-mini and deepseek-chat in Table 4, our method achieves the highest Overall score (213.78) while maintaining consistently high ACC and ETS across different context lengths. This highlights CSTree-SRI's effectiveness in mitigating performance degradation in ELC scenarios.

Table 4 also compares models enhanced with external tools. Experimental results show that traditional RAG methods offer minimal gains within context limits and only slight improvements for ELC. LongAgent improves long-text QA capability but incurs high time costs due to excessive inter-agent interactions, especially at longer contexts (e.g., ETS of 37.13 at 256K length). In contrast, CSTree-SRI outperforms these methods across all context lengths, especially beyond 256K tokens, boosting MTQA performance by 21.48%, reducing inference time by 41.11% (ETScore), and increasing accuracy by 44.17% (calculated based on CSTree-SRI with Llama-3.1-8B-Instruct as AE). We attribute this improvement to the dynamic construction of CSTree, which preserves key information in multi-turn QA, and its logic-driven hierarchical traversal strategy, effectively reducing retrieval time in extra-long context scenarios and leading to superior overall performance.

Model	32k		64k		128k		256k		Overall
	ACC(%)	ETS	ACC(%)	ETS	ACC(%)	ETS	ACC(%)	ETS	
Locally Deployed Models									
Llama-3.1-8B-Instruct	56.00	51.11	60.33	48.26	48.00	43.31	54.33	48.94	157.75
GLM-4-9B-Chat	64.00	57.14	68.33	52.43	56.67	48.68	61.00	52.32	173.62
Qwen2.5-7B-Instruct	75.00	69.44	65.33	53.95	64.67	58.98	71.00	62.77	187.17
API-Based Models									
gpt-3.5-turbo	25.67	25.50	27.33	27.11	26.00	25.77	23.33	23.20	97.66
gpt-4o-mini	<u>84.33</u>	<u>83.25</u>	<u>86.33</u>	<u>84.23</u>	75.33	73.15	74.67	69.14	208.10
deepseek-chat	**93.67**	**90.17**	**87.67**	**84.56**	77.33	72.72	75.33	70.71	210.38
Models Enhanced with External Tools									
RAG	57.33	56.59	56.00	55.02	58.33	56.81	57.00	54.55	162.74
LongAgent	66.67	46.69	65.67	51.12	62.67	38.31	65.67	37.13	179.26
CSTree-SRI (The Open-Source Model Only as Answer Expert)									
Ours (Llama-3.1 as AE)	72.67	65.03	76.33	68.71	78.33	70.45	78.33	69.06	202.35
Ours (GLM-4 as AE)	81.00	74.27	79.67	72.86	<u>82.67</u>	<u>73.22</u>	<u>80.67</u>	<u>71.15</u>	<u>211.74</u>
Ours (Qwen-2.5 as AE)	83.67	76.21	80.33	72.56	**83.00**	**73.46**	**81.00**	**75.79**	**213.78**
CSTree-SRI (A Pure Open-Source Configuration)									
Ours (Purely Llama-3.1)	68.33	60.69	68.67	59.04	65.33	58.56	69.67	58.77	182.85
Ours (Purely GLM-4)	70.33	64.86	72.33	66.21	68.00	61.96	70.67	64.72	193.23
Ours (Purely Qwen-2.5)	79.33	73.36	71.00	65.82	73.33	65.96	75.67	70.79	198.16

Table 4: Results of MTQA with LLMs and External Tool-Enhanced Methods under Different Context Lengths. The best performance is shown in **bold**, while the second best performance is represented with an <u>underline</u>.

The bottom three rows in the table 4 present that any LLM in our method can be effectively replaced with purely open-source models. Furthermore, this configuration of purely open-source models is still effective for MTQA over ELC compared to the original locally deployed models. To evaluate the resource efficiency, we also analyzed the number of LLM interactions and token consumption across different configurations. The detailed results are presented in Appendix B.3.

To further validate CSTree-SRI, we analyzed its MTQA performance across task types and difficulty levels. CSTree-SRI remains robust as question difficulty increases and excels in complex reasoning and multi-paragraph retrieval, demonstrating strong logical consistency and long-range dependency capture. Detailed results are in Appendix B.2.

5.4 Ablation Study

We conducted ablation experiments to assess the contributions of the CSTree, SE, and IE modules. Experiments were performed on the MTQA-ELC dataset with 256K-length contexts. Results are presented in Table 5, using Llama-3.1-8B-Instruct as the AE, gpt-4o-mini for SE/IE, and CSTree construction enabled as the default configuration.

Model Setting	256k	
	ACC(%)	ETS
CSTree-SRI	78.33	69.06
-w SE/IE use deepseek-chat	81.33(+3.8%)	73.35(+6.2%)
-w SE/IE use gpt-4o	79.33(+1.3%)	72.46(+4.9%)
-w SE/IE use gpt-3.5-turbo	73.00(-6.8%)	63.54(-8.0%)
-w/o CSTree	73.67(-6.0%)	64.84(-6.1%)
-w/o SE	70.00(-10.6%)	64.63(-6.4%)
-w/o IE	60.00(-23.4%)	58.60(-15.2%)

Table 5: Ablation Study on MTQA-ELC (256k-length)

Impact of Different LLMs for Expert Modules. Replacing gpt-4o-mini with the more powerful gpt-4o or deepseek-chat in the SE/IE improves both ACC and ETS. Conversely, substituting these modules with the weaker gpt-3.5-turbo leads to a decline in overall performance, highlighting the importance of strong LLMs in expert modules.

Impact of CSTree-SRI Modules. The ablation experiment results in Figure 6 show that removing CSTree results in declines in both ACC and ETS, as the framework loses logical relationships from historical information, weakening its ability to process multi-turn questions. Similarly, excluding the SE module, where non-leaf nodes store concatenated child node information instead of summarized data, reduces accuracy by 10.6% due to redundancy. This redundancy overloads the IE module's retrieval process and impairs its ability to determine further retrieval needs accurately. Notably, removing the IE module leads to the most significant performance drop, with ACC decreasing by 23.4% and ETS by 15.2%, as this module is essential for guiding the reasoning process. The introspective questioning mechanism enables the LLM to process ELC, ensuring successful multi-turn QA efficiently.

Overall, these results validate the effectiveness of each CSTree-SRI module in maintaining logical consistency, reducing retrieval redundancy, and enhancing multi-turn QA performance.

6 Conclusion

In this paper, we propose CSTree-SRI, a framework to enhance LLM performance on multi-turn QA tasks over extra-long contexts. CSTree-SRI follows an introspection-driven way to construct and search on CSTree, where logical relationships and coherence within ELC are preserved, through the collaboration of the Summary, Retrieval, Introspection and Answer expert modules. We also design the MTQA-ELC benchmark and conduct comprehensive experiments. The results demonstrate the effectiveness of our proposed CSTree-SRI.

For future work, we will refine the design of each expert module and integrate mechanisms like position encoding modifications, pre-training, and fine-tuning techniques to further improve the accuracy and efficiency of relevant context retrieval.

7 Limitations

Dependency on External LLMs for Expert Modules. The CSTree-SRI framework's reliance on third-party LLMs (e.g., gpt-4, gpt-3.5) for critical modules—Summary Expert (SE), Introspection Expert (IE), and Answer Expert (AE)—introduces systemic risks in terms of operational stability and cost efficiency. Performance bottlenecks may arise from API latency, model availability fluctuations, or unexpected service interruptions. To mitigate these risks, the framework's modular design inherently supports alternative implementations, including open-source LLMs (e.g., Llama-3, Qwen) or locally deployed models. This flexibility allows users to reduce dependency on specific vendors and enhance robustness against service disruptions. However, the financial burden of deploying high-tier LLMs—whether proprietary or self-hosted—could still render the framework economically impractical for resource-constrained users or organizations, particularly in scenarios requiring frequent or large-scale ELC processing.

8 Acknowledgments

This work has been supported by the National Key R&D Program of China under grant No. 2022YFF0902500, the National Natural Science Foundation of China under grant No.62472447, the Science and Technology Innovation Program of Hunan Province under grant No.2023RC1023, and Hunan Provincial Natural Science Foundation of China under grant No.2024JK2006, the Fundamental Research Funds for the Central Universities of Central South University. This work was carried out in part using computing resources at the High Performance Computing Center of Central South University.

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.

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

Ge Bai, Jie Liu, Xingyuan Bu, Yancheng He, Jiaheng Liu, Zhanhui Zhou, Zhuoran Lin, Wenbo Su, Tiezheng Ge, Bo Zheng, and Wanli Ouyang. 2024a. MT-bench-101: A fine-grained benchmark for evaluating large language models in multi-turn dialogues. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7421–7454, Bangkok, Thailand. Association for Computational Linguistics.

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

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2022. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pages 2206–2240. PMLR.

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

Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. 2023b. Longlora: Efficient fine-tuning of long-context large language models. *arXiv preprint arXiv:2309.12307*.

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*.

Yunpeng Huang, Jingwei Xu, Junyu Lai, Zixu Jiang, Taolue Chen, Zenan Li, Yuan Yao, Xiaoxing Ma, Lijuan Yang, Hao Chen, et al. 2023. Advancing transformer architecture in long-context large language models: A comprehensive survey. *arXiv preprint arXiv:2311.12351*.

Wai-Chung Kwan, Xingshan Zeng, Yuxin Jiang, Yufei Wang, Liangyou Li, Lifeng Shang, Xin Jiang, Qun Liu, and Kam-Fai Wong. 2024. Mt-eval: A multi-turn capabilities evaluation benchmark for large language models. *arXiv preprint arXiv:2401.16745*.

Xiaonan Li, Changtai Zhu, Linyang Li, Zhangyue Yin, Tianxiang Sun, and Xipeng Qiu. 2023. Llatrieval: Llm-verified retrieval for verifiable generation. *arXiv preprint arXiv:2311.07838*.

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

Zihan Liao, Jun Wang, Hang Yu, Lingxiao Wei, Jianguo Li, and Wei Zhang. 2024. E2llm: Encoder elongated large language models for long-context understanding and reasoning. *arXiv preprint arXiv:2409.06679*.

Yi Lu, Xin Zhou, Wei He, Jun Zhao, Tao Ji, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. Longheads: Multi-head attention is secretly a long context processor. *arXiv preprint arXiv:2402.10685*.

Bowen Peng and Jeffrey Quesnelle. 2023. Ntk-aware scaled rope allows llama models to have extended (8k+) context size without any fine-tuning and minimal perplexity degradation.

Jiezhong Qiu, Hao Ma, Omer Levy, Scott Wen-tau Yih, Sinong Wang, and Jie Tang. 2019. Blockwise self-attention for long document understanding. *arXiv preprint arXiv:1911.02972*.

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

Oguzhan Topsakal and Tahir Cetin Akinci. 2023. Creating large language model applications utilizing langchain: A primer on developing llm apps fast. In *International Conference on Applied Engineering and Natural Sciences*, volume 1, pages 1050–1056.

Szymon Tworkowski, Konrad Staniszewski, Mikołaj Pacek, Yuhuai Wu, Henryk Michalewski, and Piotr Miłoś. 2024. Focused transformer: Contrastive training for context scaling. *Advances in Neural Information Processing Systems*, 36.

Xingyao Wang, Zihan Wang, Jiateng Liu, Yangyi Chen, Lifan Yuan, Hao Peng, and Heng Ji. 2023. Mint: Evaluating llms in multi-turn interaction with tools and language feedback. *arXiv preprint arXiv:2309.10691*.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Zhuosheng Zhang, Jiangtong Li, Peng Fei Zhu, Zhao Hai, and Gongshen Liu. 2018a. Modeling multi-turn conversation with deep utterance aggregation. *ArXiv*, abs/1806.09102.

Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018b. Modeling multi-turn conversation with deep utterance aggregation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3740–3752, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jun Zhao, Can Zu, Hao Xu, Yi Lu, Wei He, Yiwen Ding, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. Longagent: Scaling language models to 128k context through multi-agent collaboration. *arXiv preprint arXiv:2402.11550*.

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.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zheng Liu, Zhicheng Dou, and Ji-Rong Wen. 2023. Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107*.

A Data Details

A.1 Data Collection

We collected a large number of English reading comprehension passages from publicly available datasets of domestic and international large-scale exams. For each question, we also obtained additional data, such as the accuracy rate of test-takers. The annotators are three undergraduate students in computer science who are familiar with reading comprehension tasks and exam question types. The annotation process involved three independent annotators labeling questions based on the original exam materials. Conflicts in labeling were resolved through discussions with two senior researchers. All exam passages and questions are publicly available on official educational websites, and the annotation work was conducted by our research team to ensure alignment with task requirements. Detailed information about the raw dataset is provided in Table 6. The abbreviations in Table 6 are defined as follows: NMET refers to the National Matriculation Entrance Test, CET denotes the College English Test, PGEE stands for the Post-graduate Entrance Examination, and TPO represents the TOEFL Practice Online.

Category	#Passages	#Words	#Questions
NMET	118	50k	446
CET	150	92k	750
PGEE	97	57k	478
TPO	207	192k	2197
Total	572	391k	3871

Table 6: Raw data statistics of MTQA-ELC

A.2 Construction Methodology

Preventing Data Leakage. To prevent "data leakage," where test data may overlap with training data, we randomized the paragraph order and shuffled multiple-choice question options. This minimizes the likelihood of LLMs generating answers based on prior exposure, ensuring a more accurate assessment of their understanding and reasoning capabilities in novel contexts.

Ensuring Fair Evaluation Across Different Lengths. To fairly evaluate model performance across varying text lengths without being influenced by data quality, we constructed three distinct "test papers" for each length. Each length's final score is the average accuracy rates and reasoning times across the three test sets.

Assessing the Gap Between LLMs and Human Performance. To evaluate the performance gap between LLMs and humans, we used the accuracy rates of test-takers for each question as the "human performance score," reflecting the real-world difficulty of the questions. The performance gap was then calculated using a series of formulas, detailed in Section 4.3.

B Additional Experiments

B.1 Single-Turn QA Evaluation

L-Eval Evaluation Setting For closed-ended tasks, we selected four datasets: Coursera, QuALITY, TOEFL, and SFcition. CSTree-SRI employed gpt-3.5-turbo for SE/IE modules while testing three AE configurations: Llama-2-7B-chat, Llama-2-13B-chat and Chatglm2-6b-8k. Baseline models used these vanilla models.

Table 7 demonstrates that our method achieves an average improvement of 13.6% when applied to different base models on the L-Eval benchmark. This indicates that our method is broadly applicable

Model	Crsr.	QuA.	TOEFL	SF	Avg.
Llama2-7B-chat	29.21	37.62	51.67	60.15	44.66
-CSTree-SRI	35.47↑	43.07↑	61.34↑	67.19↑	51.77↑
Llama2-13B-chat	35.75	42.57	60.96	54.68	48.49
-CSTree-SRI	40.26↑	46.53↑	67.82↑	64.06↑	54.67↑
Chatglm2-6b-8k	43.75	40.59	53.90	54.68	48.23
-CSTree-SRI	48.21↑	45.84↑	63.19↑	59.34↑	54.15↑

Table 7: The results of CSTree-SRI based on different model on L-Eval. The experimental data for the original models are sourced from the results reported in the L-Eval paper.

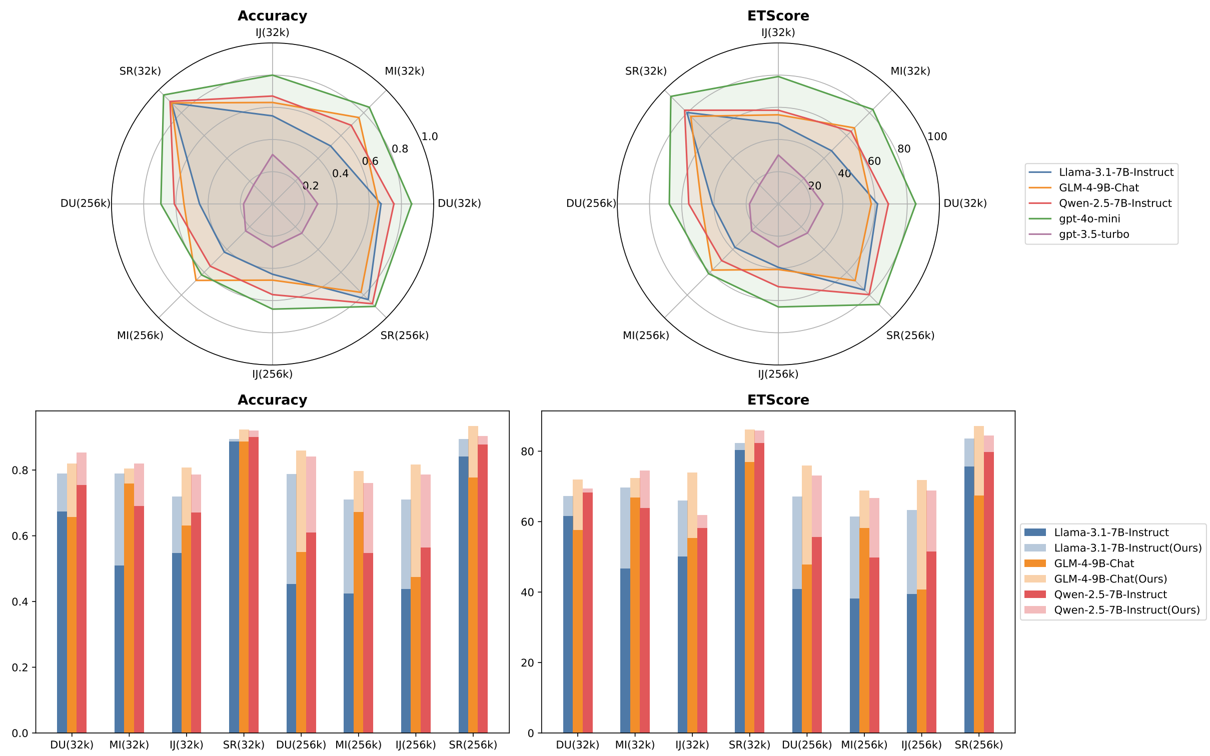

Figure 3: The radar chart represents the performance differences between models across task types. The bar chart represents the performance improvements of CSTree-SRI across task types.

Model	DU		MI		IJ		SR		Overall
	ACC(%)	ETS	ACC(%)	ETS	ACC(%)	ETS	ACC(%)	ETS	
32k Length									
Llama-3.1-7B-Instruct	67.33	61.53	50.94	46.68	54.67	50.00	88.67	80.31	180.86
-CSTree-SRI	79.00	67.33	78.87	69.67	72.00	65.97	89.33	82.26	210.21
GLM-4-9B-Chat	65.67	57.62	75.85	66.76	63.00	55.34	88.67	76.92	196.83
-CSTree-SRI	82.00	71.90	80.38	72.40	**80.67**	73.94	92.33	86.13	218.42
Qwen-2.5-7B-Instruct	75.33	68.28	69.06	63.81	67.00	58.19	90.00	82.30	201.08
-CSTree-SRI	85.33	69.41	81.89	74.50	78.67	61.77	92.00	85.86	219.71
gpt-4o-mini	**86.33**	**85.29**	**84.91**	**82.95**	80.00	**79.12**	**95.67**	**94.42**	**224.40**
gpt-3.5-turbo-16k	28.00	27.80	22.64	22.49	30.67	30.39	16.67	16.62	93.74
256k Length									
Llama-3.1-7B-Instruct	45.33	40.87	42.33	38.17	43.67	39.40	84.00	75.58	156.35
-CSTree-SRI	78.67	67.05	71.00	61.44	71.00	63.29	89.33	83.65	204.8
GLM-4-9B-Chat	55.00	47.76	67.17	58.10	47.33	40.72	77.67	67.37	199.14
-CSTree-SRI	**86.00**	**75.98**	**79.67**	**68.79**	**81.67**	**71.73**	**93.33**	87.09	**220.46**
Qwen-2.5-7B-Instruct	61.00	55.63	54.67	49.80	56.33	51.41	87.67	79.67	179.05
-CSTree-SRI	84.00	72.99	76.00	66.73	78.67	68.86	90.33	84.43	214.56
gpt-4o-mini	69.33	67.71	62.33	61.26	65.33	63.98	90.00	**88.33**	192.96
gpt-3.5-turbo-16k	18.00	17.93	23.67	23.47	27.00	26.77	25.67	25.55	96.46

Table 8: Experimental Results of MTQA for Different Types of Tasks. The best performance is shown in **bold**, while the second best performance is represented with an underline.

Model	NMET			CET			PGEE			TPO			avg.
	ACC(%)	ETS	Overall	ACC(%)	ETS	Overall	ACC(%)	ETS	Overall	ACC(%)	ETS	Overall	Overall
32k Length													
Llama-3.1-7B-Instruct	66.33	61.66	182.31	58.67	54.61	170.38	69.00	64.17	197.41	47.33	42.38	139.59	172.42
-CSTree-SRI	74.33	67.43	198.72	77.00	68.19	207.54	75.67	68.93	210.91	82.00	75.43	211.30	207.12
GLM-4-9B-Chat	64.33	58.59	178.27	61.67	55.39	176.50	55.67	50.75	170.27	73.33	62.29	193.23	179.57
-CSTree-SRI	83.33	76.24	217.01	79.67	73.14	212.89	82.00	73.92	223.76	89.67	83.09	**227.02**	220.17
Qwen-2.5-7B-Instruct	75.67	70.69	201.42	80.67	74.83	215.17	77.67	72.87	214.93	77.67	70.37	202.29	208.45
-CSTree-SRI	79.67	73.18	209.60	81.00	72.31	215.70	81.33	71.92	222.33	86.33	77.08	220.18	216.95
gpt-4o-mini	**86.67**	**85.60**	**223.75**	**83.00**	**81.82**	**219.83**	**82.67**	**79.84**	**225.09**	**89.67**	**88.47**	226.98	**223.91**
gpt-3.5-turbo-16k	37.00	36.66	122.43	31.33	31.02	114.82	26.67	26.40	111.52	12.33	12.31	67.36	104.03
256k Length													
Llama-3.1-7B-Instruct	57.33	51.64	165.01	42.00	37.91	135.65	45.00	40.59	151.67	51.67	46.51	146.49	149.71
-CSTree-SRI	68.67	60.27	188.13	72.33	62.65	197.43	67.33	58.88	196.88	88.00	79.05	221.31	200.94
GLM-4-9B-Chat	49.33	43.93	151.13	45.67	40.44	148.81	43.67	39.26	159.18	59.67	50.17	152.53	152.91
-CSTree-SRI	**79.00**	68.20	**208.97**	77.67	66.06	208.37	**78.00**	**70.00**	**218.43**	86.33	78.19	217.81	**213.40**
Qwen-2.5-7B-Instruct	70.33	63.95	191.36	56.00	51.00	164.13	58.67	53.45	179.38	74.67	67.87	193.92	182.20
-CSTree-SRI	76.67	**68.94**	204.34	**78.67**	**68.40**	**210.30**	71.33	64.81	205.05	**88.33**	79.82	**221.94**	210.41
gpt-4o-mini	69.00	67.48	188.65	65.67	64.55	183.80	62.33	60.95	186.67	81.67	**80.15**	208.43	191.89
gpt-3.5-turbo-16k	22.33	22.18	93.68	28.33	28.00	107.86	30.33	30.09	121.96	31.67	31.49	104.89	107.10

Table 9: Experimental Results of MTQA for Tasks of Different Difficulty Levels. The best performance is shown in **bold**, while the second best performance is represented with an underline.

Model	Avg. token comsumption #Input	Avg. token comsumption #Output	Avg. #Interaction Count
CSTree-SRI (A Pure Open-Source Configuration)			
Ours (Purely Llama-3.1)	23570.11	1883.58	37.71
Ours (Purely GLM-4)	22442.74	949.00	18.98
Ours (Purely Qwen-2.5)	18717.58	777.20	15.95
CSTree-SRI (The Open-Source Model Only as Answer Expert)			
Ours (Llama-3.1 as AE)	14922.58	1211.91	18.14
Ours (GLM-4 as AE)	11437.21	796.38	13.77
Ours (Qwen-2.5 as AE)	**9938.34**	**471.49**	**11.81**

Table 10: Resource Consumption Study on MTQA-ELC (256k-length). The best performance is shown in **bold**.

across various models while significantly enhancing their QA capabilities.

B.2 Multi-Turn QA Evaluation

Due to DeepSeek's widespread recognition, access to its API has become challenging, and therefore, related models were not evaluated in the experiments presented in the appendix. In future work, we plan to supplement the evaluation of its models.

Fig. 3 visualizes the experimental results of different models across various task types in multi-turn question answering, while Table 8 provides the detailed experimental data for this study. Most LLMs achieve higher accuracy and ETS scores on DU and SR tasks, indicating their inherent strength in single-paragraph retrieval. However, performance degrades significantly on MI and IJ tasks, revealing limitations in multi-paragraph retrieval and cross-context reasoning. The CSTree-SRI framework mitigates these weaknesses, demonstrating substantial improvements across all task types—particularly for MI (31.30%↑) and IJ (39.77%↑) on ETScore.

Table 9 evaluates MTQA performance on tasks of different difficulty levels. On the 32k-length task, gpt-4o-mini still achieves the highest performance; however, LLMs augmented with our CSTree-SRI demonstrate competitive results across all metrics, narrowing the gap with gpt-4o-mini. Notably, on the ELC task (256k-length), our framework outperforms gpt-4o-mini in tasks of all difficulty levels.

B.3 Resource Consumption Study

Table 10 presents the method cost analysis experiment conducted on 256k-length extra-long contexts from MTQA-ELC. Below, we explain the meaning of each statistical metric:

- **Avg. #Input token consumption:** Represents the average number of tokens input into all LLMs (SE/IE/AE) per round in the MTQA process.
- **Avg. #Output token consumption:** Represents the average number of tokens output by all LLMs (SE/IE/AE) per round in the MTQA process.
- **Avg. #Interaction Count:** Represents the average number of times each LLM (SE/IE/AE) is invoked per round during MTQA, indicating the interaction frequency between different modules in our framework.

We find that after applying our CSTree-SRI method, even when handling 256k-length extra-long contexts, the token consumption remains exceptionally low—input token consumption is under 24k, and output token consumption is under 2k. Moreover, when using Qwen-2.5-7B-Instruct as AE and GPT-4o-mini as SE/IE, the token consumption is even lower—input token consumption drops below 10k, and output token consumption falls below 0.5k, while the average interaction count among modules is only 11.81.

In table 10, we did not include the token consumption data for the Locally Deployed Models, because these models directly feed the entire 256k-length text into the LLM for every query. As a result, the average token consumption per round in multi-turn question answering is guaranteed to exceed 256k tokens, which further highlights the significant efficiency advantage of our method in terms of token consumption.

C Prompts

C.1 The Prompts for Introspective Question

C.1.1 Relative

Instruction: Please evaluate the relevance of the provided evidence to the question from the following aspects.

1. If the evidence relate to the same article as the question, respond with [Relevant]

2. If the evidence relate to the same topic, or theme as the question, respond with [Relevant]

3. If the evidence provide background knowledge or context that may help in understanding the question or related concepts, respond with [Relevant]

4.If the evidence include information could offer relevant context or serve as a contrast that helps clarify the question, respond with [Relevant]

Please judge whether the evidence is relevant to the question in order according to my standards. If it meets the standards, please return directly to the [Relevant]. Otherwise, respond with [Irrelevant].

I will provide you with multiple pieces of evidence and a question. Please indicate whether each piece of evidence is relevant to the question, separated by an @ sign. The output example is [Relevant] @ [Irrelevant] @ [Irrelevant]

Instruction: *{instruction}{question}*
Evidence: *{retrieval_content}*
Judgment:

Figure 4: The Prompt for <Relative> Introspection

C.1.2 ExtraRetr

Instruction: Based on the multiple retrieval text I found regarding this question, do you think I should continue searching for more text?

If you believe the existing text is insufficient to answer the question, please respond with [Yes] otherwise respond with [No].

Retrieval Text: *{retrieval_content}*
Question: *{question}*
Judgment:

Figure 5: The Prompt for <ExtraRetr> Introspection

C.1.3 NodeRetr

Instruction: You are an intelligent information retrieval assistant. You will be provided an instruction and a summary of an article. Your task is to determine whether it is necessary to retrieve the full content of the article based on the provided summary. There are three cases:

- If the summary relate to the same article as the question, respond with [Yes].
- If the summary suggests some similarity to the question or indicates that the article may potentially answer the question, respond with [Yes].
- If the summary already sufficiently answers the question, respond with [Yes].

If the information in the [Summary] is likely to be useful for any of these cases, please respond with [Yes]. Otherwise, respond with [No].

Summary: *{retrieval_summary}*
Instruction: *{instruction}{question}*
Judgment:

Figure 6: The Prompt for <NodeRetr> Introspection

C.1.4 Support

Instruction: You will receive an instruction, evidence, and output. Your task is to evaluate whether the output is supported by the information provided in the evidence. There are three cases:

[3-Fully supported] - All information in output is supported by the evidence, or extractions from the evidence. This is only applicable when the output and part of the evidence are almost identical.

[2-Partially supported] - The output is supported by the evidence to some extent, but there is some information in the output that is not discussed in the evidence. For instance, if the output covers multiple concepts and the evidence only discusses some of them, it should be considered a [Partially supported].

[1-No support] - The output completely ignores evidence, is unrelated to the evidence, or contradicts the evidence.

Please select from the following three options [3], [2], [1].

Instruction: *{instruction}{question}*
Evidence: *{retrieval_content}*
Output: *{answers}*
Judgment:

Figure 7: The Prompt for <Support> Introspection

C.1.5 Useful

Instruction: You are a teacher. You will receive an instruction and an output. Your task is to evaluate the student's output based on the provided instruction. You should score it according to the criteria outlined below.

Scoring Criteria:

[1-Unrelated answer]: Serious errors, confusing, Unclear and worthless.

[2-Partially related]: weak response, Multiple inaccuracies, misleading. Confusing, lacks logic.

[3-Somewhat related]: partial answer, some mistakes, Moderate clarity, includes vague parts.

[4-Relevant and mostly complete]: Generally accurate, no major errors, Clear and logical, easy to understand.

[5-Fully relevant and comprehensive answer]: Highly accurate, rich information, Very clear, logical, and valuable.

Additional Suggestions:

For higher scores, it is best to include examples and explanations that help illustrate key points. Meanwhile, Encourage thoroughness and critical thinking in responses. Please select from the following five options [5], [4], [3], [2], [1].

Instruction: *{instruction}{question}*
Output: *{answers}*
Score:

Figure 8: The Prompt for <Useful> Introspection

C.2 The Prompts for Expert Modules

C.2.1 AE Module

Instruction: *{content}{query}*. Provide the answers directly, without any introductory phrases or explanations.
Your Answer:

Figure 9: The Prompt for AE Module

Instruction: *{content}{query}*. This answer is wrong [*{preanswer}*]. Don't apologize, only provide the answers directly, without any introductory phrases or explanations.

Figure 10: The Prompt for AE Module to Regenerate Response

C.2.2 SE Module

Instruction: Please summarize the content in concise sentences, while retaining logical locators (such as unique IDs that represent paragraphs) and key information.

Content: *{}*

Figure 11: The Prompt for SE Module

C.3 The Prompts for Dataset Evaluation

C.3.1 Mixed Tasks

Instruction: Please answer the following questions based on the following information.

The content within the angle brackets <> represents paragraph IDs from various articles. These IDs are used to identify specific sections of text within different articles.

Information: *{ELC}{queries}*

Provide the answers directly, without any introductory phrases or explanations.

Your Answer:

Figure 12: The Prompt for Mixed Tasks

C.3.2 DU Task

Instruction: Please answer the following questions based on the following information.

The content within the angle brackets <> represents paragraph IDs from various articles. These IDs are used to identify specific sections of text within different articles.

The following questions are about understanding the details of paragraphs.

Information: *{ELC}{queries}*

Provide the answers directly, without any introductory phrases or explanations.

Your Answer:

Figure 13: The Prompt for DU Task

C.3.3 MI Task

Instruction: Please answer the following questions based on the following information.

The content within the angle brackets <> represents paragraph IDs from various articles. These IDs are used to identify specific sections of text within different articles.

The following questions require you to grasp the main idea of the entire article.

Information: *{ELC}{queries}*

Provide the answers directly, without any introductory phrases or explanations.

Your Answer:

Figure 14: The Prompt for MI Task

C.3.4 IJ Task

Instruction: Please answer the following questions based on the following information.

The content within the angle brackets <> represents paragraph IDs from various articles. These IDs are used to identify specific sections of text within different articles.

The following questions require you to pay attention to the logical relationship of the information in the paragraph, testing your reasoning ability.

Information: *{ELC}{queries}*

Provide the answers directly, without any introductory phrases or explanations.

Your Answer:

Figure 15: The Prompt for IJ Task

C.3.5 SR Task

Instruction: Please answer the following questions based on the following information.

The content within the angle brackets <> represents paragraph IDs from various articles. These IDs are used to identify specific sections of text within different articles.

The following questions require you to understand the meaning of phrases, sentences, or demonstrative pronouns, testing your comprehension of the entire article.

Information: *{ELC}{queries}*

Provide the answers directly, without any introductory phrases or explanations.

Your Answer:

Figure 16: The Prompt for SR Task

D Examples For Multi-turn QA over Extra-Long Context

D.1 Example For DU Task

Example of DU Task

Extra-Long Context: <article NMET 66 paragraph 1> In 1916, two girls of wealthy families, best friends from ...(84 words)... Dorothy Woodruff's granddaughter. </article NMET 66 paragraph 1>
...(256k words)...
<article TOEFL TPO 6 paragraph 5> In one example of organizing the allocation ...(117 words)... will receive insufficient moisture. </article TOEFL TPO 6 paragraph 5>

Question1: This is a question about article TOEFL TPO 120. Please choose the correct answer from options A, B, C, and D below to answer the question. According to paragraph 5, Hubbell and Johnson determined:
A. the level of aggressiveness of each of the nine species
B. the number of colonies of each of the nine species
C. the order in which the colonies in the study area had been established
D. the distribution pattern of the nests of five of the nine species
Ground Truth: D ...

Question2: This is a question about article TOEFL TPO 120. Please choose the correct answer from options A, B, C, and D below to answer the question. According to paragraph 2, some species of stingless bees are aggressive mainly toward
A. Bees from their own colony
B. Bees of their own species from different colonies
C. Nonaggressive bees that forage on the same flowers
D. Aggressive bees of other species
Ground Truth: B ...

Question3: This is a question about article CET 119. Please choose the correct answer from options A, B, C, and D below to answer the question. What makes Chris Cocalis believe there is a greater opportunity for ebike sales?
A. The younger generation's pursuit of comfortable riding.
B. The increasing interest in mountain climbing.
C. The public's concern for their health.
D. The further lowering of ebike prices.
Ground Truth: A ...

Question4: This is a question about article CET 119. Please choose the correct answer from options A, B, C, and D below to answer the question. What is the prospect of the bike industry according to Ryan Rzepecki ?
A. It will profit from ebike sharing.
B. More will be invested in bike battery research.
C. The sales of ebikes will increase.
D. It will make a difference in people's daily lives.
Ground Truth: D ...

……

Figure 17: The Example for DU Task

D.2 Example For MI Task

Example of MI Task

Extra-Long Context: <article NMET 67 paragraph 1> Can a small group of ...(54 words)... on a 24/7 basis. </article NMET 67 paragraph 1>

...(256k words)

<article PGEE 68 paragraph 7> The sharp hit to growth predicted around the ...(44 words)... may even see progress. </article PGEE 68 paragraph 7>

Question1: This is a question about article PGEE 82. Please choose the correct answer from options A, B, C, and D below to answer the question. Van Oosten believes that certain plastic objects are

A. complex in structure.
B. immune to decay.
C. inherently flawed.
D. improperly shaped.
Ground Truth: C ...

Question2: This is a question about article PGEE 82. Please choose the correct answer from options A, B, C, and D below to answer the question. The author thinks that preservation of plastics is

A. unpopular.
B. challenging.
C. costly.
D. unworthy.
Ground Truth: B ...

Question3: This is a question about article CET 113. Please choose the correct answer from options A, B, C, and D below to answer the question. What does Maryanne Taylor think of self-imposed sleeplessness ?

A. It may symbolise one's importance and success.
B. It may be practiced only by certain tech heads.
C. It may well serve as a measure of self-discipline.
D. It may turn out to be key to a successful career.
Ground Truth: A ...

Question4: This is a question about article CET 113. Please choose the correct answer from options A, B, C, and D below to answer the question. How does Dr. Sophie Bostock look at the 20-hour daily work schedule?

A. One should not adopt it without consulting a sleep expert.
B. One must be duly self-disciplined to adhere to it.
C. The general public should not be encouraged to follow it.
D. The majority must adjust their body clock for it.
Ground Truth: C ...

……

Figure 18: The Example for MI Task

D.3 Example For IJ Task

Example of IJ Task

Extra-Long Context: <article CET 143 paragraph 1> Have you ever wondered ...(35 words)... in interpersonal relationships. </article CET 143 paragraph 1>

...(256k words)

<article CET 77 paragraph 11> "We're learning that student success requires ...(43 words)... feedback loops." </article CET 77 paragraph 11>

Question1: This is a question about article TOEFL TPO 193. Please choose the correct answer from options A, B, C, and D below to answer the question. Why does the author mention “Indian mustard”?

A. To warn about possible risks involved in phytoremediation
B. To explain how zinc contamination can be reduced
C. To show that hyperaccumulating plants grow in many regions of the world
D. To help illustrate the potential of phytoremediation
Ground Truth: D ...

Question2: This is a question about article TOEFL TPO 193. Please choose the correct answer from options A, B, C, and D below to answer the question. It can be inferred from paragraph 6 that compared with standard practices for remediation of contaminated soils, phytoremediation

A. is less suitable for soils that need to be used within a short period of time
B. does not allow for the use of the removed minerals for industrial purposes
C. can be faster to implement
D. is equally friendly to the environment
Ground Truth: A ...

Question3: This is a question about article PGEE 22. Please choose the correct answer from options A, B, C, and D below to answer the question. The text suggests that immigrants now in the U.s.

A. are hardly a threat to the common culture.
B. constitute the majority of the population.
C. exert a great influence on American culture.
D. are resistant to homogenization.
Ground Truth: A ...

Question4: This is a question about article PGEE 22. Please choose the correct answer from options A, B, C, and D below to answer the question. Why are Arnold Schwarzenegger and Garth Brooks mentioned in Paragraph 5?

A. To prove their popularity around the world.
B. To show the powerful influence of American culture.
C. To reveal the public's fear of immigrants.
D. To give examples of successful immigrants.
Ground Truth: B ...

……

Figure 19: The Example for IJ Task

D.4 Example For SR Task

Example of SR Task

Extra-Long Context: <article TOEFL TPO 154 paragraph 1> While some European countries ...(75 words)... to understand the sources of their success. </article TOEFL TPO 154 paragraph 1>
...(256k words)
<article TOEFL TPO 166 paragraph 11> Regarding the appearance of celebrities ...(83 words)... like the celebrity in question. </article TOEFL TPO 166 paragraph 11>

Question1: This is a question about article TOEFL TPO 111. Please choose the correct answer from options A, B, C, and D below to answer the question. The word "simultaneously" in the passage is closest in meaning to
A. merely
B. spontaneously
C. at the same time
D. without limits
Ground Truth: C ...

Question2: This is a question about article TOEFL TPO 111. Please choose the correct answer from options A, B, C, and D below to answer the question. The word "differing" in the passage is closest in meaning to
A. increasing
B. varying
C. high
D. necessary
Ground Truth: B ...

Question3: This is a question about article TOEFL TPO 67. Please choose the correct answer from options A, B, C, and D below to answer the question. The word "comprising" in the passage 4 is closest in meaning to
A. made up of
B. covering
C. taken from
D. suggesting
Ground Truth: A ...

Question4: This is a question about article TOEFL TPO 67. Please choose the correct answer from options A, B, C, and D below to answer the question. The word "crucial" in the passage is closest in meaning to
A. established
B. understood
C. important
D. interesting
Ground Truth: C ...

……

Figure 20: The Example for SR Task