Leveraging Attention to Effectively Compress Prompts for Long-Context LLMs

Yunlong Zhao[1, 2*], **Haoran Wu**[1,3*], **Bo Xu**[1, 2]

[1]The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3]Nanjing Artificial Intelligence Research of IA, Nanjing, China
{zhaoyunlong2020, wuhaoran2018, xubo}@ia.ac.cn

Abstract

Prompt compression is increasingly studied for its potential to reduce computational costs and alleviate the burden on language models when processing lengthy prompts. Prior research has assessed token retention and removal by calculating information entropy. However, prompt compression encounters two significant challenges: (1) Information entropy, while widely used, may not be the optimal compression metric; and (2) The semantic significance of tokens is context-dependent, which renders independent token retention decisions inadequate.

We posit that the solution to these challenges lies in the intrinsic mechanism of language models. Large language models (LLMs) exhibit robust contextual processing capabilities, with recent studies on their internal dynamics revealing that the attention mechanism plays a crucial role in modeling how LLMs leverage long contexts. Building on this insight, we introduce AttnComp, a novel approach that exploits the attention mechanism within language models to guide prompt compression. Our method employs causal cross-attention from the query to the context to evaluate the significance of each token, and we develop a graph-based algorithm to efficiently cluster tokens into semantic units, thus mitigating the issue of independent dependencies.

We conduct experiments on datasets for retrieval-augmented generation and multiple long tasks involving single or multi-document QA. Our proposed method, AttnComp, outperforms previous baselines and validates the contributions of our components through analytical experiments. Compared to other methods that use a causal LM for prompt compression, our approach results in shorter latency and improved performance.

Introduction

Thanks to advanced prompting techniques, large language models (LLMs) such as ChatGPT have realized notable achievements in domains like Retrieval-Augmented Generation (RAG) (Lewis et al. 2020), Agent (Park et al. 2023), and In-Context Learning (ICL) (Dong et al. 2022). Nonetheless, these advancements present challenges to the long-context capabilities of LLMs. Recently, the development of prompt compression methods has garnered considerable interest for their potential to enhance the efficiency of context management, reduce computational and economic burdens, and minimize interference in LLMs by removing superfluous content.

Some studies have attempted to compress prompts through retrieval (Xu et al. 2024) or summary generation (Xu, Shi, and Choi 2023) methods, but these approaches each have their limitations, such as coarse retrieval granularity or latency issues with generation methods. Recently, research based information entropy theory has garnered widespread attention, with representative works including the Selective-Context (Xu, Shi, and Choi 2023) and LLMLingua series (Jiang et al. 2023a). These studies utilize a small causal model to calculate the information entropy metric, specifically the PPL, to evaluate the importance of tokens within a prompt and prune it accordingly. However, these current methods face the following issues:

- Information entropy is an empirical metric that assumes redundancy in natural language, which is not always optimal.
- A fundamental challenge in prompt compression is the assumption of strong independence underlying the process. The algorithm independently evaluates each token, deciding whether to retain or remove it, without accounting for its impact on the comprehensive semantics of the remaining prompt.

Given the challenges faced by prompt compression, two important research questions urgently need to be addressed:

Q1: How can we derive a better metric to measure the importance of information in the context?

Q2: How do we address the independence assumption in prompt compression, ensuring that removing tokens from the prompt does not affect the remaining semantics? We equate this problem to determining which tokens should be collectively considered for removal.

Salvation lies within. We contend that the solutions to the outlined questions can be derived from the intrinsic attention mechanisms of LLMs. Recent research has delved into the attention mechanism of LLMs, elucidating their core operational principles and offering vital insights into their functionalities (Wu et al. 2024). Given their robust capacity to manipulate contextual information, we assert that all

*These authors contributed equally.

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

requisite attributes for effective long-text compression are innately integrated within these attention mechanisms. Consequently, we introduce **AttnComp**, a straightforward and efficacious approach for prompt compression that exploits the native attention capabilities of LLMs.

For Question 1, we propose using query-guided cross-attention as a metric for evaluating token importance. We utilize causal cross-attention from the query to the context to assess the significance of each token within it. Compared to perplexity (PPL), attention captures richer patterns, enabling more precise identification of question-related, fine-grained semantic information. Initially, we identify the retrieval heads in the small causal LLM that integrates contextual information and then apply a maximum strategy to consolidate the importance scores. To address Question 2, we redefine the problem as minimizing semantic dependency in token grouping. We hypothesize that attention values effectively capture the degree of semantic dependency between tokens. Building on the premise that attention values reflect semantic dependencies, we develop a graph-based algorithm to efficiently group tokens into semantic units. These semantic units provide a more cohesive representation of consistent meanings, enabling our compression algorithm to operate effectively at this level, thereby mitigating challenges posed by the independence assumption.

We conduct experiments on the synthetic long-context dataset and the real-world long-context datasets from LongBench. The results show that our approach surpasses previous prompt compression baselines. Furthermore, our experimental analysis validates the effectiveness of the components we proposed.

Problem Formulation

Given an LLM input with an augmented prompt $\boldsymbol{x} = (\boldsymbol{x}_{doc}, \boldsymbol{x}_{query})$, the prompt compression system aims to compress $\boldsymbol{x}_{doc}$ to reduce the prompt length while retaining key context information, ensuring it can effectively respond to $\boldsymbol{x}_{query}$.

The objective of a prompt compression system can be formulated as:

$$\min_{\widetilde{\boldsymbol{x}}} \mathrm{D}\left(LLM\left(\widetilde{\boldsymbol{y}} \mid \widetilde{\boldsymbol{x}}\right), LLM\left(\boldsymbol{y} \mid \boldsymbol{x}\right)\right), \tag{1}$$

where $\widetilde{\boldsymbol{x}}$ denotes the compressed prompt, which is a subsequence of original prompt $\boldsymbol{x}$. $\widetilde{\boldsymbol{y}}$ and $\boldsymbol{y}$ are the outputs generated by the LLM based on $\widetilde{\boldsymbol{x}}$ and $\boldsymbol{x}$, respectively.$D(;)$ is a distance measure between two distributions. In this work, we focus on compressing the document x_{doc}, which occupies the largest portion of the prompt. The compressed prompt should be concise to maximize efficiency while remaining informative and faithful to the retrieved evidence documents.

Method: AttnComp

The series of works by LLMLingua introduces a coarse-to-fine framework. The coarse-grained compression primarily retrieves the chunked context, while the fine-grained compression focuses on pruning tokens with low information content. Our method AttnComp emphasizes fine-grained compression and seamlessly integrates with coarse-grained compression. Our contributions are twofold: we propose an cross-attention-based compression metric and introduce a semantic unit identification method with self-attention. Figure 1 illustrates our framework, and Algorithm 1 details the main process of our approach.

Algorithm 1: Pseudo code of our AttnComp

Input: A small language model M, the original context $S_{context}$, the query S_{query}, the windows size w.

Optional: Perform coarse-grained compression using an external retriever r.
if length$(S_{context}) > w$ **then**
  Split S_{context} into a list of chunks $\mathbb{S}_{\text{ori}}$, where each chunk's length does not exceed w.
else
  $\mathbb{S}_{ori} = \{S_{context}\}$
end if
Calculate the filtering ratio p based on the compression constraint.
for $C \in \mathbb{S}_{ori}$ **do**
  $\mathbf{CA}, \mathbf{SA} = M(S_{query}, C)$
  Derive the importance score for each token t_i based on $\mathbf{CA}$.
  Get the maximum spanning tree $\mathcal{T} = \mathrm{FindMST}(\mathcal{G})$, where $\mathcal{G}$ is the graph with a weight matrix $\mathbf{SA}$.
  Get the semantic units $U_1, \ldots, U_k$ by applying the community detection algorithm to the subgraph $\mathcal{T}$.
  Assign an importance score to each semantic unit U_k.
  Filter out p percent of the semantic units based on their importance scores.
end for

Output: The compressed context.

Attention Extraction

In long-context scenarios, prompts typically consist of two parts: the query and the augmented documents or demonstrations, represented as $x = (x_{query}, x_{doc})$. The key information within the context is typically query-aware, and we leverage a small language model to capture this relationship effectively. Since we are using a causal language model, we append the query at the end of the document to ensure that the query has access to the document's information. We then extract attention as follows:

$$\mathbf{Attn} = \mathrm{CasualLM}\left(x_{doc}, x_{query}\right) \tag{2}$$

$$\mathbf{CA} = \mathbf{Attn}_{1:c,n} \tag{3}$$

$$\mathbf{SA} = \mathbf{Attn}_{1:c,1:c} \tag{4}$$

where c is the document length, and n is the total prompt length. $\mathbf{CA}$ (Query-guided Cross-Attention) denotes the attention from the last token in query to the tokens in documents. $\mathbf{SA}$ (Self-Attention) denotes the attention between tokens in the document. $\mathbf{CA}$ and $\mathbf{SA}$ are employed in two key components of our algorithm: deriving the compression metric and identifying semantic units, respectively.

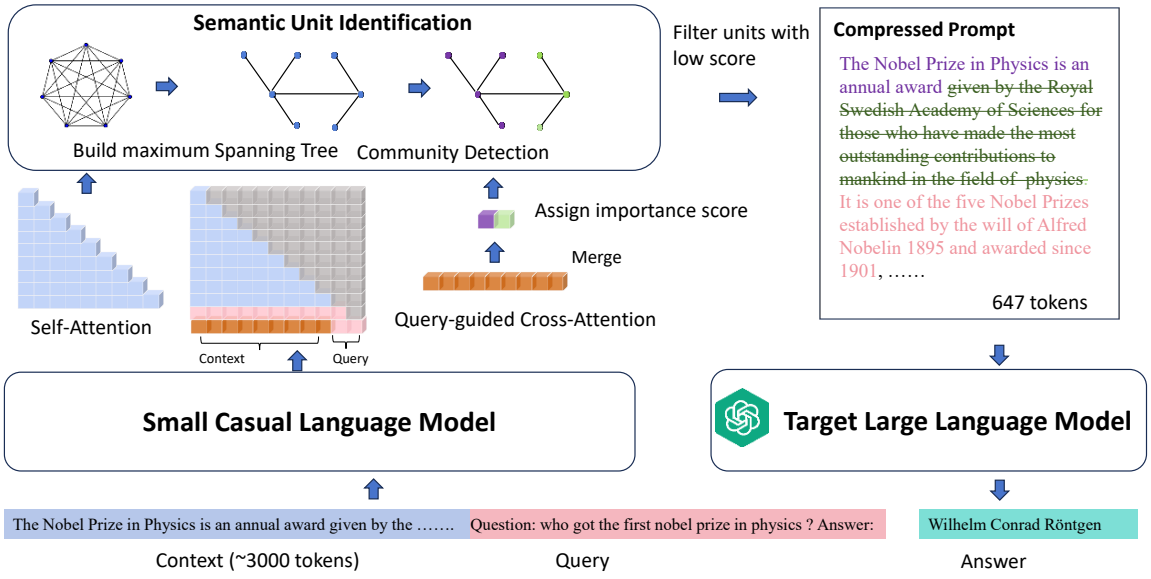

Figure 1: Framework of our proposed method.

Cross-Attention as the Compression Metric

How can we derive a more robust compression metric for evaluating token importance within context? Compared to previous metrics based on information entropy, we believe that the attention mechanism may offer a superior alternative. Recent research (Wu et al. 2024) has identified that certain attention heads, termed retrieval heads, become active in LLMs during context processing. This finding not only sheds light on the underlying mechanisms through which LLMs leverage context but also opens up a new avenue for enhancing our prompt compression method by disentangling the contextual capabilities of LLMs. As a result, we propose using the $\mathbf{CA}$ as a compression metric to decide whether to retain or discard tokens in context.

Different attention heads in LLM exhibit unique attention distributions, each reflecting distinct patterns of information utilization. To calculate the final importance score, we apply a max strategy across all attention heads. Building on insights from research on retrieval heads, we first identify the retrieval heads within the LLM that are most relevant to contextual information and select the top 20 to derive the compression metric. Formally, the important score of token at index t are defined as follows:

$$score(t) = \max_{h \in H} \left\{ \mathbf{CA}_t^h \right\} \tag{5}$$

where H represents the set of retrieval heads, $\mathbf{CA}_t^h$ represents the cross-attention scores of attention head h at index t. We adopt a max aggregation strategy for attention scores, as we observe that it more effectively preserves the information integration patterns of the retrieval head compared to averaging.

We utilize a percentile-based filtering method to adaptively select the most informative content. Specifically, we rank the importance scores and filter out the bottom $p\%$ of tokens. The value of p is determined by the target compression constraint. The remaining tokens are then combined to form the compressed prompt

Semantic Unit Identification with Self-Attention

As previously mentioned, prompt compression faces the challenge of the independence assumption. To further address this challenge, we propose a Semantic Unit Identification algorithm. This approach aims to ensure that tokens within the same unit exhibit strong semantic dependencies, while different units have weak semantic dependencies. Consequently, we can independently assess whether to retain or remove each unit.

Our approach is based on the hypothesis that the attention distribution within language models inherently encodes a mechanism for segmenting semantic units. Specifically, the self-attention between tokens captures their conditional dependencies, where tokens with strong mutual attention are semantically related and should be grouped into the same semantic unit. Conversely, tokens with weaker mutual attention exhibit lower semantic relatedness and should be assigned to different semantic units. Consequently, the task can be formally defined as finding an optimal token grouping that maximizes attention values within groups while minimizing attention values across groups.

Let S represent the set of all tokens, $S = \{t_1, t_2, \ldots, t_n\}$, where n is the number of tokens. Let $\mathbf{SA}_{pq}$ denote the causal self-attention value between token t_p and token t_q. $U_1, U_2, \ldots, U_k$ represent the k semantic units, which are groups of tokens, where each $U_i \subseteq S$, and $U_1 \cup U_2 \cup \cdots \cup$

$U_k = S$. The problem can be formulated as:

$$\max_{\{U_1,U_2,\ldots,U_k\}} \sum_{i=1}^{k} A_{\text{intra}}\ (U_i) - \lambda \sum_{1\le i<j\le k} A_{\text{inter}}\ (U_i, U_j) \tag{6}$$

where $A_{\text{intra}}(U_i) = \sum_{t_p,t_q \in U_i} \mathbf{SA}_{pq}$ represents the total attention value within group U_i, and $A_{\text{inter}}(U_i, U_j) = \sum_{t_p \in U_i, t_q \in U_j} \mathbf{SA}_{pq}$ represents the total attention value between group U_i and group U_j. λ is a weighting hyperparameter.

However, this problem presents a highly complex combinatorial optimization challenge. Given the stringent time constraints of our compression algorithm, direct computation is impractical. Therefore, we propose a heuristic method grounded in graph theory to address this problem. We first construct a maximum spanning tree to extract the core semantic structure of the prompt, then apply a community detection algorithm to partition the tokens.

We represent all tokens S as vertices V in a fully connected, undirected graph, with the $\mathbf{SA}$ matrix serving as the edge weight matrix between the vertices. We represent this graph as $\mathcal{G} = (V, w)$. Note that we apply the max strategy across all heads of $\mathbf{SA}$ to obtain scalar-weighted edges. To simplify the problem, we disregard edge direction, considering the attention from token t_i to token t_j as equivalent to the weight between them.

$$w_{ij} = w_{ji} = \mathbf{SA}_{ij} \text{ where } i > j \tag{7}$$

The maximum spanning tree (MST) is defined as the spanning tree with the greatest possible total weight compared to any other spanning tree. In a connected graph, a spanning tree is a subgraph that includes all vertices without forming any cycles. The maximum spanning tree has been employed to efficiently solve dependency parsing tasks (McDonald et al. 2005; Stanojević and Cohen 2021). Constructing the maximum spanning tree is consistent with the objective of our optimization problem.

$$\mathcal{T} = \text{FindMST}(\mathcal{G}) \tag{8}$$

Leveraging the MST, we transform the simple sequential links of tokens in the prompt into a tree-like topological structure, which highlights the semantic connections between tokens. Within this graphical structure, our goal is to identify clusters of closely linked tokens, treating them as distinct semantic units. To segment the MST, we employ the Louvain algorithm (Blondel et al. 2008), a community detection method that uncovers highly modular community structures in large networks:

$$U_1, U_2, \ldots, U_K = \text{Louvain}(\mathcal{T}) \tag{9}$$

The importance score of a semantic unit is calculated by averaging the importance scores of all tokens within it:

$$S(U_k) = \frac{1}{|U_k|} \sum_{t_i \in U_k} score(i) \tag{10}$$

We also apply a percentile-based filtering method. For the retained semantic units, the tokens are combined in their original sequence to form the final compressed prompt.

Experiments

In this section, we describe the experiments conducted to evaluate the effectiveness of our proposed approach.

Datasets and Evaluation Metric

Our experiments explore synthesized long-context scenarios within the retrieval-augmented generation setting, as well as general long-context scenarios. We use the Natural Question dataset (Kwiatkowski et al. 2019) and datasets from LongBench (Bai et al. 2023), respectively.

Natural Questions (NQ) (Kwiatkowski et al. 2019) is a classic dataset for open-domain question answering. We use a retrieval-augmented setup provided by (Liu et al. 2024). In this setup, each question is paired with 20 documents in the initial prompt, one of which contains the correct answer. The benchmark tests five positions for the ground truth document: 1st, 5th, 10th, 15th, and 20th. We evaluate using the accuracy metric as described by (Liu et al. 2024).

LongBench (Bai et al. 2023) is a benchmark designed to assess the long-context capabilities of LLMs. From this benchmark, we select datasets for single-document and multi-document question answering, including Qasper, MultiFieldQA, NarrativeQA, Musique, HotpotQA, and 2WikiMultiQA. The context lengths in these datasets range from 4k to 18k. We utilize the benchmark's provided metrics and scripts for our evaluation.

Implementation Details

In this paper, we validate our method on both open-source and commercial models. Following previous work, we use GPT-3.5-Turbo-0613 and LongChat-13B-16k on the NQ dataset, and GPT-3.5-turbo-16k and LongChat-v1.5-7B-32k on LongBench datasets. By default, we also employ the open-source model Llama-2-7B (Touvron et al. 2023) as the small LM for extracting attention, consistent with prior work. Taking into account the latency and Llama2's 4k window limitation, we set the compression algorithm's processing window to 2k for extremely long prompts. Data exceeding this window is chunked and processed over multiple passes. We implement our approach using PyTorch and Huggingface's Transformers (Wolf et al. 2019). Due to the slow execution speed in Python, we implement Prim's algorithm (Prim 1957) in C++ and access it as a dynamic library. For the Louvain algorithm, we utilize the implementation provided by the Python package *community_louvain*.

Following a similar approach to previous compression-based methods, we first apply coarse-grained compression similar to Cond.PPL, to achieve a specific compression ratio. Then, we refine the results through fine-grained compression to meet the final constraints. For the NQ dataset, we initially achieve a compression ratio of over 4x through coarse-grained compression, followed by fine-grained adjustments to satisfy the constraints. Similarly, for the LongBench dataset, consistent with previous work, we set a target

Methods	GPT3.5-Turbo						LongChat-13b						Length	
	1st	5th	10th	15th	20th	Reorder	1st	5th	10th	15th	20th	Reorder	Tokens	1/ τ
Retrieval-based Methods														
SBERT	66.9	61.1	59.0	61.2	60.3	64.4	62.6	56.6	53.9	55.0	59.1	59.1	808	3.6x
OpenAI	63.8	64.6	65.4	64.1	63.7	63.7	61.2	56.0	55.1	54.4	55.8	58.8	804	3.7x
Cond.PPL *	71.1	70.7	69.3	68.7	68.5	71.5	67.8	59.4	57.7	57.7	58.6	64.0	807	3.7x
Compression-based Methods														
Selective-Context	31.4	19.5	24.7	24.1	43.8	-	38.2	17.2	15.9	16.0	27.3	-	791	3.7x
LLMLingua	25.5	27.5	23.5	26.5	30.0	27.0	32.1	30.8	29.9	28.9	32.4	30.5	775	3.8x
LLMLingua-2	48.6	44.5	43.6	40.9	39.9	46.2	-	-	-	-	-	-	748	3.9x
LLMLingua-2†	74.0	70.4	67.0	66.9	65.3	71.9	-	-	-	-	-	-	739	3.9x
LongLLMLingua†	75.0	71.8	71.2	71.2	74.7	75.5	68.7	60.5	59.3	58.3	61.3	66.7	748	3.9x
AttnComp †	**76.6**	**73.3**	**73.1**	**72.9**	**75.0**	**76.8**	**70.1**	**62.1**	**61.6**	**61.2**	**61.3**	**68.6**	647	4.5x
Original Prompt	75.7	57.3	54.1	55.4	63.1	-	68.6	57.4	55.3	52.5	55.0	-	2,946	-
Zero-shot	56.1						35.0						15	196x

Table 1: Performance of different methods on NaturalQuestions.† indicates the method using coarse-grained compression (i.e., Cond.PPL). The baseline results are directly cited from (Jiang et al. 2023b) and (Pan et al. 2024). For the results under reorder, we apply the reordering strategy, while for the 1st to 20th positions, the documents remain in their original order.

	Acc	$1/\tau$
AttnComp	**68.6**	4.5x
- w/o Semantic Units	67.1	4.5x
- w/ Phrase	67.4	4.6x
- w/o Retrieval Heads	68.2	4.5x
- w/ PPL	61.5	4.5x
- w/ Iterative Token-level Compression	67.5	4.5x
- w/ Mistral-7B-v0.2	**68.8**	4.5x

Table 2: Ablation study.

compression limit of 2,000 tokens. We first apply coarse-grained compression to reduce the length to 4,000 tokens and then use a 50% fine-grained filtering process to meet the final compression requirement.

Baselines

Our baselines include retrieval-based methods and compression-based methods.

(i) *Retrieval-based Methods.* Retrieval-based methods use a retriever to rank documents based on their relevance to the question. They discard sentences or paragraphs with low relevance until the compression constraint is met while preserving the original document order. We select the following retrievers: SentenceBERT (Reimers and Gurevych 2020), OpenAI Embedding, and Cond.PPL (Jiang et al. 2023b) to measure the association between the query and the documents.

(i) *Compression-based Methods.* We compare our method with state-of-art methods for prompt compression.

- Selective-Context (Li et al. 2023) is the first work to discuss context compression, which prunes redundant lexical units by estimating self-information through a language model.
- LLMLingua (Jiang et al. 2023a) proposes a coarse-to-fine approach to manage compression ratio constraints. It employs iterative token-level prompt compression and utilizes perplexity as the compression metric.
- LongLLMLingua (Jiang et al. 2023b) is an improved version based on LLMLingua. It uses conditional perplexity for coarse-grained compression of the context and contrastive perplexity for fine-grained compression.
- LLMLingua-2 (Jiang et al. 2023a) defines prompt compression as a token classification task (i.e., preserve or discard). It is a BERT-based model trained on a dataset collected from GPT.

Main Results

Tables 1 and 3 compare the performance of our method in both RAG settings and general long-context scenarios. The following observations and conclusions can be drawn:

(1) On the NQ dataset under RAG settings, our method outperforms previous baselines in both performance and compression rate. This improvement is evident not only in the open-source LongChat model but also in commercial closed-source models, underscoring the effectiveness of our approach. Moreover, on datasets without reranking strategies, our method also shows improvement, partially mitigating the 'lost-in-the-middle' phenomenon. (2) On the LongBench dataset, our method significantly enhances performance in most tasks on the open-source LongChat-v1.5-7B-32k model, surpassing the baseline. For the commercial model, our method also achieves overall performance improvement. However, due to the strong contextual capabilities of GPT, there is a slight performance decline in the Qasper and NarrativeQA tasks.

Model	Methods	Qasper	MultiFieldQA	NarrativeQA	MuSiQue	HotpotQA	2WikiMultihopQA	AVG
LongChat-v1.5-7B-32k	Original Prompt	27.7	41.4	**16.9**	9.7	31.5	20.6	24.6
	LLMLingua	23.3	34.8	14.3	9.0	26.2	20.2	21.3
	LLMLingua2	26.2	32.7	8.5	9.3	24.9	26.6	21.4
	LongLLMLingua	27.0	39.9	15.5	14.7	33.2	22.4	25.5
	AttnComp	**30.4**	**43.8**	15.8	**19.8**	**40.0**	**27.6**	**29.6**
GPT3.5-Turbo	Original Prompt	**43.3**	52.3	**23.6**	26.9	51.6	37.7	39.2
	LLMLingua	25.9	35.7	12.4	12.8	38.5	35.3	26.8
	LLMLingua2	37.0	42.5	14.7	22.1	44.3	42.2	33.8
	LongLLMLingua	37.1	48.9	17.2	30.0	48.8	**48.9**	38.5
	AttnComp	38.8	**55.1**	21.5	**30.2**	**58.4**	42.7	**41.1**

Table 3: Performance of different methods on LongBench. The target compression constraint is 2000 tokens.The original prompt results are directly cited from (Bai et al. 2023).

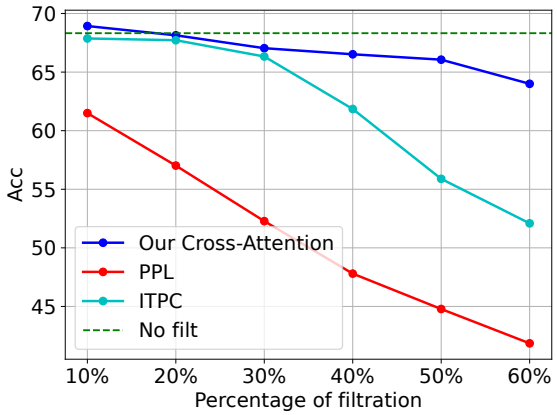

Figure 2: The effect of filtration rate across different token-level compression metrics.

Analysis

We conduct additional analysis experiments to better demonstrate the effectiveness of our method. All experiments are carried out on the NQ dataset, using LongChat-13b as the target LLM.

Ablation Study

To further validate the contributions of the various components in our method, we conduct ablation experiments on the following model variants: (1) Ours w/o Semantic Units, which compresses the prompt at the token level; (2) Ours w/ Phrase Units, which employs spaCy to merge tokens into phrase units and compresses the prompt at the phrase level; (3) Ours w/ Retrieval Heads, which uses all attention heads instead of only retrieval heads; (4) Ours w/ PPL, which uses PPL as the compression metric; (5) Ours w/ Iterative Token-level Compression, which uses iterative compression as in LongLLingua; (6) Ours w/ Mistral-7B-v0.2, which uses another recent small LM for fine-grained compression.

Table 2 presents the results of the ablation study, revealing that the removal of any individual component results in a performance decline. These findings highlight the effectiveness of our proposed approach, which involves using attention as a compression metric and introducing semantic units. The results of variant (6) further demonstrate that AttnComp proves effective when combined with different small LMs, and pairing it with more advanced models could lead to additional performance improvements.

Does Cross-Attention Serve as an Effective Compression Metric?

To further assess the effectiveness of our proposed cross-attention as a compression metric, we systematically compare the performance of various compression metrics across different filtering rates. Figure 2 shows the performance curves at these filtering rates. Our findings indicate that the cross-attention-based compression method not only enhances performance at lower filtering ratios but also significantly outperforms PPL and iterative compression methods at higher compression rates, thereby validating cross-attention as an effective compression metric. Notably, as the filtering ratio increases, our approach maintains over 90% of the original performance, even when the majority of the context is filtered out, whereas PPL and ITPC suffer from significant performance degradation. This underscores the superior efficacy of our method, particularly in fine-grained compression scenarios.

Does AttnComp Identify Good Semantic Units?

To evaluate whether our algorithm effectively identifies meaningful semantic units, we examine how well the optimization problem defined in Equation 6 is solved. Table 4 shows the objective function values, where 'Intra Attention' refers to attention scores within a semantic unit, and 'Inter Attention' denotes the scores between different semantic units. For comparison, we employ a random partitioning method that produces the same number of semantic units as our approach. Our results indicate that the intra-unit attention scores identified by our algorithm are eight times

	Intra Attention	Inter Attention
Random Units	359	11543
Semantic Units	2881	9049

Table 4: Analysis of objective function values.

The first Nobel Prize in Physics was awarded in 1[1]901 to Wilhelm Conrad Röntgen, of Germany, [2]who received 150,782 SEK, which is equal to 7,731,004 SEK in December 2007.[3]John Bardeen is the only laureate to win the prize twice—in 1956 and 1972. Maria Skłodowska[4]-Curie also won two Nobel Prizes, for physics in 1903 and chemistry in 1911.[5]William Lawrence Bragg was, until October 2014, the youngest ever Nobel laureate; he won the prize in 1915 at the age of 25……

Figure 3: A Case of semantic units. Each color represents a semantic unit.

	Raw	LongLLMLingua (4.6x)	Ours (4.5x / 9.4x)
Latency	6.31	1.72	1.59 / 1.32
Acc	55.3	67.1	68.6 / 64.3

Table 5: Efficiency analysis results.

higher than those generated by the random method, demonstrating that our approach successfully addresses the optimization problem.

Additionally, we manually examine the semantic units identified by our method. Figure 3 illustrates an example, demonstrating that the semantic units produced by our algorithm typically represent complete semantic structures, such as full phrases, clauses, or sentences.

Latency Evaluation

Table 5 shows the overall latency results (in seconds per example), including compression and response generation latencies. Unlike LongLLMLingua, our method uses a single LLM call without iterative compression and reduces compression latency. At the same performance level, our method demonstrates higher overall inference efficiency. With the higher compression rate, it retains sufficient information for comparable performance while minimizing latency. However, it is important to note that when the prompt is particularly long, the compression efficiency of our method tends to decrease, especially during the semantic unit identification stage. There is still room for further optimization in our algorithm's efficiency.

Related Work

Long Context for LLM

Long context modeling is a fundamental challenge for large language models based on Transformer. Many efforts focus on improving the model itself, such as improving the attention mechanism (Sun et al. 2023), expanding the window length of LLMs through continue pre-training (Xiong et al. 2024), or improving positional encoding (Chen et al. 2023a). Unlike these approaches, which aim to extend the window length of the LLM itself, our work takes a different direction by compressing the long context or prompt.

Prompt Compression

Prompt compression techniques had already been explored in the era of BERT-scale (Devlin 2018) language models (Goyal et al. 2020; Kim and Cho 2021; Modarressi, Mohebbi, and Pilehvar 2022). With the widespread success of large generative language models (Raffel et al. 2020; Brown et al. 2020) across various tasks (Zhao et al. 2024), prompt compression has garnered significant attention and can broadly be categorized into two main approaches: black-box compression and white-box compression. White-box compression focuses on compressing the context into summary vectors (Mu, Li, and Goodman 2024; Chevalier et al. 2023; Ge et al. 2024). However, this line of research requires the target LLM to be a model with accessible parameters and is highly task-specific. On the other hand, black-box compression (Li et al. 2023; Jiang et al. 2023a; Pan et al. 2024) typically relies on information entropy theory, using a small language model to evaluate the significance of each token within the original prompt and subsequently removing those deemed less important.

Attention in LLMs

As the core mechanism of Transformer, the attention mechanism (Vaswani et al. 2017) has been extensively studied. The prevailing view is that while the FFN layer (Dai et al. 2022) stores knowledge, attention is where the algorithm is implemented. Some work has analyzed the role of attention heads in LLMs, identifying certain retrieval heads (Wu et al. 2024) and revealing the intrinsic mechanisms by which LLMs utilize context information. Prior to LLMs, research also examined the role of attention in natural language generation (Lu et al. 2022) and phrase tagging (Gu et al. 2021). In the field of retrieval-augmented generation, many studies have built upon earlier research in ODQA that utilized attention for retrieval (Lee et al. 2022) or token elimination (Berchansky et al. 2023). Recent studies have focused on using state-of-the-art large models, such as GPT-4, to generate complete semantic units, namely propositions (Chen et al. 2023b). In contrast, our research leverages the inherent properties of attention mechanisms to achieve more efficient segmentation of these semantic units.

Conclusion

In this paper, we introduce AttnComp, a method that leverages the built-in attention mechanisms of LLMs for prompt compression. This approach employs cross-attention to derive a more effective compression metric and incorporates graph-based algorithms to identify semantic units, addressing the independence assumption inherent in prompt compression. We validated the effectiveness of this method on RAG and other common long-text tasks. Even at high filtering rates, AttnComp retains most of its performance while significantly reducing costs and inference latency. Furthermore, this method paves the way for new avenues in utilizing the contextual capabilities of LLMs, offering valuable insights for the deeper understanding and application of LLMs.

Acknowledgments

This work is supported by the National Science and Technology Major Project (2022ZD0116005) and Science and Technology Research and Development Plan of China Railway (P2023S001). The authors would like to thank the reviewers for their helpful comments and suggestions to improve the manuscript.

References

Bai, Y.; Lv, X.; Zhang, J.; Lyu, H.; Tang, J.; Huang, Z.; Du, Z.; Liu, X.; Zeng, A.; Hou, L.; et al. 2023. Longbench: A bilingual, multitask benchmark for long context understanding. *arXiv preprint arXiv:2308.14508*.

Berchansky, M.; Izsak, P.; Caciularu, A.; Dagan, I.; and Wasserblat, M. 2023. Optimizing Retrieval-augmented Reader Models via Token Elimination. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 1506–1524.

Blondel, V. D.; Guillaume, J.-L.; Lambiotte, R.; and Lefebvre, E. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10): P10008.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chen, S.; Wong, S.; Chen, L.; and Tian, Y. 2023a. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*.

Chen, T.; Wang, H.; Chen, S.; Yu, W.; Ma, K.; Zhao, X.; Yu, D.; and Zhang, H. 2023b. Dense x retrieval: What retrieval granularity should we use? *arXiv preprint arXiv:2312.06648*.

Chevalier, A.; Wettig, A.; Ajith, A.; and Chen, D. 2023. Adapting Language Models to Compress Contexts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 3829–3846.

Dai, D.; Dong, L.; Hao, Y.; Sui, Z.; Chang, B.; and Wei, F. 2022. Knowledge Neurons in Pretrained Transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 8493–8502.

Devlin, J. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dong, Q.; Li, L.; Dai, D.; Zheng, C.; Wu, Z.; Chang, B.; Sun, X.; Xu, J.; and Sui, Z. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Ge, T.; Jing, H.; Wang, L.; Wang, X.; Chen, S.-Q.; and Wei, F. 2024. In-context Autoencoder for Context Compression in a Large Language Model. In *The Twelfth International Conference on Learning Representations*.

Goyal, S.; Choudhury, A. R.; Raje, S.; Chakaravarthy, V.; Sabharwal, Y.; and Verma, A. 2020. Power-bert: Accelerating bert inference via progressive word-vector elimination. In *International Conference on Machine Learning*, 3690–3699. PMLR.

Gu, X.; Wang, Z.; Bi, Z.; Meng, Y.; Liu, L.; Han, J.; and Shang, J. 2021. Ucphrase: Unsupervised context-aware quality phrase tagging. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 478–486.

Jiang, H.; Wu, Q.; Lin, C.-Y.; Yang, Y.; and Qiu, L. 2023a. Llmlingua: Compressing prompts for accelerated inference of large language models. *arXiv preprint arXiv:2310.05736*.

Jiang, H.; Wu, Q.; Luo, X.; Li, D.; Lin, C.-Y.; Yang, Y.; and Qiu, L. 2023b. Longllmlingua: Accelerating and enhancing llms in long context scenarios via prompt compression. *arXiv preprint arXiv:2310.06839*.

Kim, G.; and Cho, K. 2021. Length-Adaptive Transformer: Train Once with Length Drop, Use Anytime with Search. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 6501–6511.

Kwiatkowski, T.; Palomaki, J.; Redfield, O.; Collins, M.; Parikh, A.; Alberti, C.; Epstein, D.; Polosukhin, I.; Devlin, J.; Lee, K.; et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7: 453–466.

Lee, H.; Kedia, A.; Lee, J.; Paranjape, A.; Manning, C. D.; and Woo, K.-G. 2022. You Only Need One Model for Open-domain Question Answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 3047–3060.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474.

Li, Y.; Dong, B.; Guerin, F.; and Lin, C. 2023. Compressing Context to Enhance Inference Efficiency of Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 6342–6353.

Liu, N. F.; Lin, K.; Hewitt, J.; Paranjape, A.; Bevilacqua, M.; Petroni, F.; and Liang, P. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12.

Lu, Y.; Zhang, J.; Zeng, J.; Wu, S.; and Zong, C. 2022. Attention analysis and calibration for transformer in natural language generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30: 1927–1938.

McDonald, R.; Pereira, F.; Ribarov, K.; and Hajic, J. 2005. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, 523–530.

Modarressi, A.; Mohebbi, H.; and Pilehvar, M. T. 2022. AdapLeR: Speeding up Inference by Adaptive Length Reduction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1–15.

Mu, J.; Li, X.; and Goodman, N. 2024. Learning to compress prompts with gist tokens. *Advances in Neural Information Processing Systems*, 36.

Pan, Z.; Wu, Q.; Jiang, H.; Xia, M.; Luo, X.; Zhang, J.; Lin, Q.; Rühle, V.; Yang, Y.; Lin, C.-Y.; et al. 2024. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint arXiv:2403.12968*.

Park, J. S.; O'Brien, J.; Cai, C. J.; Morris, M. R.; Liang, P.; and Bernstein, M. S. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, 1–22.

Prim, R. C. 1957. Shortest connection networks and some generalizations. *The Bell System Technical Journal*, 36(6): 1389–1401.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140): 1–67.

Reimers, N.; and Gurevych, I. 2020. Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4512–4525.

Stanojević, M.; and Cohen, S. B. 2021. A root of a problem: Optimizing single-root dependency parsing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 10540–10557.

Sun, Y.; Dong, L.; Huang, S.; Ma, S.; Xia, Y.; Xue, J.; Wang, J.; and Wei, F. 2023. Retentive network: A successor to transformer for large language models. *arXiv preprint arXiv:2307.08621*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Wu, W.; Wang, Y.; Xiao, G.; Peng, H.; and Fu, Y. 2024. Retrieval head mechanistically explains long-context factuality. *arXiv preprint arXiv:2404.15574*.

Xiong, W.; Liu, J.; Molybog, I.; Zhang, H.; Bhargava, P.; Hou, R.; Martin, L.; Rungta, R.; Sankararaman, K. A.; Oguz, B.; et al. 2024. Effective Long-Context Scaling of Foundation Models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 4643–4663.

Xu, F.; Shi, W.; and Choi, E. 2023. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *arXiv preprint arXiv:2310.04408*.

Xu, P.; Ping, W.; Wu, X.; McAfee, L.; Zhu, C.; Liu, Z.; Subramanian, S.; Bakhturina, E.; Shoeybi, M.; and Catanzaro, B. 2024. Retrieval meets Long Context Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Zhao, Y.; Ni, Z.; Hu, Z.; Xu, S.; and Xu, B. 2024. Bridge the Query and Document: Contrastive Learning for Generative Document Retrieval. In *2024 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.