Exploring Attention Attractors in Large Language Models

Ziheng Wang, Zihao Yue, Wenxuan Wang, Qin Jin[†]
AIM3 Lab, Renmin University of China
{zihengwang, yzihao, wangwenxuan, qjin}@ruc.edu.cn

Abstract

This paper explores *attention attractors*—tokens that draw significantly high attention—in large language models. We analyze them from three perspectives: (1) **Functionality**: We demonstrate their role in aggregating information from preceding contexts to facilitate future predictions. (2) **Distribution**: Through layer-wise and token-wise analysis, we reveal that attention attractors are widely distributed across layers but predominantly originate from low-semantic words like "_the". (3) **Mechanism**: We demonstrate the correlation between attention weights allocated to tokens with their specific activation dimension values. We hope these findings provide new insights into the attention mechanisms of large language models and inspire further exploration. Code will be released at https://github.com/luyouqi233/AttentionAttractor.

1 Introduction

Large Language Models (LLMs) have advanced rapidly, demonstrating remarkable capabilities across a wide range of tasks (OpenAI, 2024; Touvron et al., 2023; Grattafiori et al., 2024). With the attention mechanism (Vaswani et al., 2017) as a core component, these models excel at capturing long-range dependencies within a given context.

Recent research has sought to deepen our understanding of how attention operates within LLMs, particularly focusing on tokens that receive significantly high attention in models. Some studies have investigated the characteristics of these tokens, while others have explored their applications in downstream tasks, such as context pruning for long text understanding (Liu et al., 2023; Zhang et al., 2023; Cai et al., 2024; Ge et al., 2024).

In this work, we conduct a comprehensive investigation of tokens that attract extensive attention in LLMs, which we refer to as *attention attractors*. Our first objective is to examine their functionality. Existing research has proposed varying conjectures or explanations. For example, Huang et al. (2024) hypothesize that attention attractors function as "summary tokens" and excessive reliance on them can lead to hallucination. Yu et al. (2024), on the other hand, discover that tokens with weaker semantics tend to attract high attention, acting as attention sinks (Xiao et al., 2024) that offload excessive attention. Meanwhile, other studies treat these attention attractors as key tokens that preserve contextual information for model predictions, leveraging them to enhance inference efficiency in long or streaming contexts. Through our analysis, we observe that attention attractors not only receive high attention from subsequent tokens but also distribute attention to more distant tokens in preceding contexts. By inspecting the inner information flow among contexts and through these attention attractors, we confirm that these attractor tokens function as information aggregators, gathering information from preceding contexts and making it available for future predictions.

Next, we investigate how these attention attractors are distributed in the model. From a layer-wise perspective, our findings reveal significant variations in their distribution across layers, with most attractors concentrated in the initial and middle layers. Through a token-wise analysis, we identify specific words that are more likely to become attention attractors, for example, "_the", " . ", and "\n" in shallow layers and " ’ " in deeper layers. Notably, we find that words with lower semantic content are more likely to serve as attention attractors. We validate this observation through several context pruning experiments in both long context and streaming context settings, demonstrating that retaining only these "low-semantic" words in the key-value (KV) cache largely preserves model performance across most tasks.

Finally, we explore the underlying mechanism

[†] Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1148–1160

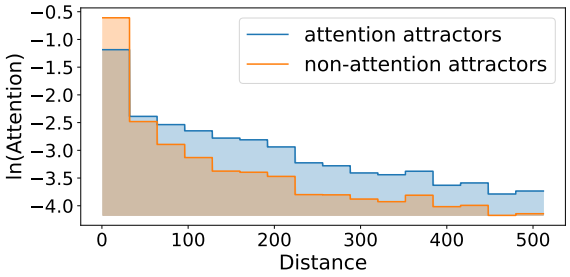


Figure 1: Distribution of attention to preceding tokens from attention attractors versus non-attention attractor tokens, based on results from layer 18 of the model.

of attention attractors from the perspective of activation value. Similar to previous works (Sun et al., 2024), we find that specific dimension activations can reflect the attention allocation for each token. We validate this finding through several long context pruning experiments, demonstrating that a retention policy based solely on single-dimension activation values in the KV cache largely preserves model performance across most tasks. This observation suggests that LLMs may pre-determine attention attractors through their hidden state representations before explicit attention computation occurs.

Collectively, our study provides an in-depth exploration of the functionality, distribution, and underlying mechanism of attention attractors, uncovering intriguing model behavior and information aggregation patterns in large language models. We hope our findings can offer valuable insights for further research and progress on understanding and development of large language models.

2 Functionality of Attention Attractors

We first investigate the role of attention attractors in large language models: why do they emerge, and what function do they serve? Previous studies have offered different interpretations. For example, Huang et al. (2024) denote these tokens as "summary tokens", while (Yu et al., 2024) attribute their prominence to over-reliance and propose reducing their attention allocation. Other works use high attention tokens as key tokens for KV cache compression (Cai et al., 2024; Zhang et al., 2023). To gain a clearer understanding, we analyze attention attractors through two perspectives: 1) attention patterns (Section 2.1), to examine how these tokens interact with context, and 2) information flow (Section 2.2), to assess their role in aggregating and transmitting information.

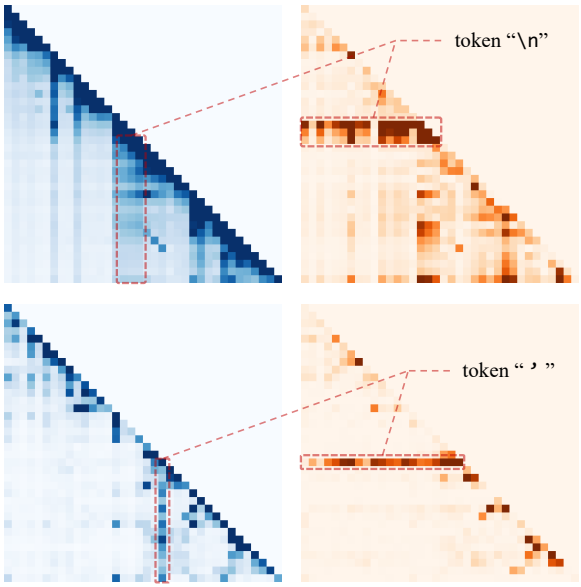


Figure 2: Attention (blue) and saliency (orange) maps from the Llama2-7b model. The top row displays results from a shallow layer (layer 1), while the bottom row shows results from a deeper layer (layer 29).

2.1 Attention Analysis

Setup. We analyze the attention patterns of attractor tokens using the Llama2-7b model (Touvron et al., 2023). The model is provided with text from the Quality Test dataset (Pang et al., 2022), using 232 long texts that can fulfill the context window of 4,096 of the model. Texts longer than this are truncated. A token is identified as an attention attractor if it receives at least 0.03% of the total attention from subsequent tokens across all cross-attention weights.

Results. We observe that attention attractors not only receive more attention from subsequent tokens but also attend to a wider range of preceding contexts compared to other tokens. Fig. 1 presents the attention distribution of attractors versus non-attractors. As illustrated, attention to preceding tokens generally declines with distance for all tokens, but attention attractors exhibit a more even distribution, allocating greater attention to distant tokens rather than just adjacent ones. This distinct behavior suggests that attention attractors aggregate information from the broader context and provide this condensed information for future use. This aligns with the "anchor token" hypothesis proposed by Wang et al. (2023).

2.2 Information Flow Analysis

To further examine the role of attention attractors, we analyze information flow within language models using the saliency score (Simonyan et al., 2014). The saliency score of a token quantifies its influence on model prediction and is commonly used to represent the information flow between tokens (Wang et al., 2023; Yue et al., 2024). In the context of language models, the saliency score between two tokens is defined as:

$$I_l = \left| \mathbb{E}_h (A_{h,l} \odot \frac{\partial \mathcal{L}(x)}{\partial A_{h,l}}) \right|, \quad (1)$$

where $A_{h,l}$ represents the attention value of the h-th attention head in the l-th layer, $\odot$ denotes element-wise product, and the saliency score is averaged across all heads. $I_l(i,j)$ measures the significance of information flow from the j-th token to the i-th token with respect to the loss function $\mathcal{L}(x)$. We employ the same experimental setup as in Section 2.1 and compute saliency scores for every token pair in the model's context window.

Results. We first present qualitative results from our saliency analysis. As depicted in Fig. 2, tokens that draw significant attention from subsequent contexts also tend to receive substantial information from preceding contexts. This observation reinforces our hypothesis that attention attractors serve as information aggregators.

For further validation, we employ quantitative metrics akin to those used by Wang et al. (2023) to quantify the information flow within the model. We explore the information flow from (1) the context to attention attractors and (2) attention attractors to target (prediction) positions. For a set of attention attractors $A = \{a_i\}_{i=1}^{K}$ and other tokens $W = \{w_i\}_{i=1}^{N}$ from the context, we compute the average significance of information flow from context tokens to attention attractors as follows:

$$S_{wa} = \frac{\sum_{(i,j)\in C_{wa}} I_l(i,j)}{|C_{wa}|}, \quad (2)$$
$$C_{wa} = \{(a_k, j) : k \in [1, K], j < a_k\},$$

where C_{wa} represents the set of valid token pairs where j (a preceding token) contributes information to a_k (an attractor). We also compute the average significance of information flow between any other tokens pair:

$$S_{ww} = \frac{\sum_{(i,j)\in C_{ww}} I_l(i,j)}{|C_{ww}|}, \quad (3)$$
$$C_{ww} = \{(i,j) : j < i\} - C_{wa}.$$

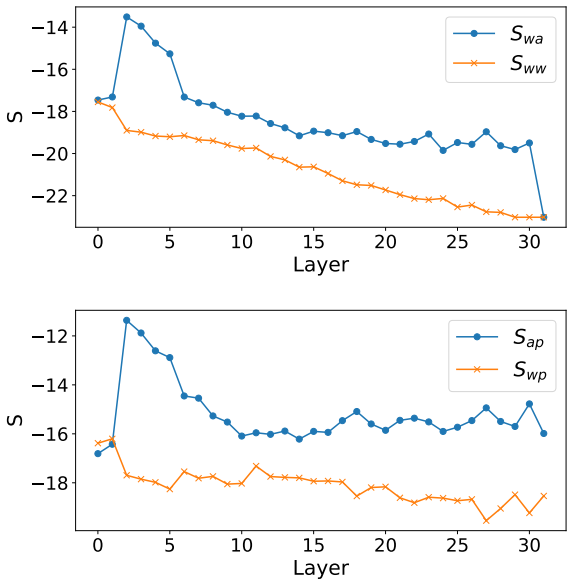


Figure 3: Layer-wise average significance of information flow between attention attractors (a), prediction positions (p), and other tokens (w) in Llama2-7b. Results on additional models are provided in Appendix A.1.

Similarly, we compute the average significance of information flow from attention attractors or other context tokens to target prediction tokens ($P = \{p_i\}_{i=1}^{M}$) as well. We backward language modeling loss of the last 128 tokens for each sequence to obtain saliency scores.

As demonstrated in Fig. 3 (top), the information flow from contexts to attractors is significantly more pronounced than between other tokens, suggesting that attention attractors are effective at gathering information from preceding contexts. Additionally, Fig. 3 (bottom) shows that the information flow from attractors to prediction positions is also more substantial than that from other tokens. This indicates that attention attractors contribute more crucial information to final predictions. These findings underscore a distinct information forwarding path in large language models: Context Tokens → Attention Attractors → Prediction Tokens.

Takeaways: Attention attractors play the role as essential information hubs: collecting information from the broader context, and transmitting the condensed contextual information to target positions for model prediction.

3 Distribution of Attention Attractors

Building on our confirmation of the information aggregation functionality of attention attractors, in this section, we investigate how these attention at-

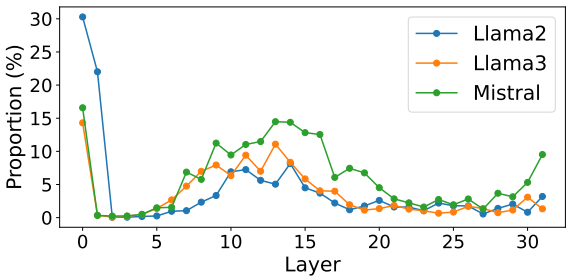
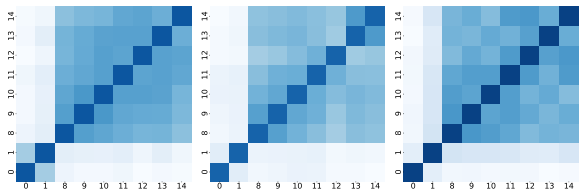


Figure 4: Proportion of attention attractors among all tokens in each layer of the model.



(a) Llama2 (b) Llama3 (c) Mistral

Figure 5: Similarity of attention attractor token between each pair of layers. We show only the layers with a significant number of attention attractors, as indicated in Fig. 4.

tractors are distributed in the model to facilitate the delivery of contextual information. We analyze the distribution of attention attractors from both layer-wise and token-wise perspectives.

3.1 Layer-wise Analysis

We first investigate how attention attractors are distributed across different layers. For this analysis, we employ Llama2-7b (Touvron et al., 2023), Llama3-8b (Grattafiori et al., 2024) and Mistral-7b (Jiang et al., 2023) as models for analysis, using the same dataset as in Section 2.1. As illustrated in Fig. 4, the number of attention attractors varies greatly across layers but follows similar trends in both models. The initial layers contain a large number of attention attractors, which rapidly decline as the layers deepen. Interestingly, attention attractors re-emerge in the middle layers before disappearing again toward the end.

A closer examination of the cross-layer evolution of these attention attractors reveals that the specific tokens selected as attractors also change across layers. Fig. 5 illustrates the similarity of the attractor token set between two layers. While neighboring layers exhibit relatively similar attention attractors (e.g., layer 0 and layer 1, as well as layers 8 through 14), attractors in shallow layers differ completely from those in deeper layers.

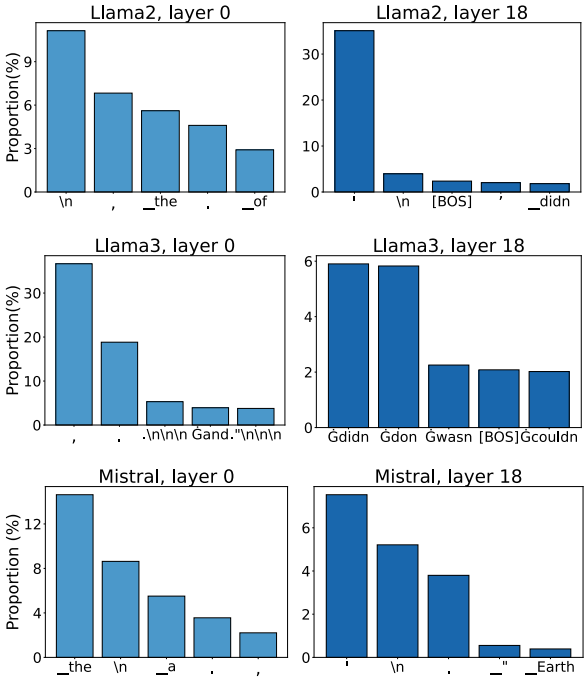


Figure 6: Proportion of the most frequent attention attractors in the shallow and middle layers of the model. Interestingly, in the middle layers of Llama2-7b and Mistral-7b, the token " ' " is the most frequent attention attractor. In contrast, Llama3-7b frequently employs "didn" and "don", which typically precede " ' ".

This suggests that the tokens responsible for aggregating information change at different stages of context encoding. Next, we investigate how attention attractors are distributed from a token-wise perspective, i.e., which tokens are more likely to become attention attractors.

3.2 Token-wise Analysis

Consistent with previous works (Yu et al., 2024; Devoto et al., 2024), we observe that attention attractors frequently correspond to semantically limited words such as "_the" and the line break token "\n", as shown in Fig. 6. In shallow layers, words that mostly frequently become attention attractors include "_the" and "\n", etc. From an information aggregator perspective, leveraging such frequent words, which are dispersed throughout documents, is well-suited for collecting local information from neighboring contexts in shallow layers. In contrast, deeper layers shift attention attractors toward less frequent words, such as punctuation marks like "'", suggesting a transition to higher-level and longer-range information aggregation. We also illustrate the variation of attention attractors across layers in Fig. 7, where dominant attention attractor tokens

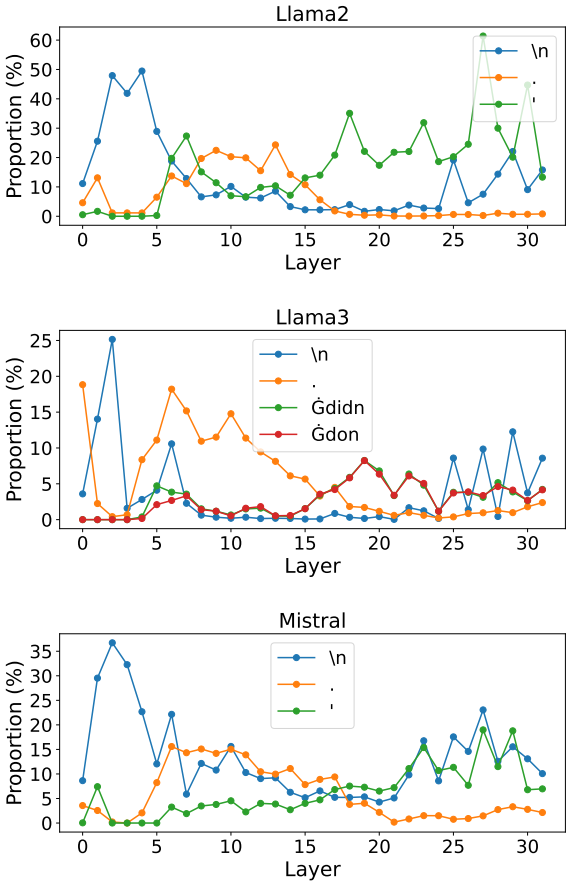


Figure 7: Proportion of frequent attention attractors across layers.

vary greatly in different layers.

A noteworthy phenomenon is that attention attractors often emerge from low-semantic tokens, such as stop words and punctuation marks. One possible explanation is that tokens with minimal intrinsic meaning serve as suitable placeholders for storing aggregated contextual information. This observation parallels findings in vision transformers (Darcet et al., 2024), where certain tokens (denoted as registers) receiving significantly high attention often correspond to background regions in images that are not directly relevant to the primary task.

3.3 Validation by Context Pruning

To validate our hypothesis that attention attractors often appear on low-semantic tokens, we conduct context pruning experiments. Specifically, we retain only low-semantic tokens in the key-value (KV) cache and assess the model's ability to maintain performance despite the reduced context.

Task and Baselines. We explore two tasks with context pruning. (1) The **long context pruning** task requires the model to compress a long text context into a shorter one and use the compressed contextual information for question answering. The context only needs to be compressed once. We use the LongBench dataset (Bai et al., 2024) for long context pruning evaluations. Following the suggestion they gave, we use the Lost in the Middle method (Liu et al., 2024) for data samples that are too long to exceed the context length of the models. We compare our simple strategy with the L2Norm method (Devoto et al., 2024), which selects tokens with the lowest L2 norm for context pruning, and the full KV cache baseline. (2) The **streaming context pruning** task requires the model to process a long streaming text input that can far exceed the model's context length. This involves dynamic pruning to ensure the compressed context fits the model's context window. Following Yao et al. (2024), we conduct experiments on DailyDialog. We compare our strategy with StreamingLLM (which employs a sliding context window) (Xiao et al., 2024) and SirLLM (which only retains high-entropy tokens) (Yao et al., 2024).

Setup. We design a pruning strategy that selectively retains only low-semantic tokens in long context tasks to explore their potential for preserving contextual information. To approximate "low semanticity", we take a heuristic approach based on word frequency in a large corpus—assuming that high-frequency words (e.g., "the") carry lower information entropy. Specifically, we select the top 256 most frequent tokens in the Fineweb Sample-10BT dataset (Penedo et al., 2024) as our target tokens. For long context pruning, in addition to keeping the selected frequent tokens, we also preserve the first four tokens as attention sinks as suggested by Xiao et al. (2024). This strategy yields about a 50% context length reduction. We apply the same compression ratio to L2Norm for a fair comparison. Note that the L2Norm method retains all tokens in the first two layers. We use Llama3-8b-Instruct, Llama2-7b-Chat, Llama2-13b-Chat and Qwen2-7b-Instruct (QwenTeam, 2024) as the base models to verify our hypothesis from different model sizes and different model architectures. For streaming context pruning, we adopt the SirLLM implementation but replace its token selection strategy with our simple frequency-based criterion. As in the long-context setting, we retain the first four tokens as attention sinks. Following SirLLM, we use Vicuna-7b (Chiang et al., 2023) for experiments with a fixed context length of 512.

Results. The results for long context pruning are

Table 1: Long context pruning performance on LongBench.

Method	Summarization			Few-shot Learning			Code		Single-Document QA			Multi-Document QA			Synthetic	
	GovReport	QMSum	MultiNews	TREC	TriviaQA	SAMSum	Lcc	RB-P	NrtvQA	Qasper	MF-en	HotpotQA	2WikiMQA	Musique	PCount	PRe
Llama3-8b-Instruct																
Full Cache	28.8	23.0	26.6	73.5	90.3	42.2	59.3	55.4	24.9	31.8	39.7	44.9	37.0	22.8	3.1	71.0
L2Norm	**24.8**	**21.3**	**25.0**	**63.5**	79.9	32.5	32.9	37.3	**19.3**	**15.3**	**29.1**	**32.4**	**29.2**	**15.7**	**5.0**	**70.5**
Ours	20.5	**21.3**	22.1	5.0	**89.4**	**39.4**	**41.6**	**40.1**	17.5	8.2	20.0	23.2	15.8	10.5	3.7	27.9
Llama2-7b-Chat																
Full Cache	26.7	21.0	26.0	0.0	20.9	6.6	14.8	17.7	17.8	25.6	34.3	25.4	28.1	8.4	3.0	6.5
L2Norm	**23.5**	16.4	**23.1**	**0.0**	2.7	3.8	**19.4**	**22.5**	4.6	7.4	15.3	4.7	8.1	1.2	2.8	2.6
Ours	20.5	**20.1**	22.1	**0.0**	**16.0**	**6.3**	13.1	15.2	**12.9**	**14.8**	**26.6**	**16.2**	**19.4**	**6.9**	**2.9**	**7.0**
Llama2-13b-Chat																
Full Cache	27.3	20.7	26.2	0.0	17.1	9.5	15.4	21.4	18.7	16.1	28.4	12.2	14.8	6.4	4.0	13.0
L2Norm	**24.2**	16.4	**24.5**	**0.0**	5.4	**9.2**	**10.1**	**11.5**	4.7	5.5	14.7	6.0	5.0	1.3	2.8	2.0
Ours	21.9	**20.0**	22.7	**0.0**	**23.8**	8.6	9.1	10.9	**10.0**	**9.6**	**21.9**	**11.0**	**13.6**	**3.4**	**4.2**	**9.0**
Qwen2-7b-Instruct																
Full Cache	36.4	23.4	26.9	77.0	89.8	44.7	63.3	61.2	25.8	43.6	45.7	13.3	13.4	8.9	4.5	76.0
L2Norm	**30.2**	**22.6**	**24.4**	**44.5**	**86.9**	44.9	26.3	27.4	**18.8**	**28.7**	**36.5**	11.8	**16.9**	6.9	6.0	13.0
Ours	22.4	20.8	20.4	5.5	85.0	**46.7**	**35.2**	**33.1**	10.9	19.2	25.6	**12.0**	11.9	**7.3**	**6.5**	**54.5**

Table 2: Streaming context pruning performance on DailyDialog.

Method	Vicuna-7b-v1.3	Vicuna-7b-v1.5
StreamingLLM	57.6	69.7
SirLLM	59.2	**70.1**
Ours	**59.4**	**70.1**

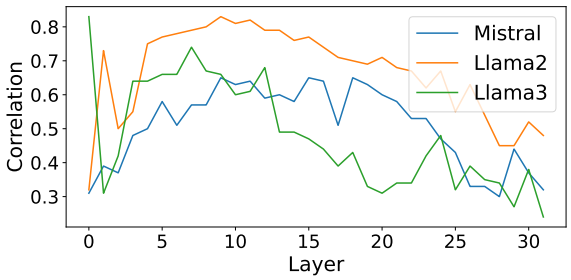


Figure 8: Layer-wise average of maximum absolute Spearman correlation between single-dimensional activation values and attention.

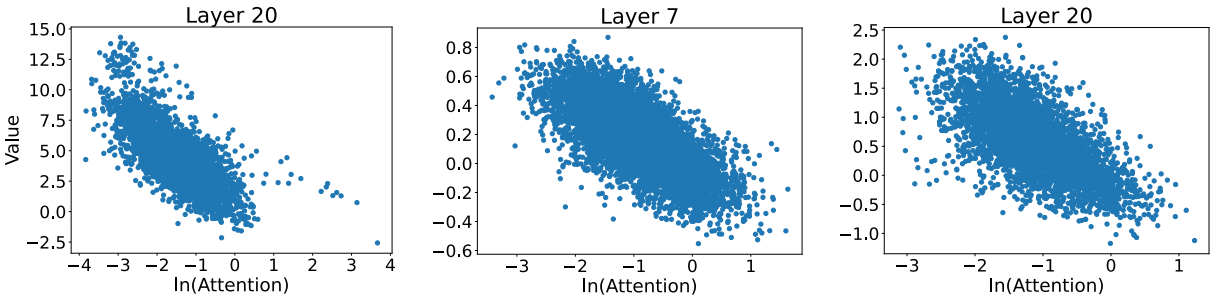
shown in Table 1. Despite using a straightforward strategy that preserves only frequent words in the KV cache and without relying on model states for token selection, our method performs comparably to L2Norm on many tasks. Interestingly, it is particularly effective for tasks with more casual inputs, such as summarization. However, we identify important exceptions. For instance, in the TREC benchmark (Li and Roth, 2002), classification label tokens (which carry critical task-specific information) frequently serve as attention attractors. In certain layers (e.g., Layers 18 and 23) of Llama3-8B-Instruct, over 40% of these label tokens act as attention attractors. Simply retaining only low-semantic tokens for context pruning would filter out these crucial labels, leading to a significant drop in task performance. This suggests that the token-wise distribution of attention attractors is complex and depends on the specific task and context.

Similarly, results for streaming context pruning (Table 2) show that our strategy achieves competitive performance against both StreamingLLM and the recently proposed SirLLM. These findings further validate the effectiveness of our token selection strategy underpinned by observations from attention attractors.

Takeaways: Low-semantic words are more likely to serve as attention attractors for information aggregation.

4 Mechanism of Attention Attractors

In previous sections, we explored the roles of attention attractors and their distributions in language models. However, it remains unclear how the models identify these attractors. In this section, we delve deeper into the underlying mechanism of attention attractors.



Llama2, dim 1076 Llama3, dim 2265 Mistral, dim 1935

Figure 9: Dimension activation values vs. attention of tokens.

4.1 Attention and Activations Correlation

Building on prior work (Sun et al., 2024; Devoto et al., 2024; Cancedda, 2024), we investigate how attention allocation relates to individual activation dimensions in transformer models. Sun et al. (2024) find that certain tokens receiving high attention tend to have sharply peaked activations (referred to massive tokens). We hypothesize a more generalizable correlation between attention weights and specific activation dimensions.

Setup. We analyze three models—Llama2-7b, Llama3-8b, and Mistral-7b. Following the preprocess method of Sun et al. (2024), we sample 10 inputs from the PG-19 dataset (Rae et al., 2020). For each model and each layer, we calculate Spearman's rank correlation coefficient between attention weights and per-dimension activation values. We exclude results with low statistical significance ($p < 0.05$) to ensure analytical reliability.

Findings. As shown in Fig. 8, we observe that in shallow layers, attention weights can correlate strongly with even a single activation dimension, potentially implying that such dimensions may serve as salient indicators for attention mechanisms. We illustrate this phenomenon with several examples in Fig. 9. However, this correlation gradually weakens in deeper layers, suggesting that attention at later stages is likely governed by more complex, multi-dimensional representations rather than any single dominant dimension.

4.2 Validation by Context Pruning

Based on the observation that attention weights correlate with token activations, we further hypothesize that activation patterns in specific dimensions are predictive of attention attractors and can be leveraged to select important tokens for long context pruning.

Setup. We design a pruning strategy that retains tokens layer-wise based on their activations along a specific single dimension—identified through correlation with attention weights. Specifically, for each layer, we compute Spearman's rank correlation between attention weights and individual dimension activations across 400 randomly sampled examples from the Infinity-Instruct dataset (BAAI, 2024), following the methodology described in Section 4.1. We then select 2,044 tokens per layer with the highest activation values in the most correlated dimension. Consistent with prior work (Xiao et al., 2024; Devoto et al., 2024), we additionally preserve the first four tokens (to serve as attention sinks) and retain all tokens in the first two layers. To ensure a fair comparison, we apply the same compression ratio to the L2Norm pruning baseline.

Results. As shown in Table 3, our dimension-based pruning strategy, relying solely on a single activation dimension per layer, achieves performance comparable to L2Norm across multiple tasks. This supports the utility of single-dimension activations as a proxy for identifying attention-critical tokens. However, the performance of such a simple strategy is not consistently robust across all benchmarks. On certain tasks, pruning based solely on single-dimension activations leads to significant performance degradation and falls short of the L2Norm baseline. We hypothesize that L2Norm's head-wise token selection introduces additional flexibility, potentially enhancing pruning stability and improving information retention.

Takeaways: Specific dimension activations indicates attention allocation.

5 Related Work

Observing and interpreting attention attractors. Tokens that attract significantly high attention in language models are a common focus of studies. For instance, Clark et al. (2019) observe specific

Table 3: Long context pruning performance on LongBench.

Method	Summarization			Few-shot Learning			Code		Single-Document QA			Multi-Document QA			Synthetic	
	GovReport	QMSum	MultiNews	TREC	TriviaQA	SAMSum	Lcc	RB-P	NrtvQA	Qasper	MF-en	HotpotQA	2WikiMQA	Musique	PCount	PRe
Llama3-8b-Instruct																
Full Cache	28.8	23.0	26.6	73.5	90.3	42.2	59.3	55.4	24.9	31.8	39.7	44.9	37.0	22.8	3.1	71.0
L2Norm	23.3	20.7	25.9	**50.0**	**79.9**	**30.6**	49.5	39.7	17.4	15.2	**30.3**	28.6	**29.2**	**11.3**	**6.8**	**64.5**
Ours	**24.6**	**20.9**	**26.3**	36.5	79.8	30.1	**57.0**	**56.5**	**18.8**	**17.9**	29.9	**32.9**	32.3	15.3	2.9	48.9
Llama2-7b-Chat																
Full Cache	26.7	21.0	26.0	0.0	20.9	6.6	14.8	17.7	17.8	25.6	34.3	25.4	28.1	8.4	3.0	6.5
L2Norm	24.0	**16.6**	24.9	**0.0**	**3.8**	3.7	**19.0**	21.5	3.9	7.1	15.1	4.7	8.9	1.2	**3.8**	**3.6**
Ours	**26.2**	16.4	**25.6**	**0.0**	3.6	**6.0**	17.1	**22.1**	**4.3**	**9.4**	**15.6**	**6.4**	**11.9**	**1.7**	3.4	1.2
Llama2-13b-Chat																
Full Cache	27.3	20.7	26.2	0.0	17.1	9.5	15.4	21.4	18.7	16.1	28.4	12.2	14.8	6.4	4.0	13.0
L2Norm	24.5	16.6	**25.8**	**0.8**	4.3	**9.2**	**16.7**	18.9	3.4	5.3	**14.8**	5.4	**7.2**	1.0	**2.5**	2.0
Ours	**25.8**	**16.6**	**25.8**	0.0	**4.4**	9.1	16.6	**19.7**	**4.6**	**6.1**	12.0	**6.4**	5.3	**2.7**	2.2	**2.5**
Qwen2-7b-Instruct																
Full Cache	36.4	23.4	26.9	77.0	89.8	44.7	63.3	61.2	25.8	43.6	45.7	13.3	13.4	8.9	4.5	76.0
L2Norm	25.0	20.3	26.5	**39.0**	78.6	**46.0**	51.0	29.5	11.5	25.6	30.9	**23.9**	**20.8**	**11.3**	**6.0**	**6.5**
Ours	**30.7**	**22.9**	**27.4**	8.5	**86.3**	45.0	**59.1**	**49.7**	**15.6**	**37.5**	**34.4**	10.9	12.2	7.2	3.5	5.5

attention heads in the BERT model (Devlin et al., 2019) that consistently assign high attention to the [SEP] token, describing them as "no-op" operations. Similarly, Xiao et al. (2024) identify initial tokens in sequences that receive high attention and label them as *attention sinks*, which help language models manage excessive attention loads. Yu et al. (2024) note that tokens conveying minimal semantic content attract unusually high attention in intermediate layers, and reducing these weights can enhance performance. Further observations include PyramidKV (Cai et al., 2024), which note a shift in attention patterns from a broad-spectrum to a more focused mode on key tokens as layers deepen. H2O (Zhang et al., 2023) links high attention to token co-occurrence frequency, noting performance declines when such co-occurrences are removed. Sun et al. (2024) correlate massive activations in large models with high attention to specific tokens, suggesting these activations act as crucial bias terms. L2Norm (Devoto et al., 2024) and others like StreamingDialogue (Li et al., 2024) and OPERA (Huang et al., 2024) observe relationships between attention weights, token norms, and their capacity to aggregate contextual information. Expanding beyond language models, Darcet et al. (2024) discovered *register tokens* in vision transformers (Dosovitskiy et al., 2021) located in task-irrelevant background areas but attracting abnormally high attention. Building on these foundational studies, we aim to provide a more comprehensive observation and explanation of attention attractors in large language models.

Utilizing attention attractors. In addition to observing and interpreting attention attractors, numerous works have explored their potential for enhancing downstream applications. For example, StreamingLLM (Xiao et al., 2024) introduces a sliding window attention mechanism for streaming inputs, leveraging the unique roles of attention sinks. Yu et al. (2024) and Huang et al. (2024) focus on improving model accuracy and mitigating hallucinations by reducing the model's over-reliance on attention attractors. Furthermore, context pruning methods use high-attention tokens as key tokens to retain crucial contextual information, improving both the efficiency of key-value (KV) cache management and inference speed (Liu et al., 2023; Zhang et al., 2023; Cai et al., 2024; Ge et al., 2024). These strategies help streamline model computations while maintaining performance. We hope that our findings will inspire further research to leverage attention attractors for more efficient and effective applications in downstream tasks.

6 Conclusion

This work provides a comprehensive exploration of attention attractors in large language models,

yielding several key findings: (1) We confirm that attention attractors act as information aggregators in language models, as evidenced by our detailed information flow analysis. (2) Attention attractors are predominantly found in the initial and middle layers of models and are often linked to low-semantic tokens. (3) The attention allocated to specific tokens correlates strongly with some activation dimension values, which can be further leveraged to predict attention attractors. We hope that these findings enhance the understanding of attention mechanisms in large language models and inspire further research.

Limitations

While our study provides a comprehensive exploration of attention attractors, it is confined to the Llama, Mistral, and Qwen models with relatively small parameter sizes. This limitation may introduce potential bias and affect the generalizability of our findings. To address this, we plan to extend our investigation to a broader range of models, including those with varying sizes and architectures, to ensure that our insights are applicable across different model types and scales.

Ethics Statement

This work aims to explore the attention attractors in large language models to enhance their transparency and explainability. We do not foresee any significant ethical concerns. The models and datasets utilized are publicly accessible and widely used, meaning our findings may reflect any limitations or biases inherent in these resources.

Acknowledgment

This work was partially supported by the National Natural Science Foundation of China (No. 62576347).

References

BAAI. 2024. Infinity instruct.

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. 2024. 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.

Zefan Cai, Yichi Zhang, Bofei Gao, Yuliang Liu, Tianyu Liu, Keming Lu, Wayne Xiong, Yue Dong, Baobao Chang, Junjie Hu, and Wen Xiao. 2024. Pyramidkv: Dynamic kv cache compression based on pyramidal information funneling.

Nicola Cancedda. 2024. Spectral filters, dark signals, and attention sinks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4792–4808, Bangkok, Thailand. Association for Computational Linguistics.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. 2024. Vision transformers need registers. In *The Twelfth International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Alessio Devoto, Yu Zhao, Simone Scardapane, and Pasquale Minervini. 2024. A simple and effective l_2 norm-based strategy for KV cache compression. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18476–18499, Miami, Florida, USA. Association for Computational Linguistics.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Suyu Ge, Yunan Zhang, Liyuan Liu, Minjia Zhang, Jiawei Han, and Jianfeng Gao. 2024. Model tells you what to discard: Adaptive KV cache compression for LLMs. In *The Twelfth International Conference on Learning Representations*.

Aaron Grattafiori, Abhimanyu Dubey, and Abhinav Jauhri et al. 2024. The llama 3 herd of models.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. 2024. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13418–13427.

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

Jianan Li, Quan Tu, Cunli Mao, Zhengtao Yu, Ji-Rong Wen, and Rui Yan. 2024. Streamingdialogue: Prolonged dialogue learning via long context compression with minimal losses. In *Advances in Neural Information Processing Systems*, volume 37, pages 86074–86101. Curran Associates, Inc.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

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

Zichang Liu, Aditya Desai, Fangshuo Liao, Weitao Wang, Victor Xie, Zhaozhuo Xu, Anastasios Kyrillidis, and Anshumali Shrivastava. 2023. Scissorhands: Exploiting the persistence of importance hypothesis for llm kv cache compression at test time. In *Advances in Neural Information Processing Systems*, volume 36, pages 52342–52364. Curran Associates, Inc.

OpenAI. 2024. Gpt-4 technical report.

Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi, Nikita Nangia, Jason Phang, Angelica Chen, Vishakh Padmakumar, Johnny Ma, Jana Thompson, He He, and Samuel Bowman. 2022. QuALITY: Question answering with long input texts, yes! In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5336–5358, Seattle, United States. Association for Computational Linguistics.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

QwenTeam. 2024. Qwen2 technical report.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. 2020. Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Workshop Track Proceedings*.

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

Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. 2024. Massive activations in large language models. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*.

Hugo Touvron, Louis Martin, and Kevin Stone et al. 2023. Llama 2: Open foundation and fine-tuned chat models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023. Label words are anchors: An information flow perspective for understanding in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9840–9855, Singapore. Association for Computational Linguistics.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2024. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*.

Yao Yao, Zuchao Li, and Hai Zhao. 2024. SirLLM: Streaming infinite retentive LLM. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2611–2624, Bangkok, Thailand. Association for Computational Linguistics.

Zhongzhi Yu, Zheng Wang, Yonggan Fu, Huihong Shi, Khalid Shaikh, and Yingyan Celine Lin. 2024. Unveiling and harnessing hidden attention sinks: Enhancing large language models without training through attention calibration. In *Forty-first International Conference on Machine Learning*.

Zihao Yue, Liang Zhang, and Qin Jin. 2024. Less is more: Mitigating multimodal hallucination from an EOS decision perspective. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11766–11781, Bangkok, Thailand. Association for Computational Linguistics.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, Zhangyang "Atlas" Wang, and Beidi Chen. 2023. H2o: Heavy-hitter oracle for efficient generative inference of large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 34661–34710. Curran Associates, Inc.

Appendix

A Additional Results

A.1 Information Flow Analysis

To validate the generalizability of our findings, we performed the same experiment on Llama3-8b and Mistral-7b, adhering to the setup in Section 2.2. The results, as illustrated in Fig. 10 and Fig. 11, confirm that our conclusions hold true across different models.

A.2 Context Pruning

We illustrate some examples of the text input before and after pruning using our strategy described in Section 3.3 in Fig. 12. Although the pruned text loses the majority of tokens containing key information, it largely retains model performance, further validating the role of attention attractors as effective information aggregators.

A.3 Additional Analysis on Attention and Activation Dimensions

Attention scores vs. activation values. To further illustrate the correlation between attention allocation and activation dimension values, we partition tokens into four groups based on their specific dimension values, and independently measure the average attention scores received by each group of tokens. We sample 100 instances from the PG-19 dataset for analysis. As shown in Fig. 13, tokens with smaller activation values in the specified dimension obviously receive higher attention weights across almost all layers, further validating the correlation between attention weights and activation dimension values.

Intervention experiment. To further investigate whether the observed correlation has a causal impact on model behavior, we conduct a series of controlled intervention experiments. Specifically, we sample 100 instances from the PG-19 dataset and evaluate perplexity using Llama3-8B-Instruct under four different intervention settings applied at each layer: (1) zeroing out the most correlated activation dimension; (2) replacing the most correlated activation dimension with random values; (3) zeroing out a randomly selected activation dimension; and (4) replacing a randomly selected activation dimension with random values.

Our results demonstrate that interventions tar-

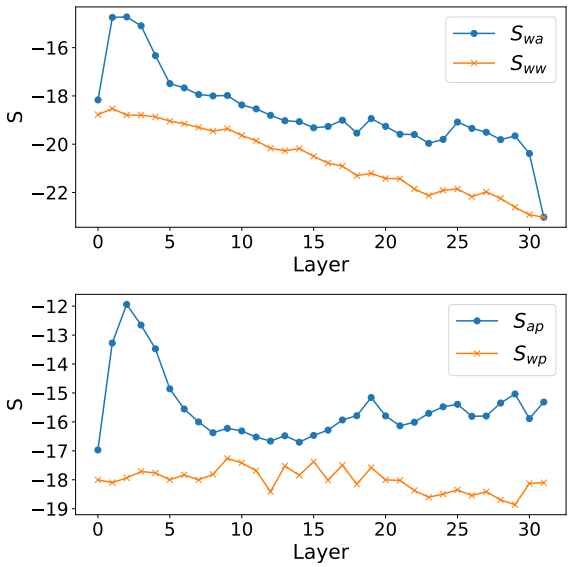


Figure 10: Layer-wise average significance of information flow between attention attractors (a), prediction positions (p), and other tokens (w) in Llama3-8b.

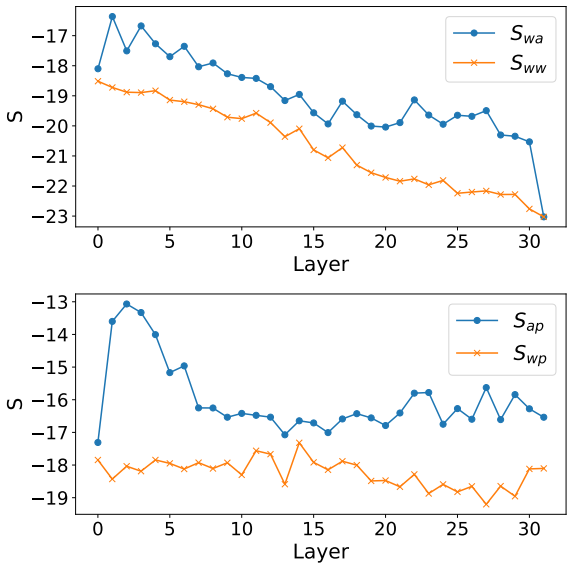


Figure 11: Layer-wise average significance of information flow between attention attractors (a), prediction positions (p), and other tokens (w) in Mistral-7b.

BEF: Mercurian night settled black and thick over the Q City Spaceport. Tentative fingers of light flicked and probed the sky, and winked out.
AFT: and over the. T of fed anded the, and wed out.

BEF: "Here she comes," somebody in the line ahead said.
AFT: "Here she in the said.

BEF: Shano coughed, his whole skeletal body jerking. Arthritic joints sent flashes of pain along his limbs. Here she comes, he thought, feeling neither glad nor sad.
AFT: Sh ced, hisal jing. Arthicses of his. Here she, he,.

BEF: He coughed and slipped polarized goggles over his eyes.
AFT: He ced and gg over his.

BEF: The spaceport emerged bathed in infra red. Hangars, cradles, freighter catapults and long runways stood out in sharp, diamond-clear detail. High up, beyond the cone of illumination, a detached triple row of bright specks—portholes of the liner Stardust—sank slowly down.
AFT: The b inra. H,ad,s and long out in,-. High up, the of, a of—th of the l St—s down.

Figure 12: Context before and after pruning using our strategy.

geting the highly correlated dimensions precipitate a degradation in language modeling performance, increasing perplexity from 13.2 to 14.3 and 14.4 for (1) and (2), respectively. Conversely, identical interventions applied to randomly selected control dimensions yield negligible deviations (resulting in perplexities of 13.2 and 13.3 for (3) and (4)). These findings substantiate the premise that dimensions exhibiting strong correlations with attention weights play a functional role in model predictions. Nevertheless, the magnitude of the performance drop remains bounded. We postulate that this robustness is attributable to representational redundancy distributed across highly correlated dimensions within a given layer, a phenomenon that corroborates the observations detailed in our preceding analyses.

Multi-head analysis. We further examine the relationship between attention weights and activation dimension values at the level of individual attention heads. Specifically, we analyze the key vectors of each head before applying Rotary Positional Embedding (RoPE) (Su et al., 2024). As shown in Fig. 14, individual heads can exhibit a more pronounced correlation between attention weights and activation values. This observation reinforces our findings in Section 4 and highlights the potential for head-specific optimization in context pruning strategies.

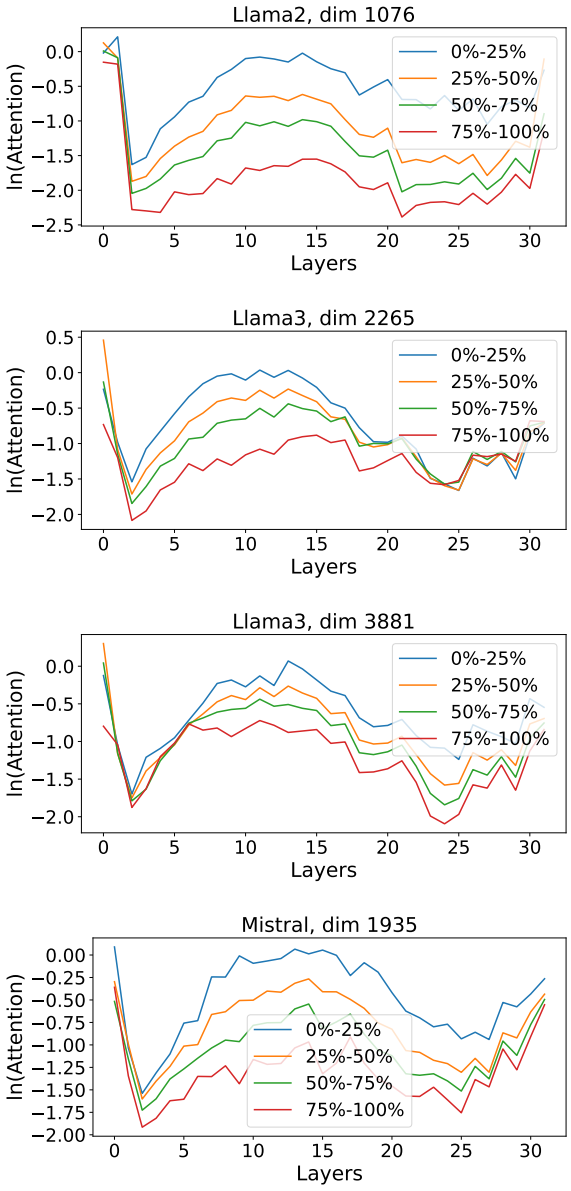


Figure 13: Layer-wise average attention across different groups.

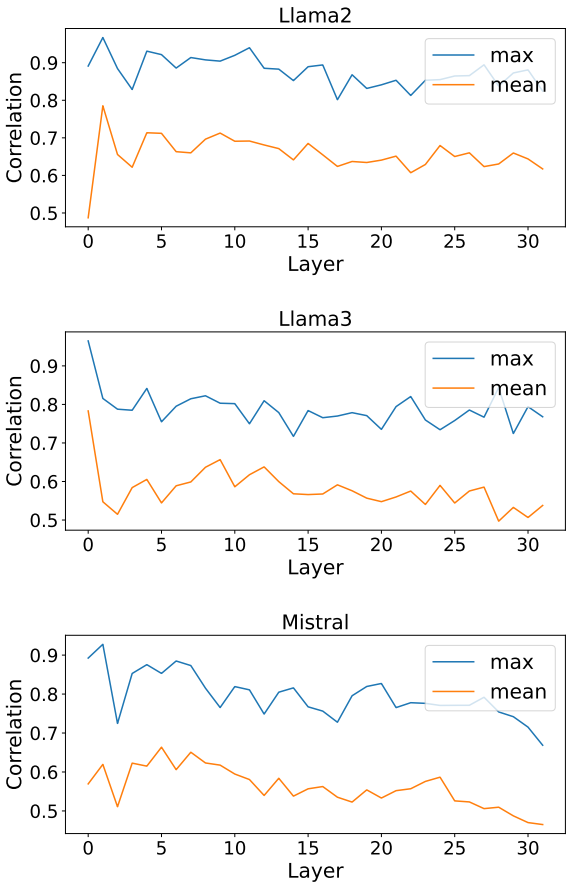


Figure 14: Layer-wise head-averaged maximum and average absolute Spearman correlation between single-dimensional activation values and attention.