Cooperative or Competitive? Understanding the Interaction between Attention Heads From A Game Theory Perspective

Xiaoye Qu[1*], Zengqi Yu[1*], Dongrui Liu[2], Wei Wei[1†], Daizong Liu[3], Jianfeng Dong[4], Yu Cheng[5]

[1]Cognitive Computing and Intelligent Information Processing (CCIIP) Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology
[2]Shanghai AI Laboratory [3]Peking University [4]Zhejiang Gongshang University
[5]The Chinese University of Hong Kong
xiaoye@hust.edu.cn, weiw@hust.edu.cn

Abstract

Despite the remarkable success of attention-based large language models (LLMs), the precise interaction mechanisms between attention heads remain poorly understood. In contrast to prevalent methods that focus on individual head contributions, we rigorously analyze the intricate interplay among attention heads through a novel framework based on the Harsanyi dividend, a concept from cooperative game theory. Our analysis reveals that significant positive Harsanyi dividends are sparsely distributed across head combinations, indicating that most heads do not contribute cooperatively. Moreover, certain head combinations exhibit negative dividends, indicating implicit competitive relationships. To further optimize the interactions among attention heads, we propose a training-free Game-theoretic Attention Calibration (GAC) method. Specifically, GAC selectively retains heads demonstrating significant cooperative gains and applies fine-grained distributional adjustments to the remaining heads. Comprehensive experiments across 17 benchmarks demonstrate the effectiveness of our proposed GAC and its superior generalization capabilities across diverse model families, scales, and modalities. The source code is available at: https://github.com/queng12322/GAC.

1 Introduction

Recently, Large language models (LLMs) (Achiam et al., 2023; Bai et al., 2023; Dubey et al., 2024) have gained considerable attention for their impressive performance across diverse tasks. A critical component enabling this performance is the attention mechanism (Vaswani et al., 2017), which effectively captures relationships between tokens.

Previous works mainly focus on studying the role of each individual attention head. As shown in Figure 1(a), these works (Voita et al., 2019; Behnke and Heafield, 2020; Wang et al., 2021; Zayed et al., 2024; Zhou et al., 2024; Jin et al., 2024) assess the contribution of a single attention head via pruning it. Nevertheless, the inference process for specific tasks relies on the collective operation of multiple heads. It raises the fundamental question: *What is the interplay among attention heads? Cooperative or competitive?* A thorough understanding of the interaction among attention heads is crucial for both optimizing model performance and demystifying the black-box nature of LLMs.

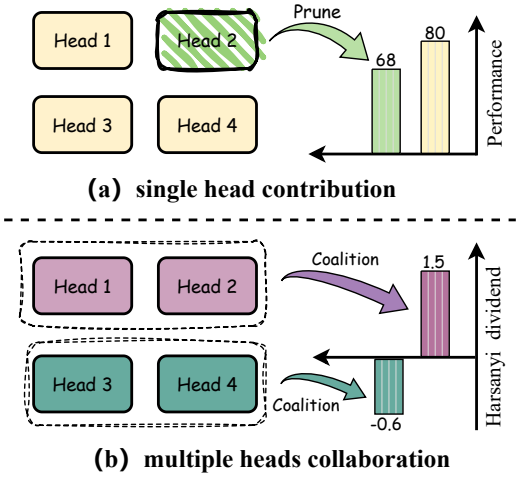

Figure 1: (a) Previous works study the contribution of each single attention head in isolation. (b) We model attention heads as players in a cooperative game process, utilizing the Harsanyi dividend to capture the benefits of collaborations among multiple attention heads.

We investigate this problem from the perspective of game theory. As shown in Figure 1(b), we consider the prediction process of the LLM as a game and attention heads as players. Then, we utilize the Harsanyi dividend (Harsanyi, 1982) to quantify the interactions for each attention head coalition. Harsanyi dividend was originally proposed in game theory to measure the interactions

*Equal contribution.
†Wei Wei is the corresponding author.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 14079–14099
July 27 - August 1, 2025

between different players, which makes itself a natural metric to dissect the compositional capability of attention heads. Besides, it theoretically satisfies the efficiency, linearity, dummy, and symmetry axioms, which ensures the trustworthiness of the interpretations for our analysis. Conceptually, let p_1, p_2, and p_3 be the gain of the game with only head 1, only head 2, and both, respectively. The Harsanyi dividend (p_3 - p_1 - p_2) denotes the added gain from combining two heads. Here a positive dividend suggests that cooperation exists among these heads, while a negative value indicates implicit competition.

Through an in-depth analysis of the Harsanyi distribution, we reveal its sparsity, where only a small fraction of attention head coalitions yield substantial dividends. A large number of coalitions demonstrate near-zero dividends, implying a lack of cooperative effect. Furthermore, negative dividends observed in specific coalitions highlight inherent conflicts among alliance members. These observations naturally lead to the question: *Can we optimize these interactions among attention heads to improve the capability of attention layers?*

To this end, we develop a training-free Game-theoretic Attention Calibration (GAC). Specifically, we first identify the salient group of attention heads that exhibit significant positive Harsanyi dividends. Subsequently, to alleviate the competition between attention heads that results in negative Harsanyi dividends, we smooth the excessive attention weight allocations in attention heads outside the salient group by applying fine-grained distributional adjustments. This approach yields a significant improvement in the overall Harsanyi dividend.

To comprehensively evaluate the effectiveness and generalization of our GAC, we conduct experiments across both LLM and MLLM, utilizing a diverse set of 17 benchmark datasets. Our proposed GAC can achieve up to a 10% higher accuracy than the vanilla inference across eight text classification benchmarks. Moreover, our method effectively promotes inter-head collaboration with an increased overall Harsanyi dividend.

To sum up, we make the following contributions:

- To the best of our knowledge, we are the first to model attention heads with a multivariate cooperative game process and adopt the Harsanyi dividend to investigate the cooperative mechanisms among them.
- From the perspective of the game theory, we propose a training-free Game-theoretic Attention Calibration (GAC) for optimizing the interactions between attention heads.
- Extensive experiments on 17 benchmark datasets, encompassing both LLMs and MLLMs, validate the efficacy and generalizability of our proposed GAC method across diverse model families, scales, and modalities.

2 Related Work

Game Theory. The fundamental principle of game theory is to allocate payoffs to participants in a fair and reasonable manner. It is formally constituted by a set of players and a characteristic function (Chalkiadakis et al., 2022). The characteristic function assigns to each coalition of players a real-valued quantity representing the collective payoff achievable by those players when collaborating to accomplish a given task. Game theory has found many applications in the field of model interpretability (Jin et al., 2023; Liu et al., 2023a; Wang et al., 2024; Fang et al., 2024), but there is little exploration of the attention mechanisms. Harsanyi dividend (Harsanyi, 1982) is first proposed in game theory to measure the interplay between different players. This paper adopts it to analyze the interactions between attention heads as it satisfies many mathematical axioms as described in Appendix A.

Pruning Attention Head. Michel et al. (2019) evaluates the importance of each individual head by pruning it. They observe that a large percentage of attention heads can be removed at test time without significantly impacting performance. Studies have further shown that certain heads play specialized roles (Clark et al., 2019b; Qu et al., 2020; Zhou et al., 2024; Zayed et al., 2024; Qu et al., 2024b). In this paper, instead of investigating the contribution of a single attention head, we make the first attempt to study the interaction between attention heads. Furthermore, we propose a training-free optimization method for these interactions that improves overall model capability. This contrasts with previous pruning methods, which primarily focused on performance preservation.

Attention Sink. StreamLLM (Xiao et al., 2023) first identifies the presence of the attention sink phenomenon, which refers to a token that receives disproportionately high attention scores but contributes limited semantic information. StreamLLM notes that attention sink is typically found only in

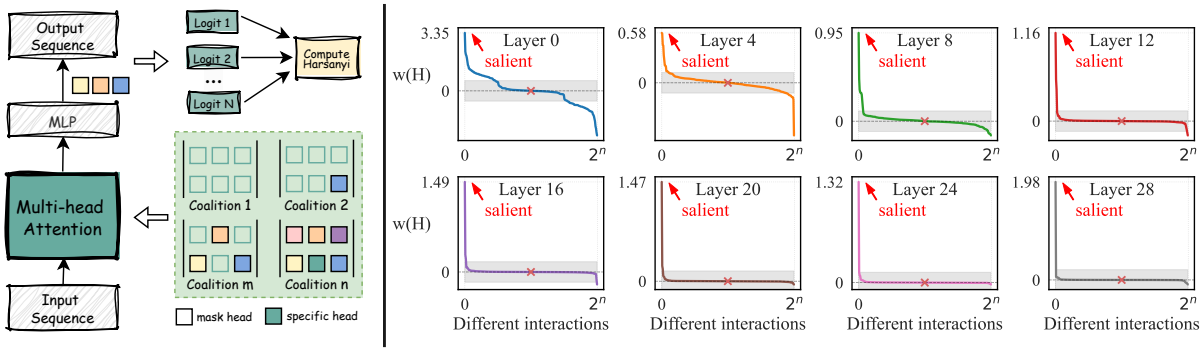

Figure 2: (a) We first mask partial attention heads to form different coalitions. Subsequently, we compute the Harsanyi dividend for each coalition to investigate the interaction among attention heads. (b) The distribution of the Harsanyi dividend for each coalition on each layer. For clarity, we sort the strength of interaction effects in descending order. More layers are demonstrated in Figure 7 of the Appendix.

the initial token. Subsequently, ACT (Yu et al., 2024) discovers that attention sinks occur not only at the start of sequences but also within later tokens. This attention sink is then used for KV cache optimization (Wan et al., 2024; Wu and Tu, 2024), efficient inference (Zhang et al., 2024), and other applications (Liu et al., 2024b,a,c, 2022; Zhu et al., 2023; Tao et al., 2023; Qu et al., 2024a, 2025, 2024c, 2023; Guo et al., 2024; Dong et al., 2022, 2024). In this paper, we smooth the excessive attention weights to alleviate the inter-head competition.

3 Analyzing Interactions

In this paper, we analyze the intricate interplay between attention heads with the Harsanyi dividend. We first briefly introduce the attention mechanism in Section 3.1 and the Harsany dividend in Section 3.2. Then, we use the Harsany dividend to quantify the interaction among different combinations of attention heads (Section 3.3).

3.1 Attention Mechanism

LLMs typically consist of L transformer blocks, each comprising a feed-forward network (FFN) and a multi-head attention (MHA) module that captures the pairwise relationships among all N input tokens in an input sequence. Specifically, for a given input $\mathbf{X} \in \mathbb{R}^{N\times d}$ to the l-th block (here we omit layer notations), the output feature of MHA can be represented as:

$$\begin{aligned} \text{MHA}(X) &= HW_o, H = (h_1 \oplus h_2 \oplus \cdots \oplus h_n) \\ h_i &= A_i W_v^i,\ A_i = Softmax\left(\frac{W_q^i W_k^{iT}}{\sqrt{d/n}}\right) \end{aligned} \tag{1}$$

where $W_q^i = f_q^i(X)$, $W_k^i = f_k^i(X)$, $W_v^i = f_v^i(X)$, f_q^i, f_k^i, and f_v^i are projection layers. $\mathbf{A}_i \in \mathbb{R}^{N\times N}$ is the attention map generated at head h_i. $\oplus$ represents concatenation.

3.2 The Harsanyi dividend

The Harsanyi dividend (Harsanyi, 1982) is a classical concept in game theory used to measure the interactions among players. Given a set of players $N = \{1, 2, \ldots, n\}$ participating in a game v, certain rewards can be achieved. The function $v(\cdot)$ maps any subset of players $S \subseteq N$ to a real number, representing the reward obtained by that subset.

During a game of Harsanyi dividend, it is important to note that players typically do not contribute to the reward independently, instead, they interact with one another, forming various coalitions or patterns that collectively affect the outcome.

Definition 1. Harsany Dividend: *Given a set of participants $N = \{1, 2, ..., n\}$, and a value function $v(S)$, the Harsanyi dividend $w(S|N)$ for a specific coalition $S \subseteq N$ is defined as*:

$$w(S|\mathcal{N}) = \sum_{S'\subseteq S} (-1)^{|S'|-|S|} \cdot v(S'). \tag{2}$$

It quantifies the marginal contribution of the elements in S to all possible coalitions that can be formed within S. $v(S')$ is the characteristic function, representing the value achieved by the subset S' when its members cooperate.

3.3 Analyzing the Interactions of Attention Heads with Harsanyi dividend

To analyze the interaction between attention heads, we take attention heads as players and adopt the

Harsanyi dividend to analyze the interactions between them. Formally, we represent the characteristic function $v(\cdot)$ as the normalized prediction logits of classifying the sample x to the ground-truth category following (Liu et al., 2023a). Notably, using accuracy or logits directly as the characteristic function leads to poor discriminability.

$$v(\mathbf{x}) = \log\left(\frac{p(y = y_{\text{truth}}|\mathbf{x})}{1 - p(y = y_{\text{truth}}|\mathbf{x})}\right) \quad (3)$$

To construct combinations of different attention heads, as shown in Figure 2(a), we propose to mask the specific attention head in Eq. 1 with the undifferentiated attention (Zhou et al., 2024):

$$\hat{h}_i = Softmax(\frac{\epsilon W_q^i W_k^{iT}}{\sqrt{d/n}})W_v^i \quad (4)$$

where a very small coefficient ϵ ($1e^{-5}$) is multiplied with the parameter matrix. It hinders the head from extracting the critical information from the input sequence and degenerates the attention weight to the mean value. We have explored other methods for ablating attention heads and compared them in Table 7 of the Appendix.

Subsequently, we mask partial attention heads to form a coalition $\hat{H}$ and exploit the Harsanyi dividend to quantitatively measure the interaction on the characteristic function in Eq. 3:

$$\begin{aligned} \hat{H} &= (h_1 \oplus \hat{h}_i \oplus \hat{h}_j \oplus h_n) \\ w(H|\hat{H}) &= \sum_{H \subseteq \hat{H}} (-1)^{|H|-|\hat{H}|} \cdot v(\hat{H}) \end{aligned} \quad (5)$$

As shown in Figure 2(b), we sort the strength of interaction effects in descending order for each layer, we can observe: (1) **Only a small number of coalitions have significant positive effects** $w(\hat{H}|H)$, whose Harsanyi dividend is significantly larger than 0. These interactions are sparse throughout the distribution. (2) **Most coalitions have almost zero effect**, i.e, $w(\hat{H}|H) \approx 0$. It means that most combinations do not contribute cooperatively. (3) **Intriguingly, some coalitions lead to negative Harsanyi values**, demonstrating specific attention heads exhibit implicit competition with each other. This new observation may help explain phenomena like knowledge conflict (Su et al., 2024) or catastrophic forgetting (Luo et al., 2023) in the LLM.

It is intriguing that the sparse Harsanyi distribution of attention heads in Figure 2(b) is similar to the concept-emergence phenomenon (Ren et al., 2024, 2023; Liu et al., 2023a) proposed in explaining DNN, where only a small number of subsets Ω_{salient} make salient interaction effects on the network output, and can be considered as interactive concepts. For these interactive concepts in DNNs, there are the below theorem:

Theorem 1. *Given an input sample $x \in \mathbb{R}^n$ with n variables indexed by $N = \{1, ..., n\}$, there are 2^n different masked samples x_T, for specific set H, $H \subseteq N$, Ren et al. (2024) has proven*:

$$\begin{aligned} \forall T \subseteq N, v(\boldsymbol{x}_T) &= \sum_{H \subseteq T} w(H|\boldsymbol{x}) \\ &\approx \sum_{H \in \Omega_{\text{salient}} \& H \subseteq T} w(H|\boldsymbol{x}) \end{aligned} \quad (6)$$

The first part of the equation adheres to the efficiency axiom of Harsanyi dividend (Harsanyi, 1982), introduced in Appendix A. The second part means that the interactive concepts in Ω_{salient} can well approximate network outputs on anyone x_T of the 2^n masked samples. Therefore, the subsets leading to salient interactions are most important for the predictive capability of the model as $v(\boldsymbol{x}) \approx \sum_{H \in \Omega_{\text{salient}}} w(H|\boldsymbol{x})$.

4 Game-theoretic Attention Calibration

Following our above analysis, which reveals the sparsely distributed significant Harsanyi dividends and the presence of negative dividends, we seek to optimize the interactions to promote inter-head collaboration and improve model capability.

To this end, we propose a training-free Game-theoretic Attention Calibration (GAC) method. Inspired by the principles outlined in Theorem 1, we propose to selectively retain heads demonstrating salient cooperative gains. Moreover, to alleviate the competition between attention heads that results in negative Harsanyi dividends, we apply fine-grained distributional adjustments to the remaining heads.

4.1 Identify Salient Group

In this section, we aim to identify the attention heads that create salient interactions and preserve these heads. Notably, it is challenging to identify a combination in which all possible interactions are salient. Therefore, we propose two approximation methods and will compare them in the ablation study. In the first method, we simply adopt the maximum Harsanyi to identify a salient group h containing multiple heads h_i:

$$G(h) = \text{argmax}(w(\hat{H}|H)) \tag{7}$$

Besides using a single maximum Harsany value, in the second method, we select all combinations whose Harsanyi values are positive and then identify the maximum common subset within this set.

$$G'(h) = \underset{w(H_i|\hat{H})>0}{\arg\max} \left|\bigcap \hat{H}_i\right| \tag{8}$$

4.2 Calibrate Attention Distribution

After identifying the salient group, we redistribute attention scores among the heads outside of the salient group, aiming to alleviate competition between attention heads within and outside the salient group. Specifically, we first identify focused tokens with excessive attention scores. Subsequently, the attention scores of these focused tokens will be diminished and the attention weights of the remaining tokens will accordingly increase.

Formally, for the attention map of the i-th head $\mathbf{A}_i$ in Eq. 1, $\mathbf{A}_i[m,n]$ denotes the relationship between the m-th and n-th tokens. We define the average attention score as below:

$$a_i = [\sum_{n=1}^{m} \mathbf{A}_i[m,n]/m, \forall m \in \{2,...,N\}] \tag{9}$$

Here, $a_i[m]$ denotes the average attention score of the m-th token at i-th head. We do not adjust the first token considering its importance (Xiao et al., 2023). We then adjust the m-th token with significantly high average scores:

$$S_i = \{t \in \{2,...,N\} | a_i[t] > \alpha\} \tag{10}$$

where α is a threshold hyperparameter deciding the focused tokens. Subsequently, we diminish the attention score in i-th head for all $s \in S_i$ for each row k in attention map A_i:

$$\widehat{A}_i[k,s] = A_i[k,s] \times \beta \tag{11}$$

where β is a hyperparameter controlling how much we diminish the excessive attention scores. Finally, we slightly increase the attention scores of each other tokens $s \notin S_i$ and ensures that the sum of each row k remains one:

$$\widehat{A}_i[k,t] = A_i[k,t] + \sum_{s \in S_i} (A_i[k,s] - \widehat{A}_i[k,s]) \times \frac{A_i[k,t]}{\sum_{i \in \{1,...,N\}-S_i} A_i[k,i]} \tag{12}$$

Algorithm 1 Game-theoretic Attention Calibration (GAC)

Input: $A = \{A_1, A_2, \ldots, A_n\}$ for n attention heads
Parameter: α, β
Output: Calibrated attention maps $\hat{A}$

for all attention heads $A_i \in A$ **do**
  Mask partial attention heads using Eq. (4)
  Form coalition $\hat{H}$ using Eq. (5)
  Compute Harsanyi dividend for $\hat{H}$ using Eq. (5)
end for
Identify salient group $G(h)$ using Eq. (7)
for each attention head $A_i \in A$ **do**
  if $A_i \notin G(h)$ **then**
    for each token $m \in \{1, \ldots, N\}$ **do**
      Compute average attention score using Eq. (9)
      Identify focused tokens S_i using Eq. (10)
    end for
    for focused tokens $s \in S_i$ **do**
      Diminish attention score using Eq. (11)
    end for
    Distribute additional score using Eq. (12)
  end if
end for

With this method, we allocate the additional attention score proportionally based on the weights of the tokens. We also investigate the uniform distribution in Table 8 of the Appendix.

After describing our proposed training-free Game-theoretic Attention Calibration (GAC), we summarize the process into below Algorithm 1.

5 Experiment

5.1 Experimental Settings

Models. We evaluate our method on four LLMs, including Llama3.1-8B-chat (Dubey et al., 2024), Qwen2.5-7B-Instruct, Qwen2.5-14B-Instruct, and Qwen2.5-32B-Instruct (Team, 2024).

Datasets: We evaluate three types of LLM tasks with 16 different benchmark datasets: ❶ **Text Classification**: SST2 (Socher et al., 2013), SST5 (Socher et al., 2013), MR (Pang and Lee, 2005), AGNews (Zhang et al., 2015), TREC (Voorhees and Tice, 2000), CB (De Marneffe et al., 2019), and BoolQ (Clark et al., 2019a). ❷ **Multiple choice**: Hellaswag (Zellers et al., 2019), ARCE (Clark et al., 2018), PIQA (Bisk et al., 2020), OB (Mihaylov et al., 2018), ARCC (Clark et al., 2018), COPA (Wang et al., 2019), CQA (Talmor et al., 2018). ❸ **Open-ended question answering**: SQuAD v1 (Rajpurkar et al., 2016) and SQuAD v2 (Rajpurkar et al., 2018).

In addition, we construct an evaluation set to compute the Harsanyi dividend for each combination and describe it in Appendix D.

Model	Method	SST2	SST5	MR	SUBJ	AGNews	TREC	CB	BoolQ	Average
Llama-3.1-8B-Instruct	Vanilla	92.09	48.32	91.74	68.70	73.16	11.60	64.29	81.99	66.49
	ACT	93.12	49.23	92.40	71.55	74.13	14.40	67.86	83.67	68.30
	Sahara	92.89	49.41	92.57	72.60	73.76	13.00	67.86	83.21	68.16
	GAC	94.15	51.86	92.78	76.85	88.34	43.80	82.14	86.27	**77.02**
Qwen2.5-7B-Instruct	Vanilla	92.89	49.14	90.43	49.60	81.68	18.20	83.93	85.41	68.91
	ACT	92.55	49.86	90.99	66.90	83.34	19.00	89.29	85.90	77.23
	Sahara	94.50	49.68	90.90	53.50	82.28	22.40	91.07	87.36	71.46
	GAC	94.84	54.22	93.71	81.15	87.47	31.80	94.64	88.65	**78.31**
Qwen2.5-14B-Instruct	Vanilla	92.55	49.23	91.28	65.60	85.86	25.4	83.93	83.98	72.23
	ACT	93.46	50.14	92.03	68.40	86.13	27.60	87.50	85.23	73.81
	Sahara	93.23	50.23	92.03	69.45	86.18	27.00	87.50	85.09	73.84
	GAC	95.76	51.32	93.15	80.55	86.32	49.60	98.21	86.09	**80.13**
Qwen2.5-32B-Instruct	Vanilla	94.38	51.59	91.46	81.2	85.82	22.4	83.93	87.86	74.83
	ACT	94.72	52.04	91.84	82.15	85.99	26.40	85.71	88.32	75.90
	Sahara	94.72	52.40	91.74	84.10	86.00	26.60	85.71	88.17	76.18
	GAC	97.13	54.95	93.62	95.60	88.96	30.40	92.86	88.71	**80.28**

Table 1: Accuracy comparison of our GAC and recent methods on text classification datasets across four LLMs.

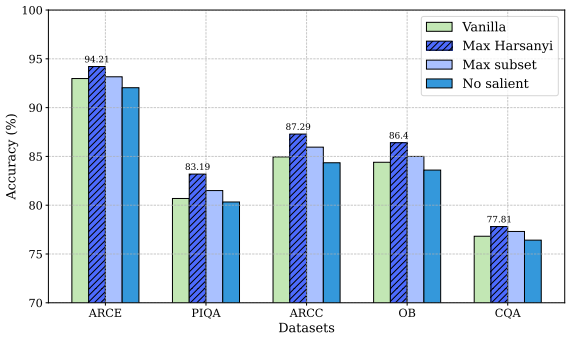

Figure 3: Ablation of identifying salient groups.

Baselines: We compare our GAC with vanilla inference and two recent methods ACT (Yu et al., 2024) and Sahara (Zhou et al., 2024). ACT individually considers each head and modifies the attention map for each single attention head. Sahara is first proposed to identify the most important head for model safety. We adapt it to our paper by identifying the most substantial performance improvements when omitting the specific heads.

Implementation details In all our experiments, we adopt the maximum Harsanyi to identify salient groups and set α to 0.1 and β to 0.1 in our GAC method. Considering the number of coalitions (2^n) is large when there are numerous heads, similar to game theory in explaining DNN (Liu et al., 2023a; Shen et al., 2021), we group the attention heads corresponding to the same key/value head as a player and then analyze the interactions between these players in our paper. In this way, there are eight players in each layer for our evaluated models.

5.2 Main Results

Text classification. As shown in Table 1, we validate GAC on diverse text classification datasets. The results demonstrate that GAC consistently improves accuracy compared to the vanilla inference baseline. Concretely, GAC delivers average accuracy gains of 10.53%, 9.40%, 8.08%, and 5.45% for Llama3.1-8B, Qwen2.5-7B, Qwen2.5-14B, and Qwen2.5-32B, respectively. Notably, on challenging benchmarks, such as TREC and CB, GAC obtains significant improvement. Furthermore, our GAC also significantly surpasses ACT and Sahara. These experiments further validate the robustness of our GAC in improving the model capability.

Domain-specific multiple choice. Moreover, we evaluate GAC on domain-specific multiple-choice datasets in Table 2. On average, GAC yields accuracy gains ranging from 1.67% to 2.93% across various models. Notably, we observe a peak accuracy improvement of 5.2% on a single dataset OB. Moreover, our GAC also surpasses the previous method ACT and Sahara. Especially on Qwen2.5-32B-Instruct, our GAC obtains significantly better performance than them. These experiments validate the generalization of our GAC across different model families and sizes.

Open-ended question-answering. To further assess the generalization of our GAC, we evaluate it on open-ended question-answering with the established SQuAD v1 and SQuAD v2 datasets. In this context, we employ the F1 score as the characteristic function. For more complex open-ended tasks, GPT evaluation can be adopted as the characteristic

Model	Method	Hellaswag	ARCE	PIQA	ARCC	OB	CQA	Average
Llama-3.1-8B-Instruct	Vanilla	70.85	92.98	80.69	84.95	84.40	76.82	81.78
	ACT	71.51	93.51	82.64	85.95	86.00	77.23	82.81
	Sahara	71.72	93.33	82.32	86.29	85.40	77.07	82.69
	GAC	72.29	94.21	83.19	87.29	86.40	77.81	**83.53**
Qwen2.5-7B-Instruct	Vanilla	85.56	95.79	84.93	88.63	82.20	83.13	86.71
	ACT	86.08	96.67	85.80	90.64	84.40	83.62	87.87
	Sahara	86.02	96.67	85.85	89.97	84.20	83.37	87.68
	GAC	85.95	96.84	86.18	91.30	85.80	84.19	**88.38**
Qwen2.5-14B-Instruct	Vanilla	89.75	97.02	85.13	92.98	84.60	84.03	88.92
	ACT	89.84	97.72	89.21	93.31	86.00	84.44	90.87
	Sahara	89.87	97.89	91.28	93.65	86.20	84.52	90.57
	GAC	91.02	99.13	87.27	94.31	88.32	85.67	**90.95**
Qwen2.5-32B-Instruct	Vanilla	88.14	96.84	85.53	92.64	83.00	84.52	88.45
	ACT	88.35	97.19	87.15	92.97	85.40	86.90	89.66
	Sahara	89.67	97.54	86.61	92.97	85.40	86.57	89.79
	GAC	90.56	98.59	89.53	95.41	88.20	85.99	**91.38**

Table 2: Accuracy comparison of our GAC and recent methods on multiple choice datasets across four LLMs.

Model	Method	SQuAD v1	SQuAD v2
Llama-3.1-8B-Instruct	Vanilla	84.98/72.00	36.11/24.80
	GAC	88.16/79.64	40.16/32.47
	Improv.	3.18/7.64	4.05/7.67

Table 3: Our proposed GAC on open-ended question answering datasets. Each result for SQuADv1/v2 is presented as the exact match score/F1 score.

α	Hellaswag	ARCE	PIQA	ARCC	OB	CQA
Vanilla	70.85	92.98	80.69	84.95	84.40	76.82
0.05	72.13	94.21	82.64	86.96	86.40	77.48
0.10	72.29	94.21	83.19	87.29	86.40	77.81
0.15	72.16	93.86	82.97	86.62	85.80	77.64
0.20	72.04	93.51	81.66	86.29	86.40	77.64

Table 4: Ablation study on α selection.

function. We leave it for future study. Considering these datasets are significantly large, we only conduct experiments on the LLaMA3.1-8B model. As shown in Table 3, GAC consistently outperforms vanilla inference across all metrics on SQuAD v1 and SQuAD v2. Specifically, GAC demonstrates a 3.18% increase in exact match score, and a 7.64% gain in F1 score over the baseline in SQuAD v1. Similarly, our proposed GAC achieves obvious improvement on SQuAD v2. These results further demonstrate the generalization of our GAC.

6 Ablation Study

6.1 Ways to Identify Salient Group

In this section, we compare two methods in Section 4.1 to identify the salient group. Our experiments are performed on challenging multi-choice datasets.

β	Hellaswag	ARCE	PIQA	ARCC	OB	CQA
Vanilla	70.85	92.98	80.69	84.95	84.40	76.82
0.7	71.64	93.68	82.54	86.96	85.40	77.40
0.5	71.45	93.86	83.03	87.29	86.20	77.48
0.3	71.52	93.86	82.64	86.96	85.80	77.64
0.1	72.29	94.21	83.19	87.29	86.40	77.81

Table 5: Ablation study on β selection.

As shown in Figure 3, using the simple maximum Harsanyi dividend achieves better results than identifying the maximum subset. Notably, without identifying the salient group and performing attention distribution calibration for all heads leads to poorer performance than the vanilla as the distribution of all heads is forced to be smooth. Therefore, we adopt the maximum Harsanyi due to its simplicity and better performance.

6.2 Ablation of Attention Calibration

Here we analyze the hyper-parameters α and β in Section 4.2 for attention calibration. Specifically, α defines the threshold for identifying tokens with excessive attention weight. To assess the robustness of our GAC to varying α values, we conduct experiments using Llama3.1-8B-chat across a range of α values. As shown in Table 4, GAC consistently surpasses the vanilla inference, and the selection of α has a minimal impact on the performance, meaning excessive attention has a significantly larger weight. In this study, we empirically set α to 0.1.

Moreover, β governs the decrease degree. Similar to the above experiment, we conduct experiments with various β values. As shown in Table

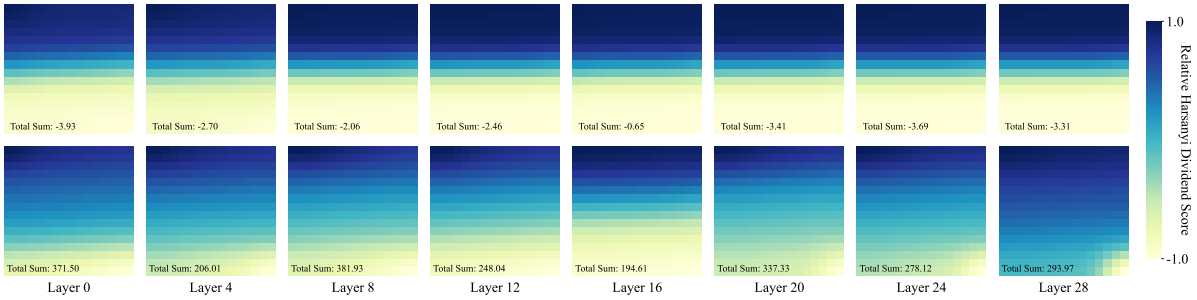

Figure 4: The distribution of the Harsanyi dividend on each layer before (first row) and after (second row) applying GAC. Each cell represents a relative Harsanyi dividend score for better reference. Higher values indicate stronger cooperation. In addition, the total sum of the Harsanyi dividend for all coalitions obtains a significant increase.

Dataset	Model	Method	Accuracy	Precision	Recall	F1
MSCOCO	Random	Vanilla	90.58	96.94	83.60	89.78
		GAC	**92.32**	96.16	88.16	**91.99**
	Popular	Vanilla	89.30	93.64	83.60	88.33
		GAC	**90.00**	93.85	85.60	**89.54**
	Aversarial	Vanilla	86.56	88.86	83.60	86.15
		GAC	**87.64**	91.31	83.20	**87.07**
A-OKVQA	Random	Vanilla	92.24	92.30	92.15	92.23
		GAC	**93.36**	92.13	94.36	**93.67**
	Popular	Vanilla	89.00	86.67	92.15	89.33
		GAC	**90.00**	88.01	92.63	**90.16**
	Adversarial	Vanilla	80.56	74.84	92.16	82.61
		GAC	**83.04**	78.88	90.24	**84.18**
GQA	Random	Vanilla	91.72	91.02	92.55	91.78
		GAC	**92.94**	92.12	92.48	**92.47**
	Popular	Vanilla	84.76	80.00	92.55	85.82
		GAC	**86.64**	84.32	89.99	**87.06**
	Adversarial	Vanilla	81.56	75.82	92.56	83.36
		GAC	**84.08**	80.69	89.60	**84.91**

Table 6: The performance of vanilla and our GAC on the MLLM benchmark POPE. The best performances of accuracy and F1 score within each setting are **bolded**.

5, a strong smooth coefficient 0.1 leads to better performance. Therefore, we set β to 0.1.

6.3 Generalization Study on MLLM Model

In this section, we demonstrate the generalization of our GAC method on the Multimodal Large Language Model (MLLM) llava-v1.6-mistral-7b (Liu et al., 2023b). Specifically, we conduct experiments on POPE benchmark (Li et al., 2023) for a comprehensive study that consists of three subsets. The results in Table 6 show that our proposed GAC can produce consistency gains than the vanilla inference across different splits of the POPE benchmark. We further compare our GAC with two recent training-free methods VCD (Leng et al., 2024) and CODE (Kim et al., 2024) in Table 9 of the Appendix. However, these two powerful methods do not improve upon this robust MLLM, further demonstrating the effectiveness of our GAC.

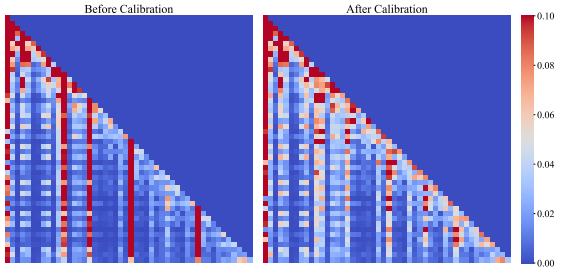

Figure 5: Visualization of the attention distribution before and after calibration. After calibration, the columns with excessive attention are smoothed.

7 Visualization of Harsanyi Distribution and Attention Distribution

In this section, we provide two visualizations to help understand our study. Firstly, we demonstrate the Harsany dividend before and after applying GAC. Figure 4 shows that the Harsanyi dividend for each layer generally increases, as indicated by the progressively darkening color. Furthermore, we compute the total Harsanyi dividends for all coalitions and observe a significant increase in the total sum after applying GAC, namely most coalitions present synergistic effects. More visualizations are depicted in Figure 8 and Figure 9. Notably, the substantial reduction in negative Harsanyi dividends verifies the efficacy of our calibration method.

Moreover, to understand the distribution calibration in Section 4.2, we visualize the Llama3.1-8B-Instruct attention maps before and after calibration. As depicted in Figure 5, the original excessive attention weight is substantially diminished, while the attention distribution across the first token and other tokens remains largely unchanged.

8 Conclusion

In this paper, we present a novel framework, grounded in the Harsanyi dividend from game theory, to dissect the intricate interplay among atten-

tion heads. Our analysis reveals that significant Harsanyi dividends are sparse and the existence of negative Harsanyi dividends. To further optimize the interactions among attention heads, we propose a training-free Game-theoretic Attention Calibration (GAC) method. Comprehensive experiments across 17 benchmarks demonstrate that GAC effectively promotes inter-head collaboration and improves model capabilities across diverse model families, scales, and modalities. Moreover, the discovered interaction offers a path toward a deeper understanding of the behaviors of LLMs.

Limitation

The limitations of this paper include:

- In this paper, we make the first attempt to uncover the interactions among attention heads. Limited by computational resources, we do not conduct experiments on highly complex datasets, such as the math dataset GSM8K (Cobbe et al., 2021), as they typically require generating lengthy responses.
- In this paper, we focus on unveiling the potential of game theory in analyzing the attention and leaving the efficiency improvement for future study. Our GAC needs to compute the Harsanyi dividend for each coalition, therefore the computation cost is larger than in previous works. However, these costs can be reduced by sampling methods, such as importance sampling or Monte Carlo sampling.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No.62276110, No.62172039, and in part by the fund of the Joint Laboratory of HUST and Pingan Property & Casualty Research (HPL). The authors would also like to thank reviewers for their comments on improving the quality of this paper.

References

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

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Maximiliana Behnke and Kenneth Heafield. 2020. Losing heads in the lottery: Pruning transformer. In *The 2020 Conference on Empirical Methods in Natural Language Processing*, pages 2664–2674. Association for Computational Linguistics (ACL).

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Georgios Chalkiadakis, Edith Elkind, and Michael Wooldridge. 2022. *Computational aspects of cooperative game theory*. Springer Nature.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019a. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019b. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

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

Marie-Catherine De Marneffe, Mandy Simons, and Judith Tonhauser. 2019. The commitmentbank: Investigating projection in naturally occurring discourse. In *proceedings of Sinn und Bedeutung*, volume 23, pages 107–124.

Jianfeng Dong, Xianke Chen, Minsong Zhang, Xun Yang, Shujie Chen, Xirong Li, and Xun Wang. 2022. Partially relevant video retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 246–257.

Jianfeng Dong, Xiaoman Peng, Daizong Liu, Xiaoye Qu, Xun Yang, Cuizhu Bao, and Meng Wang. 2024. Temporal sentence grounding with relevance feedback in videos. *Advances in Neural Information Processing Systems*, 37:43107–43132.

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

Xiang Fang, Zeyu Xiong, Wanlong Fang, Xiaoye Qu, Chen Chen, Jianfeng Dong, Keke Tang, Pan Zhou, Yu Cheng, and Daizong Liu. 2024. Rethinking weakly-supervised video temporal grounding from a game perspective. In *European Conference on Computer Vision*, pages 290–311. Springer.

Dan Guo, Kun Li, Bin Hu, Yan Zhang, and Meng Wang. 2024. Benchmarking micro-action recognition: Dataset, method, and application. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(7):6238–6252.

John C Harsanyi. 1982. A simplified bargaining model for the n-person cooperative game. *Papers in game theory*, pages 44–70.

Peng Jin, Jinfa Huang, Pengfei Xiong, Shangxuan Tian, Chang Liu, Xiangyang Ji, Li Yuan, and Jie Chen. 2023. Video-text as game players: Hierarchical banzhaf interaction for cross-modal representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2472–2482.

Zhuoran Jin, Pengfei Cao, Hongbang Yuan, Yubo Chen, Jiexin Xu, Huaijun Li, Xiaojian Jiang, Kang Liu, and Jun Zhao. 2024. Cutting off the head ends the conflict: A mechanism for interpreting and mitigating knowledge conflicts in language models. *arXiv preprint arXiv:2402.18154*.

Junho Kim, Hyunjun Kim, Yeonju Kim, and Yong Man Ro. 2024. Code: Contrasting self-generated description to combat hallucination in large multi-modal models. *arXiv preprint arXiv:2406.01920*.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. 2024. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13872–13882.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*.

Daizong Liu, Yang Liu, Wencan Huang, and Wei Hu. 2024a. A survey on text-guided 3d visual grounding: Elements, recent advances, and future directions. *arXiv preprint arXiv:2406.05785*.

Daizong Liu, Mingyu Yang, Xiaoye Qu, Pan Zhou, Yu Cheng, and Wei Hu. 2024b. A survey of attacks on large vision-language models: Resources, advances, and future trends. *arXiv preprint arXiv:2407.07403*.

Daizong Liu, Mingyu Yang, Xiaoye Qu, Pan Zhou, Xiang Fang, Keke Tang, Yao Wan, and Lichao Sun. 2024c. Pandora's box: Towards building universal attackers against real-world large vision-language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Daizong Liu, Pan Zhou, Zichuan Xu, Haozhao Wang, and Ruixuan Li. 2022. Few-shot temporal sentence grounding via memory-guided semantic learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(5):2491–2505.

Dongrui Liu, Huiqi Deng, Xu Cheng, Qihan Ren, Kangrui Wang, and Quanshi Zhang. 2023a. Towards the difficulty for a deep neural network to learn concepts of different complexities. *Advances in Neural Information Processing Systems*, 36:41283–41304.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. In *NeurIPS*.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2023. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? *Advances in neural information processing systems*, 32.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *arXiv preprint cs/0506075*.

Xiaoye Qu, Qiyuan Chen, Wei Wei, Jishuo Sun, and Jianfeng Dong. 2024a. Alleviating hallucination in large vision-language models with active retrieval augmentation. *arXiv preprint arXiv:2408.00555*.

Xiaoye Qu, Daize Dong, Xuyang Hu, Tong Zhu, Weigao Sun, and Yu Cheng. 2024b. Llama-moe v2: Exploring sparsity of llama from perspective of mixture-of-experts with post-training. *arXiv preprint arXiv:2411.15708*.

Xiaoye Qu, Yafu Li, Zhaochen Su, Weigao Sun, Jianhao Yan, Dongrui Liu, Ganqu Cui, Daizong Liu, Shuxian Liang, Junxian He, et al. 2025. A survey of efficient reasoning for large reasoning models: Language, multimodality, and beyond. *arXiv preprint arXiv:2503.21614*.

Xiaoye Qu, Jiashuo Sun, Wei Wei, and Yu Cheng. 2024c. Look, compare, decide: Alleviating hallucination in large vision-language models via multi-view multi-path reasoning. *arXiv preprint arXiv:2408.17150*.

Xiaoye Qu, Pengwei Tang, Zhikang Zou, Yu Cheng, Jianfeng Dong, Pan Zhou, and Zichuan Xu. 2020. Fine-grained iterative attention network for temporal language localization in videos. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4280–4288.

Xiaoye Qu, Jun Zeng, Daizong Liu, Zhefeng Wang, Baoxing Huai, and Pan Zhou. 2023. Distantly-supervised named entity recognition with adaptive teacher learning and fine-grained student ensemble. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13501–13509.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Jie Ren, Mingjie Li, Qirui Chen, Huiqi Deng, and Quanshi Zhang. 2023. Defining and quantifying the emergence of sparse concepts in dnns. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20280–20289.

Qihan Ren, Yang Xu, Junpeng Zhang, Yue Xin, Dongrui Liu, and Quanshi Zhang. 2024. Towards the dynamics of a dnn learning symbolic interactions. *arXiv preprint arXiv:2407.19198*.

Wen Shen, Qihan Ren, Dongrui Liu, and Quanshi Zhang. 2021. Interpreting representation quality of dnns for 3d point cloud processing. *Advances in Neural Information Processing Systems*, 34:8857–8870.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Zhaochen Su, Jun Zhang, Xiaoye Qu, Tong Zhu, Yanshu Li, Jiashuo Sun, Juntao Li, Min Zhang, and Yu Cheng. 2024. Conflictbank: A benchmark for evaluating the influence of knowledge conflicts in llm. *arXiv preprint arXiv:2408.12076*.

Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. 2020. The shapley taylor interaction index. In *International conference on machine learning*, pages 9259–9268. PMLR.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Yunbo Tao, Daizong Liu, Pan Zhou, Yulai Xie, Wei Du, and Wei Hu. 2023. 3dhacker: Spectrum-based decision boundary generation for hard-label 3d point cloud attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14340–14350.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. *arXiv preprint arXiv:1905.09418*.

Ellen M Voorhees and Dawn M Tice. 2000. Building a question answering test collection. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 200–207.

Zhongwei Wan, Ziang Wu, Che Liu, Jinfa Huang, Zhihong Zhu, Peng Jin, Longyue Wang, and Li Yuan. 2024. Look-m: Look-once optimization in kv cache for efficient multimodal long-context inference. *arXiv preprint arXiv:2406.18139*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Hanrui Wang, Zhekai Zhang, and Song Han. 2021. Spatten: Efficient sparse attention architecture with cascade token and head pruning. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, pages 97–110. IEEE.

Jin Wang, Shichao Dong, Yapeng Zhu, Kelu Yao, Weidong Zhao, Chao Li, and Ping Luo. 2024. Diagnosing the compositional knowledge of vision language models from a game-theoretic view. *arXiv preprint arXiv:2405.17201*.

Haoyi Wu and Kewei Tu. 2024. Layer-condensed kv cache for efficient inference of large language models. *arXiv preprint arXiv:2405.10637*.

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

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 *International Conference on Machine Learning*.

Abdelrahman Zayed, Gonçalo Mordido, Samira Shabanian, Ioana Baldini, and Sarath Chandar. 2024. Fairness-aware structured pruning in transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22484–22492.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Zhenyu Zhang, Shiwei Liu, Runjin Chen, Bhavya Kailkhura, Beidi Chen, and Atlas Wang. 2024. Q-hitter: A better token oracle for efficient llm inference via sparse-quantized kv cache. *Proceedings of Machine Learning and Systems*, 6:381–394.

Zhenhong Zhou, Haiyang Yu, Xinghua Zhang, Rongwu Xu, Fei Huang, Kun Wang, Yang Liu, and Yongbin Fang, Junfeng andxiao2023efficient Li. 2024. On the role of attention heads in large language model safety. *arXiv preprint arXiv:2410.13708*.

Jiahao Zhu, Daizong Liu, Pan Zhou, Xing Di, Yu Cheng, Song Yang, Wenzheng Xu, Zichuan Xu, Yao Wan, Lichao Sun, et al. 2023. Rethinking the video sampling and reasoning strategies for temporal sentence grounding. *arXiv preprint arXiv:2301.00514*.

A Axioms of the Harsanyi dividend

In this paper, we adopt the Harsanyi dividend (Harsanyi, 1982) to quantify the interactions for each attention head coalition as it satisfies many axioms, which provide solid theoretical foundations for our explanations. Specifically, the Harsanyi dividend $w_{\mathcal{S}}$ in Eq. 2 satisfies seven desirable axioms, including the *efficiency, linearity, dummy, symmetry, anonymity, recursive* and *interaction distribution* axioms:

(1) *Efficiency axiom*. The output score of a model can be decomposed into effects of different causal patterns, *i.e.* $v(\boldsymbol{x}) = \sum_{\mathcal{S}\subseteq\mathcal{N}} w_{\mathcal{S}}$.

(2) *Linearity axiom*. If we merge the output scores of the two models $t(\cdot)$ and $u(\cdot)$ into the output of model $v(\cdot)$, *i.e.* $\forall \mathcal{S} \subseteq \mathcal{N}$, $v(\boldsymbol{x}_{\mathcal{S}}) = t(\boldsymbol{x}_{\mathcal{S}}) + u(\boldsymbol{x}_{\mathcal{S}})$, the corresponding causal effects $w^t_{\mathcal{S}}$ and $w^u_{\mathcal{S}}$ can also be merged as $\forall \mathcal{S} \subseteq \mathcal{N}, w^v_{\mathcal{S}} = w^t_{\mathcal{S}} + w^u_{\mathcal{S}}$.

(3) *Dummy axiom*. If a variable $i \in \mathcal{N}$ is a dummy variable, *i.e.* $\forall \mathcal{S} \subseteq \mathcal{N}\backslash\{i\}, v(\boldsymbol{x}_{\mathcal{S}\cup\{i\}}) = v(\boldsymbol{x}_{\mathcal{S}}) + v(\boldsymbol{x}_{\{i\}})$, it has no causal effect with other variables, $\forall \mathcal{S} \subseteq \mathcal{N}\backslash\{i\}, w_{\mathcal{S}\cup\{i\}} = 0$.

(4) *Symmetry axiom*. If the input variables $i, j \in \mathcal{N}$ cooperate with other variables in the same manner, $\forall \mathcal{S} \subseteq \mathcal{N}\backslash\{i, j\}, v(\boldsymbol{x}_{\mathcal{S}\cup\{i\}}) = v(\boldsymbol{x}_{\mathcal{S}\cup\{j\}})$, then they have the same causal effects with other variables, $\forall S \subseteq \mathcal{N}\backslash\{i, j\}, w_{\mathcal{S}\cup\{i\}} = w_{\mathcal{S}\cup\{j\}}$.

(5) *Anonymity axiom*. For any permutations π on $\mathcal{N}$, we have $\forall \mathcal{S} \subseteq \mathcal{N}, w^v_{\mathcal{S}} = w^{\pi v}_{\pi\mathcal{S}}$, where $\pi\mathcal{S} \triangleq \{\pi(i)|i \in \mathcal{S}\}$, and the new model πv is defined by $(\pi v)(\boldsymbol{x}_{\pi\mathcal{S}}) = v(\boldsymbol{x}_{\mathcal{S}})$. This indicates that causal effects are not changed by the permutation.

(6) *Recursive axiom*. The causal effects can be computed recursively. For $i \in \mathcal{N}$ and $\mathcal{S} \subseteq \mathcal{N}\backslash\{i\}$, the causal effect of the pattern $\mathcal{S} \cup \{i\}$ is equal to the causal effect of $\mathcal{S}$ in the presence of i minus the causal effect of $\mathcal{S}$ in the absence of i, *i.e.* $\forall \mathcal{S} \subseteq \mathcal{N} \setminus \{i\}, w_{\mathcal{S}\cup\{i\}} = w_{\mathcal{S}|i \text{ present}} - w_{\mathcal{S}}$. $w_{\mathcal{S}|i \text{ present}}$ denotes the causal effect when the variable i is always present as a constant context, *i.e.* $w_{\mathcal{S}|i \text{ present}} = \sum_{\mathcal{S}'\subseteq\mathcal{S}} (-1)^{|\mathcal{S}|-|\mathcal{S}'|} \cdot v(\boldsymbol{x}_{\mathcal{S}'\cup\{i\}})$.

(7) *Interaction distribution axiom*. This axiom characterizes how causal effects are distributed for a class of "interaction functions" (Sundararajan et al., 2020). The interaction function $v_{\mathcal{T}}$ parameterized by a subset of variables $\mathcal{T}$ is defined as follows. $\forall \mathcal{S} \subseteq \mathcal{N}$, if $\mathcal{T} \subseteq \mathcal{S}$, $v_{\mathcal{T}}(\boldsymbol{x}_{\mathcal{S}}) = c$; otherwise, $v_{\mathcal{T}}(\boldsymbol{x}_{\mathcal{S}}) = 0$. The function $v_{\mathcal{T}}$ models the causal effect of the pattern $\mathcal{T}$, because only if all variables in $\mathcal{T}$ are present, will the output value be increased by c. The causal effects encoded in the function $v_{\mathcal{T}}$ satisfy $w_{\mathcal{T}} = c$, and $\forall \mathcal{S} \neq \mathcal{T}$, $w_{\mathcal{S}} = 0$.

B Undifferentiated Attention

In Section 3.3, we use undifferentiated attention (Zhou et al., 2024) to mask specific attention heads. In this section, we provide a detailed derivation of this attention. Let denote the unscaled attention weights as z, *i.e.*:

$$z = \frac{W^i_q W^{iT}_k}{\sqrt{d_k/n}} \tag{13}$$

Therefore, the softmax function for the input vector z_i scaled by the small coefficient ϵ can be rewritten as:

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{14}$$

For the scaled input ϵz_i, when ϵ is very small, the term ϵz_i approaches zero. Using the first-order approximation of the exponential function around zero: $e^{\epsilon z_i} \approx 1 + \epsilon z_i$, we get:

$$\text{Softmax}(\epsilon z_i) \approx \frac{1 + \epsilon z_i}{\Sigma_j(1 + \epsilon z_i)} = \frac{1 + \epsilon z_i}{N + \epsilon\Sigma_j z_j} \tag{15}$$

where N is the number of elements in z. As ϵ approaches zero, the numerator and denominator respectively converge to 1 and N. Thus, the output simplifies to:

Dataset	Undifferentiated Attention	Dropping Head
HellaSwag	72.29	72.04
ARCE	94.21	93.86
PIQA	83.19	83.19
ARCC	87.29	87.29
OB	86.40	86.40
CQA	77.81	77.64

Table 7: Different methods for masking attention heads across multiple-choice tasks.

Method	Vanilla	Uniform Distribution	Ours
ARCE	92.98	93.68	94.21
PIQA	80.69	82.37	83.19
ARCC	84.95	87.29	87.29
OB	84.40	86.00	86.40
CQA	76.82	77.31	77.81

Table 8: Ablation of distributing the additional attention.

$$\text{Softmax}(\epsilon z_i) \approx \frac{1}{N} \tag{16}$$

Finally, the output h_i of the attention head degenerates to $\frac{1}{N} W_v^i$, which holds exactly when $\epsilon = 0$.

C Methods for masking heads

In our proposed GAC method, we use undifferentiated attention to mask attention heads. In this section, we compare undifferentiated attention with a simple dropping strategy (Michel et al., 2019), namely $\hat{h}_i = 0$ instead of using Eq. 4.

In theory, undifferentiated attention can achieve a finer degree of control over the influence that a particular attention head exerts on the output. To further demonstrate the effectiveness of undifferentiated attention, we compare these two methods in Table 7 and we can observe that undifferentiated attention achieves slightly better performance than the simple dropping strategy.

D Validation Set Description

In our proposed GAC method, we first compute the Harsanyi dividend for each coalition and then identify the salient group. In this way, the attention heads in the salient group do not undergo any attention calibration during inference. To compute the Harsanyi dividend, we need a validation set. Specifically, following (Yu et al., 2024), for each task $\mathcal{T} = \{D_1, \ldots, D_Q\}$, consisting of Q different datasets, we initially create a small held-out dataset C by uniformly sampling M data samples from each dataset, i.e., each dataset D_q in $\mathcal{T}$ has M samples in C. In this paper, we sample 300 samples (i.e. $M = 300$) from each dataset to form the validation set. In our preliminary experiments, we found that more validation data will not lead to significant performance improvement. It means that the current data volume of the validation set is sufficient to identify the salient group. Notably, our validation set is constructed with significantly fewer samples than the overall dataset, comprising less than 10% of the total data.

E Methods for distributing attention

After smoothing the excessive attention in Section 4.2, the following question is how to redistribute this reduced attention across other tokens. In this section, we evaluate another redistribution strategy, where the additional attention is uniformly distributed across all tokens. As shown in Table 8, the results demonstrate that allocating the additional attention score proportionally based on the weights of tokens leads to better performance.

Dataset	Model	Method	Accuracy	Precision	Recall	F1
MSCOCO	Random	Vanilla	90.58	96.94	83.60	89.78
		VCD	87.32	97.07	76.96	85.85
		CODE	89.42	97.14	82.32	89.12
		GAC	**92.32**	96.16	88.16	**91.99**
	Popular	Vanilla	89.30	93.64	83.60	88.34
		VCD	85.68	93.22	76.96	84.31
		CODE	88.56	93.62	82.32	87.61
		GAC	**90.00**	93.85	85.60	**89.54**
	Aversarial	Vanilla	86.56	88.86	83.60	86.15
		VCD	82.24	86.11	76.88	81.23
		CODE	86.60	89.35	84.24	86.72
		GAC	**87.64**	91.31	83.20	**87.07**
A-OKVQA	Random	Vanilla	92.24	92.30	92.15	92.22
		VCD	89.12	91.65	86.07	88.77
		CODE	91.76	91.94	91.95	91.95
		GAC	**93.36**	92.13	94.36	**93.23**
	Popular	Vanilla	89.00	86.67	92.15	89.33
		VCD	86.64	87.04	86.07	86.55
		CODE	88.64	87.90	91.95	89.88
		GAC	**90.00**	88.01	92.63	**90.26**
	Adversarial	Vanilla	80.56	74.84	92.16	82.60
		VCD	78.20	74.81	85.04	79.60
		CODE	80.40	75.36	88.40	81.36
		GAC	**83.04**	78.88	90.24	**84.18**
GQA	Random	Vanilla	91.72	91.02	92.55	91.78
		VCD	88.36	90.80	85.35	87.99
		CODE	91.20	90.97	91.97	91.47
		GAC	**92.94**	92.12	92.48	**92.30**
	Popular	Vanilla	84.76	80.00	92.55	85.82
		VCD	82.64	80.94	85.35	83.09
		CODE	84.00	82.39	89.91	85.99
		GAC	**86.64**	84.32	89.99	**87.06**
	Adversarial	Vanilla	81.56	75.82	92.56	83.36
		VCD	79.00	75.87	85.04	80.20
		CODE	80.96	76.50	90.60	82.96
		GAC	**84.08**	80.69	89.60	**84.91**

Table 9: The performance of our proposed GAC on the MLLM benchmark POPE. We compare our method with two recent training-free methods VCD and CODE. The best performances of accuracy and F1 score within each setting are **bolded**. When applied to a strong base model llava-v1.6-mistral-7b, VCD and CODE fail to yield performance improvements. Conversely, our proposed GAC method demonstrates a substantial performance boost.

Figure 6: Harsanyi dividend distribution of each layer, using classification datasets.

Figure 7: Harsanyi dividend distribution of each layer, using multiple choice datasets.

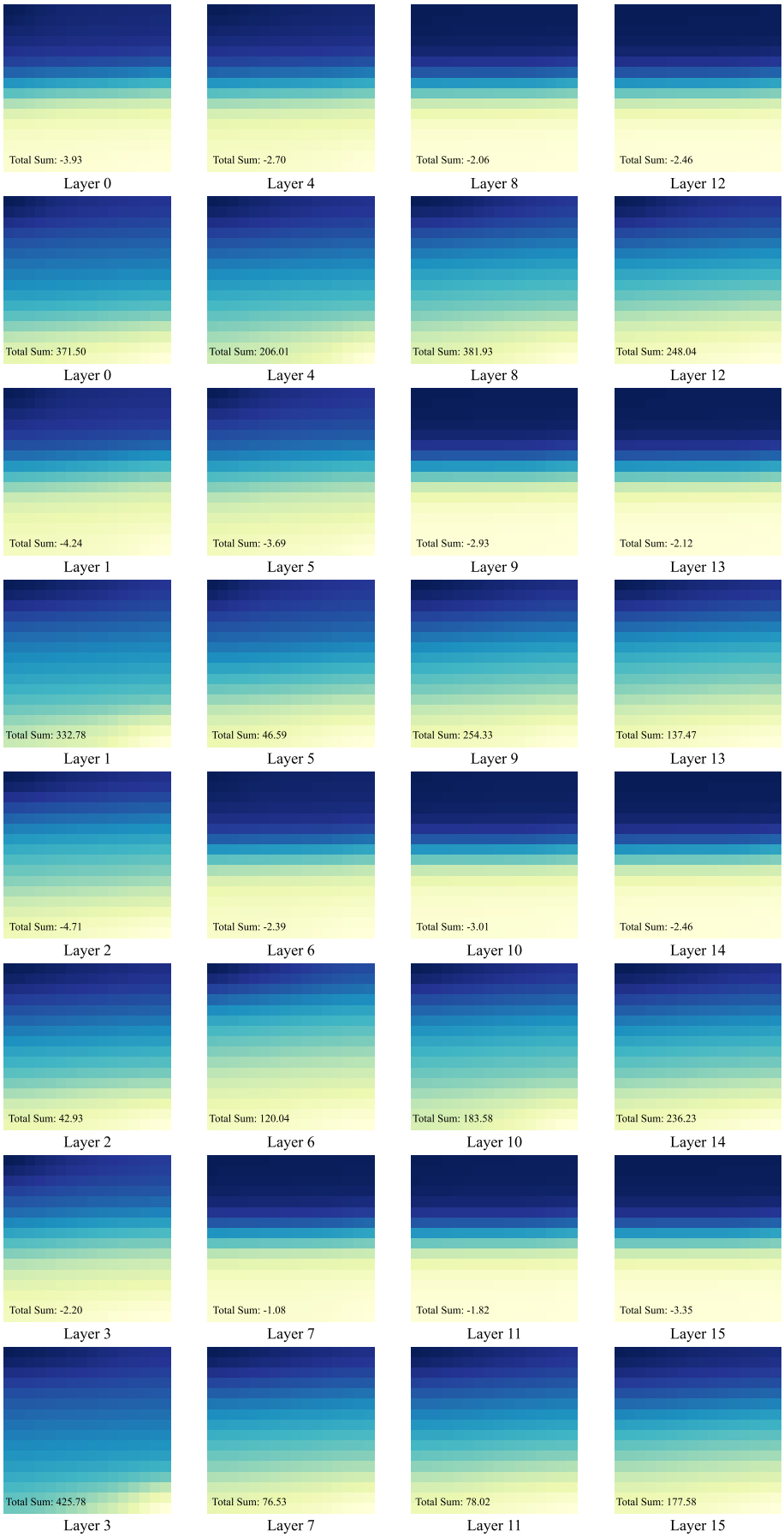

Figure 8: The distribution of the Harsanyi dividend on each layer before (odd-numbered rows) and after (even-numbered rows) applying GAC.

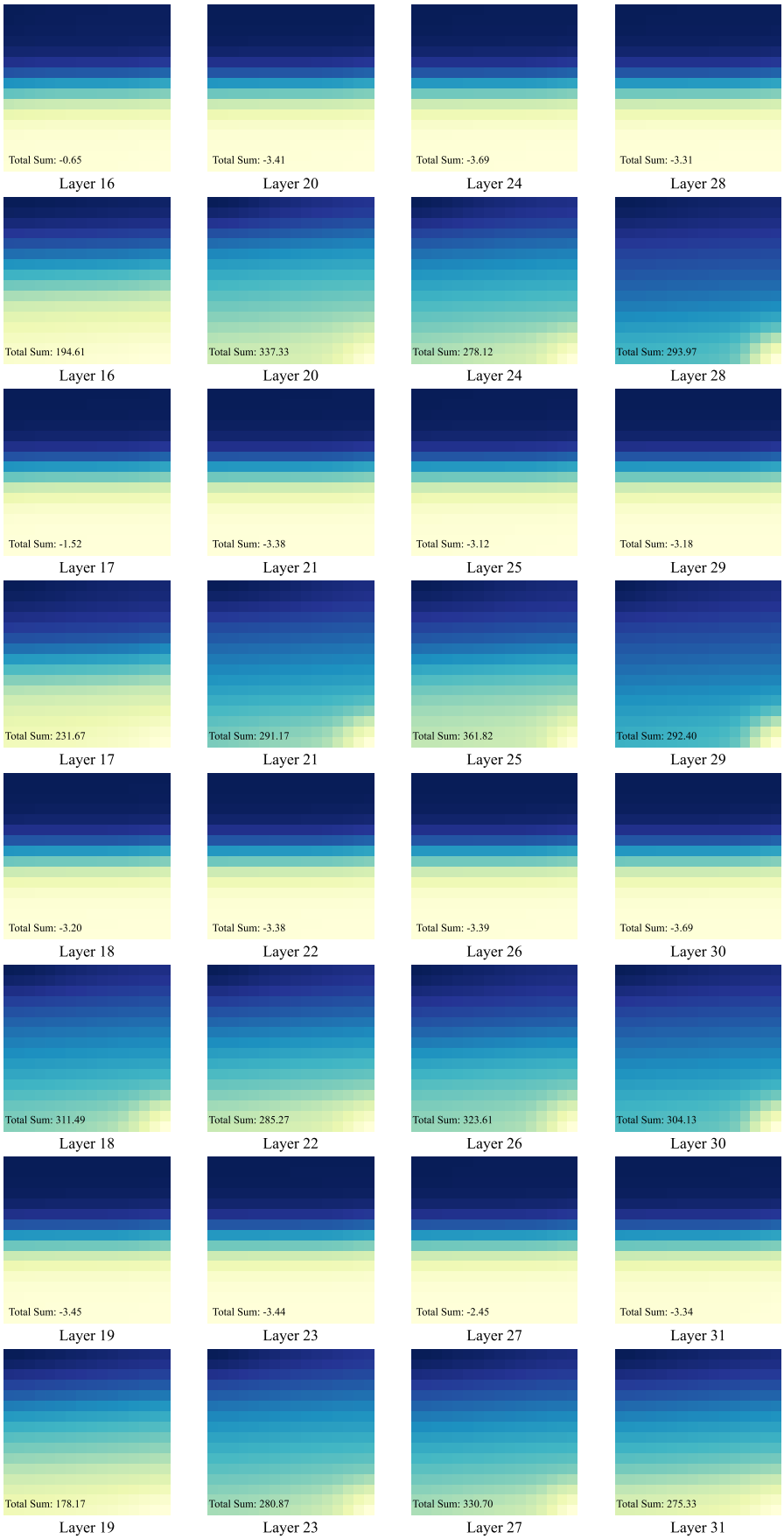

Figure 9: The distribution of the Harsanyi dividend on each layer before (odd-numbered rows) and after (even-numbered rows) applying GAC.

Prompt for classification tasks

- SST2:
  - "Classify the sentiment of the user's message into one of the following categories:'positive' or 'negative'.
    -
    Sentence: <sentence>
    Sentiment: "
- SST5:
  - "Classify the sentiment of the user's message into one of the following categories:'terrible', 'negative', 'neutral', 'positive', or 'great'.
    -
    Sentence: <sentence>
    Sentiment: "
- MR:
  - "Classify the sentiment of the movie's review into one of the following categories:'positive' or 'negative'.
    -
    Review: <sentence>
    Sentiment: "
- AGNews:
  - "Classify the news articles into the categories of 'World', 'Sports', 'Business', or 'Technology'.
    -
    Article: <sentence>
    Category: "
- TREC:
  - "Classify the given questions into the following categories of 'Description', 'Entity', 'Expression', 'Person', 'Number', or 'Location'.
    -
    Question: <sentence>
    Type: "
- CB:
  - "Read the following paragraph and determine if the hypothesis is true.
    -
    Premise: <premise> Hypothesis: <hypothesis>. Answer: "
- BoolQ:
  - "Read the text and answer the question by True or False.
    -
    Text: <passage> Question: <question>?
    Answer: "

Prompt for multi-choice tasks

- Hellaswag:
  - "Complete the following sentence with an appropriate ending."
    <Question>
    <choice 1>
    <choice 2>
    <choice 3>
    ...
    Answer:"
- ARCE:
  - "Generate the correct answer to the following question."
    <Question>
    <choice 1>
    <choice 2>
    <choice 3>
    ...
    Answer:"
- ARCC:
  - "Generate the correct answer to the following question."
    <Question>
    <choice 1>
    <choice 2>
    <choice 3>
    ...
    Answer:"
- PIQA:
  - "Generate the correct solution to accomplish the following goal."
    <Question>
    <choice 1>
    <choice 2>
    <choice 3>
    ...
    Answer:"
- OB:
  - "Generate the most appropriate answer to the following question."
    <Question>
    <choice 1>
    <choice 2>
    <choice 3>
    ...
    Answer:"

Prompt for multi-choice tasks

- CQA:
  - "Generate the correct answer to the following question."
    <Question>
    <choice 1>
    <choice 2>
    <choice 3>
    ...
    Answer:"

Prompt for open-ended question answering

For open-ended question answering, we use the following prompt:

- SQuAD v1
  - Based on the given context, provide only one accurate answer to the question. Follow these rules:
    * Do not include multiple options or explanations.
    * Answer in a single sentence, and do not add any extra text after the answer.

    Title: [title]
    Background: [background]
    Q: [first question]
    A: [first answer]
    Q: [final question]
    A: [completion]

- SQuAD v2
  - Based on the given context, provide only one accurate answer to the question. Follow these rules:
    * Do not include multiple options or explanations.
    * Answer in a single sentence, and do not add any extra text after the answer. If the question cannot be answered based on the context, please reply with 'unanswerable'.

    Title: [title]
    Background: [background]
    Q: [first question]
    A: [first answer]
    Q: [final question]
    A: [completion]