KNOWLEDGE-AWARE ATTENTION STEERING AND REDISTRIBUTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models often fail to faithfully utilize the knowledge provided in context, producing answers that contradict the given evidence. To bridge this gap, we systematically investigate the attention allocation across different functional input regions and its causal contribution to information fusion and final predictions via information-flow analysis and extensive attention interventions. We trace this unfaithfulness to a striking mechanistic root: attention heavily goes where information is not. Through large-scale analysis, we reveal three core mechanistic insights: (1) **Attention-Saliency Imbalance.** The system region absorbs over 60% of attention (up to 80% in LLaMA) yet contributes marginally to prediction, whereas the knowledge region drives information fusion and prediction in mid-to-late layers with merely 16% of attention. (2) **Redundant Attention Bias.** A substantial portion of system attention functions as a structural bias rather than informative computation, and scaling it down is harmless and even yields performance gains. (3) **Attentional Alignment.** Both causal interventions and supervised fine-tuning trajectories confirm that elevating the knowledge attention in mid-to-late layers causally improves knowledge utilization and accuracy. Motivated by these findings, we propose **K**nowledge-Aware **A**ttention **S**teering and **R**edistribution (KASR). During inference, KASR dynamically intercepts the redundant attention from the system region at critical layers and redistributes it to knowledge tokens, thereby amplifying the latent factual signals. KASR is training-free and serves as a plug-and-play inference approach. Extensive experiments on 10 models across 5 knowledge-intensive benchmarks demonstrate that KASR yields consistent improvements, raising accuracy by 3.83% on average with negligible overhead.

1 INTRODUCTION

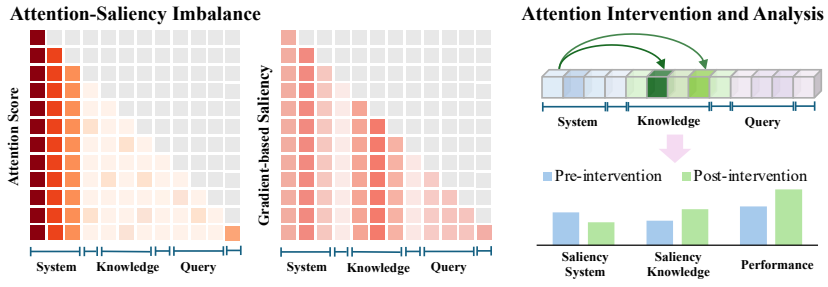


Figure 1: Attention-saliency imbalance across regions (left). KASR reallocates redundant system attention to knowledge, raising knowledge utilization and accuracy (right).

Large Language Models (LLMs) rely on self-attention to capture contextual dependencies and model long-range relations, achieving strong performance in language understanding (Ouyang et al., 2022; Wang et al., 2025), generation (Liu et al., 2024), and information querying (Jiang et al., 2025). Yet in context-grounded reasoning and question answering (Q&A), LLMs still exhibit **knowledge unfaithfulness**, where generated responses contradict or insufficiently utilize the knowledge provided in context, leading to erroneous outputs (Bi et al., 2024; Shi et al., 2024; Huang et al., 2025). This

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

raises a basic question: do LLMs actually attend to and use the provided knowledge during inner inference and computation workflows?

Mechanistic interpretability offers a lens on this failure. A growing body of work documents sparse and systematically biased attention in LLMs(Yin et al., 2025; Wang et al., 2026; Yi et al., 2026). Attention sink effect, for instance, shows that softmax normalization causes LLMs to assign disproportionately high attention to semantically barren anchor tokens, such as initial tokens, punctuation, or formatting placeholders (Jiang et al., 2025; Xiao et al., 2023; Gu et al., 2025). However, this line of work focuses primarily on where attention is allocated (*Attention Allocation*) (Yu et al., 2024b; Zhang et al., 2024; Kang et al., 2025), while leaving open whether that attention actually governs information fusion and predictive computations (*Information Contribution*). In knowledge-intensive tasks particularly, whether attention allocation aligns with causal contribution across distinct functional regions (System, Knowledge, and Query) remains unexamined.

To answer this question, we shift the unit of analysis from individual tokens to functional semantic regions and profile each region's attention allocation against its causal contribution to information fusion and prediction. Via information-flow and gradient-based saliency probing, we unveil a pronounced **Attention-Saliency Imbalance**: in the mid-to-late layers, the system region absorbs the majority of attention yet contributes far below its share, whereas the knowledge region does the opposite, driving information fusion and prediction on relatively little attention. Saliency is a first-order estimate rather than proof of causation, so we further test the imbalance by direct manipulation. Attention interventions and an analysis of supervised fine-tuning (SFT) trajectories both reveal that: **(1) The system region holds a considerable amount of redundant attention that can be suppressed without hurting accuracy; (2) The knowledge region's attention ratio, in the mid-to-late layers, rises and falls together with the intensity of factual processing and overall accuracy.** These results establish that the imbalance is not a benign artifact but a driver of knowledge under-utilization.

Motivated by these observations, we propose Knowledge-Aware Attention Steering and Redistribution (KASR). At layers where knowledge saliency outstrips its attention share (in practice, the mid-to-late layers), KASR reclaims redundant attention of the most heavily attended system tokens and redistributes it directly to the knowledge tokens, amplifying factual signals without renormalizing the whole distribution. Unlike prior steering methods that require offline head profiling or user-specified spans, KASR modifies no parameters, adds no forward pass at inference, and requires no auxiliary model, making it a self-contained and lightweight plug-and-play method (Figure 1). Experiments on five knowledge-intensive benchmarks and ten models from three LLM families demonstrate that KASR improves performance across diverse knowledge formats, model scales, and architectures, achieving an average accuracy gain of 3.83% with negligible inference overhead.

In summary, we reveal an Attention-Saliency Imbalance in knowledge-grounded LLMs: across functional input regions, the attention a region receives is a poor predictor of how much it contributes to the prediction. We validate through attention interventions that system attention is largely redundant and that reallocating it to the knowledge region improves accuracy, with SFT trajectory analysis providing consistent evidence. We propose KASR, a training-free, plug-and-play method that enhances knowledge focus and utilization, turning this analysis into consistent gains across model families, scales, and knowledge-intensive benchmarks.

2 Related Work

Knowledge Utilization in LLMs. Effectively exploiting external knowledge is essential for improving LLM factual consistency and domain capabilities (Gao et al., 2023; Deng et al., 2023). SFT internalizes knowledge through parameter updates but requires training costs and adapts poorly to evolving knowledge (Deng et al., 2023). Retrieval-Augmented Generation injects retrieved evidence into the context, yet LLMs often fail to faithfully utilize the provided knowledge (Wu et al., 2025). To implicitly elevate factual awareness, CAD (Shi et al., 2024), RHIO (Huang et al., 2025), and ATB (Bi et al., 2024) amplify knowledge through contrastive decoding or probability adjustment. However, they incur extra inference cost and, more critically, mitigate knowledge under-utilization without explaining its underlying mechanism.

Attention Analysis in LLMs. Recently, growing efforts have sought to understand LLM inference through the Transformer's internal mechanisms (Sun et al., 2024; Roh et al., 2026; Liu et al.,

2026; Yu et al., 2026; 2024a). Studies on attention sinks reveal that substantial attention is persistently allocated to semantically uninformative tokens, such as initial and formatting tokens (Wang et al., 2026; Gu et al., 2025). StreamLLM (Xiao et al., 2023) interprets them as implicit memory caches, whereas Massive Activation (Sun et al., 2024) attributes them to pre-training biases reflected in highly activated hidden dimensions. Motivated by these findings, several methods directly manipulate attention distributions to steer model behavior (Zhang et al., 2024; Yin et al., 2025; Kang et al., 2025),such as ACT (Yu et al., 2024b), Pasta (Zhang et al., 2024), VAR (Kang et al., 2025), and VAF (Yin et al., 2025). Although these works show that attention modification can steer model performance, they remain largely heuristic and provide little understanding of whether attention across functional input regions truly contributes to information propagation and prediction, or how attention allocation affects knowledge utilization.

Unlike prior methods that rely on output-space optimization or heuristic attention editing, we first analyze how attention allocation relates to information contribution and model behavior, and then derive an analysis-guided attention redistribution strategy.

3 Analysis and Method

This section answers two questions: **(1) Does the attention allocated to different input regions align with their actual predictive contribution? (2) If not, which attention can be safely redistributed to reinforce knowledge utilization?** We first introduce our functional input region partition (Section 3.1), and then dissect the relationship between attention allocation and predictive contribution via gradient-based saliency and attention interventions (Section 3.2). Finally, based on these findings, we propose KASR (Section 3.3).

3.1 Preliminaries

An LLM backbone comprises L Transformer blocks, each with a Multi-Head Attention (MHA) module and a Feed-Forward Network (FFN). We focus on MHA, which governs how information flows between input positions:

$$MHA^{l,h}(x_i^{l-1}) = \sum_{j \le i} A_{i,j}^{l,h} x_j^{l-1} \mathbf{W}_{OV}^{l,h}, \; \boldsymbol{A}_i^{l,h} = softmax(\frac{(x_i^{l-1}\mathbf{W}_Q^{l,h})(\mathbf{X}_{\le i}^{l-1}\mathbf{W}_K^{l,h})^\top}{\sqrt{D}_k}) \quad (1)$$

where $x_i^{l-1} \in \mathbb{R}^D$ is the input of the i-th token of layer l, D is the hidden dimension. $\mathbf{W}_{OV}^{l,h} \in \mathbb{R}^{D\times D}$, $\mathbf{W}_Q^{l,h} \in \mathbb{R}^{D\times D_k}$, and $\mathbf{W}_K^{l,h} \in \mathbb{R}^{D\times D_k}$ are the output-value, query, and key projection of the h-th head at layer l. $\mathbf{X}_{\le i}^{l-1} = [x_0^{l-1}, \ldots, x_i^{l-1}]$ is the prefix up to position i. For a sequence of length N, $A_{N,j}^{l,h}$ measures the model's factual focus on x_j^{l-1} at head h of layer l for next-token prediction.

In knowledge-grounded Q&A and reasoning, different input segments serve distinct semantic functions (White et al., 2023): the *system prompt* defines model behavior, the *knowledge* supplies external evidence, and the *query* specifies the objective. We therefore partition the input into six functional regions listed in Table 2 and analyze attention allocation and contribution at the region level rather than the token level. These regions are defined by their semantic roles rather than specific delimiters or model architectures, allowing the same partitioning scheme to be applied across the diverse model families evaluated in Table 3. This lets us track how much the model's factual focus is on the functional region $\mathcal{I}$ during next-token prediction, i.e., $\mathbf{A}_{N,\mathcal{I}}$. For example, $\boldsymbol{A}_{N,\mathcal{I}_{sys}}$ denotes the attention weight allocated to the system region $\mathcal{I}_{sys}$ when producing the target prediction.

3.2 Empirical Analysis of Knowledge Attention

We investigate LLMs' internal mechanisms in knowledge-grounded Q&A from three angles: ***attention allocation***, ***information-flow contribution***, and ***causal attention intervention***. The first two reveal that the attention a region receives is a poor predictor of how much it contributes; the third shows that correcting this misallocation improves knowledge utilization. For clarity, we report the results for Qwen2.5-1.5B (Bai et al., 2023) on the DXY (Yan et al., 2025) medical diagnosis benchmark; the same conclusions hold across 10 LLMs from 3 families and 5 benchmarks. Additional settings, prompts, and results are given in the Appendix A.

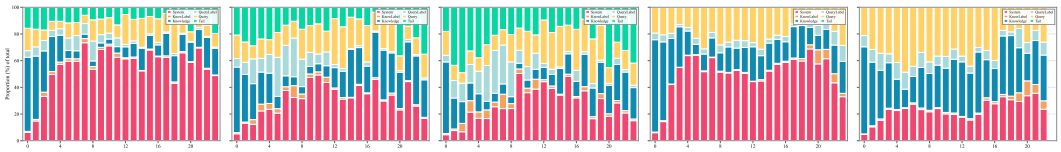


Figure 2: (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

3.2.1 Attention-Saliency Imbalance

Attention weights from the prediction position to preceding tokens are commonly read as the model's contextual focus (Kang et al., 2025; Zhang et al., 2024). Whether that reading holds is an empirical question: a token can be heavily attended yet barely affect the output. We leverage gradient-based saliency technique to quantify how much each functional region actually shapes the prediction. Let $\mathcal{L}$ be the loss of the gold answer. By a first-order Taylor expansion (Wang et al., 2023), the saliency score of each element within the attention matrix is approximated as:

$$\mathcal{L}(A^{l,h}+\Delta A^{l,h}) \approx \mathcal{L}(A^{l,h}) + \left\langle \frac{\partial \mathcal{L}}{\partial A^{l,h}}, \Delta A^{l,h} \right\rangle, \quad \Delta\mathcal{L} \approx \left\langle \frac{\partial \mathcal{L}}{\partial A^{l,h}}, A^{l,h} \right\rangle = \sum_{i,j} A^{l,h}_{i,j} \cdot \frac{\partial \mathcal{L}}{\partial A^{l,h}_{i,j}} \quad (2)$$

where $\langle \cdot, \cdot \rangle$ is the Frobenius inner product, so $A^{l,h}_{i,j} \cdot \frac{\partial \mathcal{L}}{\partial A^{l,h}_{i,j}}$ measures how much the connection from position j to position i in head h of layer l matters to the prediction. Aggregating over heads yields a layer-wise saliency matrix:

$$I_l = \left| \sum_h A^{l,h} \cdot \frac{\partial \mathcal{L}}{\partial A^{l,h}} \right| \quad (3)$$

From I_l, we define two region-level saliency metrics. The first is the share of prediction saliency contributed by region $\mathcal{I}$; the second is the share of information flowing from $\mathcal{I}$ into the query region:

$$S_l(N,\mathcal{I}) = \frac{\sum_{j\in\mathcal{I}} I_l(N,j)}{\sum_{j\le N} I_l(N,j)}, \quad S_l(\mathcal{I}_{qry},\mathcal{I}) = \frac{\sum_{i\in\mathcal{I}_{qry}}\sum_{j\in\mathcal{I}} I_l(i,j)}{\sum_{i\in\mathcal{I}_{qry}}\sum_{j\le i} I_l(i,j)} \quad (4)$$

In both metrics, the first argument denotes where information arrives and the second where it comes from, so $S_l(N,\mathcal{I}_{know})$ and $S_l(\mathcal{I}_{qry},\mathcal{I}_{know})$ measure how much the knowledge region drives the prediction and how much of it reaches the query, respectively.

We further evaluate the regional contribution to residual stream updates, reflecting how much information from each region is incorporated into hidden-state representations:

$$R_l(N,\mathcal{I}) = \frac{\left|\left|\sum_{i\in\mathcal{I}} A^l_{N,i} x^{l-1}_i W^l_{OV}\right|\right|_2}{\left|\left|\sum_{i\le N} A^l_{N,i} x^{l-1}_i W^l_{OV}\right|\right|_2} \quad (5)$$

Figure 2 shows the attention distribution and the saliency contribution for next-token generation and regional interaction. Based on these results, we derive three key insights:

(i) Attention-saliency imbalance across input regions. Beyond the initial syntactic layers, the system region absorbs more than 60% of the attention (nearly 80% in LLaMA) while its saliency share stays well below that. Conversely, the knowledge region receives only 16.12%, but drives nearly twice

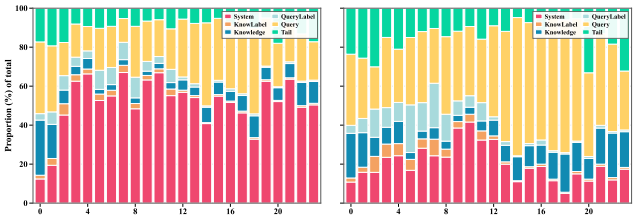


Figure 3: The attention allocation and saliency contribution after modifying system prompt.

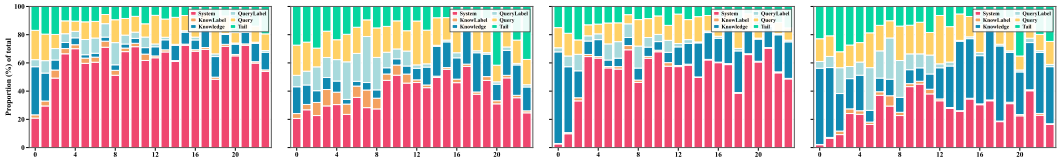


Figure 4: (a) and (b) show the attention–saliency patterns with task-relevant triplet knowledge, while (c) and (d) show those without task-relevant knowledge.

saliency. This indicates that attention allocation is not a faithful indicator of a region's importance, and unveils a pronounced resource misallocation in knowledge-based tasks.

(ii) System attention is largely redundant and under-exploited. To examine whether attention assigned to $\mathcal{I}_{sys}$ is indispensable, we move the disease label list from the system prompt to the query region, cutting the semantic load of $\mathcal{I}_{sys}$. Figure 3 shows that system attention drops only slightly and the attention-saliency imbalance widens. Inspired by Massive Activation (Sun et al., 2024), which suggests that highly activated hidden states function primarily as training biases, we take this as evidence that much of system attention serves as a structural bias: system tokens precede everything else under causal masking, so they are the natural sink for the attention mass that softmax must place somewhere.

(iii) Knowledge dominates the mid-to-late layers. Both $S_l(N, \mathcal{I}_{know})$ and $S_l(\mathcal{I}_{qry}, \mathcal{I}_{know})$ rise sharply in the mid-to-late layers, indicating that knowledge plays an increasingly important role in contextual interaction at this stage and drives the final prediction. This trend holds across knowledge formats (textual vs. triplets), knowledge relevance (gold vs. ungrounded context), model scales, and architectures. Figure 4 adds that *when related knowledge is absent, both attention allocation and saliency of $\mathcal{I}_{know}$ rise slightly*, as if the model spends more of its budget there when the prediction is more challenging. This suggests that the mid-to-late layers are where knowledge processing concentrates, making them the natural target for intervention.

3.2.2 Attention Interventions and SFT Trajectories

The imbalance above is correlational: saliency tells us which connections the loss is sensitive to, not what happens if we change them. We therefore manipulate attention directly, and separately track how SFT reshapes it, to ask whether knowledge attention is a lever on knowledge utilization or merely a symptom of it.

Attention intervention. We rescale the post-softmax attention weights of a target region $\mathcal{I}$ and renormalize the distribution as:

$$\hat{A}_{i,j} = \frac{\beta A_{i,j}}{\mathcal{T}}, j \in \lambda\mathcal{I} \wedge i \geq j, \quad \mathcal{T} = \sum_{j \in \lambda\mathcal{I}} \beta A_{i,j} + \sum_{j \notin \mathcal{I}} A_{i,j} \tag{6}$$

where $\beta < 1$ suppresses attention and $\beta > 1$ amplifies it, while λ denotes the proportion of selected tokens within the target region. We perform both global and layer-wise interventions on each functional region independently. Figure 5 reports a representative subset: global system attenuation with $\lambda \in \{0.1, 0.2, 0.3, 0.5, 0.75, 1.0\}$ and $\beta \in \{0.0, 0.25, 0.5, 0.75\}$, and layer-wise amplification with $\lambda \in \{1.0, 0.5\}$ and $\beta \in \{1.5, 2.0\}$.

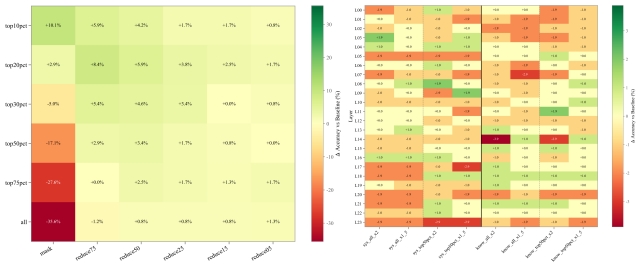


Figure 5: Model performance under (a) global system-attention attenuation and (b) layer-wise attention amplification. In (a), the y-axis denotes λ and the x-axis the attenuation ratio. In (b), the y-axis denotes the Transformer layer and the x-axis the intervention setting (e.g., know_all_x2 doubles the attention of all knowledge tokens before renormalization).

Observations: (1) Severely disrupting attention in any region consistently degrades performance, indicating that the model depends on stable attention patterns and that naive masking is harmful. (2) Except

under aggressive interventions, attenuating $\mathcal{I}_{sys}$ attention is largely harmless and often improves performance. (3) Amplifying $\mathcal{I}_{know}$ in the mid-to-late layers generally helps, whereas amplifying $\mathcal{I}_{sys}$ does not.

Part of the attention assigned to $\mathcal{I}_{sys}$ is thus redundant and can be safely redistributed. To further uncover *why pruning system attention improves performance* and *where the released attention goes*, we inspect internal attention dynamics under interventions and training.

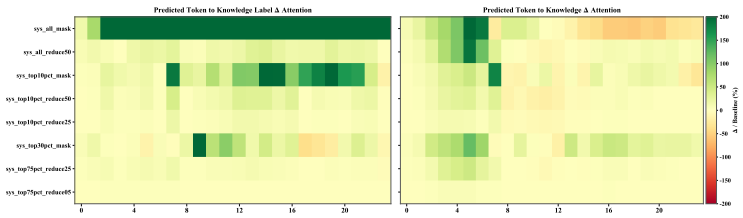


Figure 6: $\Delta \boldsymbol{A}_{*N,\mathcal{I}*know_label}$ and $\Delta \boldsymbol{A}_{*N,\mathcal{I}*know}$ after $\mathcal{I}*sys$ attention attenuation. The x-axis denotes Transformer layers, and the y-axis denotes the $\mathcal{I} * sys$ intervention settings.

Where the released attention goes. Figure 6 tracks $\Delta \boldsymbol{A}_{N,\mathcal{I}_{know}}$ and $\Delta \boldsymbol{A}_{N,\mathcal{I}_{know_label}}$ under interventions that produce substantial performance changes in Figure 5.

Observation. Nearly all performance gains coincide with increased attention to the knowledge region, especially in the mid-to-late Transformer layers. This underscores that reducing system attention does not directly improve performance; instead, it frees attention budget for the knowledge region, enhancing knowledge utilization. Performance degradation, however, is often accompanied by excessive attention to $\mathcal{I}_{know_label}$. For example, masking all system tokens (sys_all_mask) increases $\mathcal{I}_{know_label}$ attention by more than 100×. We attribute this to the loss of architectural attention sinks in $\mathcal{I}_{sys}$, which forces $\mathcal{I}_{know_label}$ to absorb the structural biases inherent to softmax normalization, ultimately disrupting internal Transformer dynamics.

Attention changes under SFT. Interventions perturb a fixed model; SFT instead lets the model reorganize its own attention while internalizing the same knowledge through cross-entropy training. Figure 7 compares regional attention and saliency before and after fine-tuning.

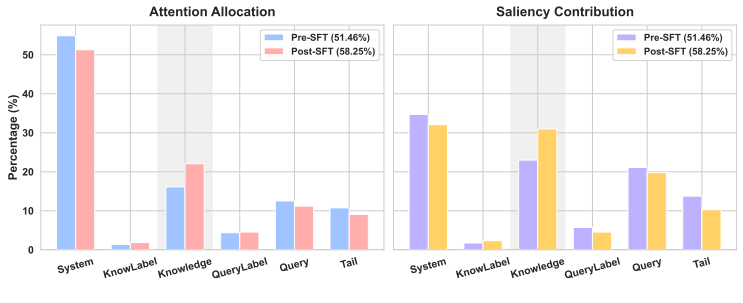


Figure 7: Changes in regional attention allocation and saliency contribution before and after SFT.

Observation: SFT improves accuracy by 6.79%, with a 5.94% increase in knowledge attention and a 7.98% increase in its prediction saliency, while both attention and saliency in the system region decrease. Training thus reallocates attention in the same direction as our interventions, which consistently shows a strong positive relationship among knowledge attention, knowledge processing, and model performance.

3.3 Knowledge-Aware Attention Steering and Redistribution

Based on the analyses in Section 3.2, we propose KASR, a saliency-guided attention redistribution method. KASR reallocates attention in layers where Knowledge exhibits high saliency relative to its attention allocation, thereby targeting the strongest Attention–Saliency Imbalance. For an L-layer Transformer, the selected layers are defined as:

$$r_l = \frac{1}{|\mathcal{D}_{\text{val}}|} \sum_{i \in \mathcal{D}_{\text{val}}} \frac{S_l(i, \mathcal{I}_{\text{know}})}{\sum_{j \in \mathcal{I}_{\text{know}}} A^l_{i,j}}, \quad L^* = \text{TopK}_{l \in \{1,\dots,L\}} \left(r_l, \lceil 0.5L \rceil\right) \tag{7}$$

Specifically, we rank all layers by the ratio between the saliency contribution of the Knowledge region and its allocated attention, and select the top 50% for redistribution. A higher ratio indicates

a stronger mismatch between information contribution and attention allocation, identifying layers where Knowledge is under-attended relative to its predictive contribution. Since saliency computation requires ground-truth labels, L^* is determined exclusively from the validation set and remains fixed during test-time inference. At test time, KASR only computes input-dependent attention online and redistributes it within the predefined layers, requiring no test labels.

Unlike the empirical L_1 re-normalization in Section 3.2.2, KASR performs targeted attention redistribution. After selecting the saliency-critical layers, the attention reclaimed from $\mathcal{I}_{sys}$ is reallocated proportionally to knowledge tokens:

$$\Delta A^l = (1-\beta) \cdot \sum_{j \in \lambda \mathcal{I}_{sys}} A^l_{i,j},\ \hat{A}^l_{i,k} = A^l_{i,k} + \Delta A^l \cdot \frac{A^l_{i,k}}{\sum_{k \in \mathcal{I}_{know}} A^l_{i,k}} \tag{8}$$

where $\beta \in (0, 1]$ represents the retention ratio of the system attention. By directly transferring the attention from the system region to informative knowledge tokens, KASR avoids introducing new attention biases while satisfying the sum-to-one constraint. KASR enhances knowledge utilization during inference without modifying model parameters or requiring any additional training, making it a lightweight plug-and-play inference strategy.

4 Experiments

In this section, we evaluate KASR against representative baselines and further analyze its impact on internal attention dynamics and inference efficiency (Section 4.2). We then conduct ablation studies on the redistribution ratio, target regions, and intervention layers (Section 4.3). *Further implementation details and result analyses are provided in the Appendix B.1.*

4.1 Experiment setup

Implementation. KASR is a training-free inference-time method and can be simply applied to various LLMs without additional training, models, or inference steps. We employ 10 open-source LLMs with various scales and versions as our base models, which span three foundational architectures: Qwen (Bai et al., 2023), LLaMA (Touvron et al., 2023), and InternLM (Cai et al., 2024). All experiments are implemented in PyTorch and evaluated on A800 80GB GPUs, with results averaged over three runs. Unless otherwise specified, we set $\lambda = 0.1$.

Baselines. To comprehensively evaluate KASR, we compare it against training-free contrastive decoding and heuristic attention-intervention baselines: CAD (Shi et al., 2024) amplifies knowledge by computing a contrastive logit distribution between predictions conditioned on correct and perturbed knowledge. Both ACT (Yu et al., 2024b) and PASTA (Zhang et al., 2024) identify effective attention heads on a validation set and reuse them during test-time inference. ACT identifies sink tokens whose attention ratio exceeds $5/N$, redistributes 60% of their attention to the remaining tokens, and retains the heads yielding performance gains. PASTA reallocates attention from non-instruction tokens to instruction tokens and retains the top 1% performing heads. Following its original implementation, we adapt PASTA by treating the knowledge region as the target instruction region.

Datasets. We evaluate KASR on 5 knowledge-intensive benchmarks spanning 3 categories. (1) Knowledge reasoning. DXY (Yan et al., 2025) and GMD (Liu et al., 2022) require models to predict the final diagnosis from candidate diseases by reasoning over provided medical knowledge and patient symptoms. (2) Professional knowledge Q&A. MMLU-Professional Law (LAW) and MMLU-Professional Medicine (MED) (Wang et al., 2024) demand multi-choice reasoning anchored in specialized domain expertise. (3) Fact-based reasoning. StrategyQA (Geva et al., 2021) requires multi-hop reasoning over multiple background facts. For datasets with explicit knowledge inputs (DXY, GMD, and StrategyQA), we partition the input into 6 functional regions (Section 3.1). For MMLU, which lacks explicit background context, we instead divide the input into 4 regions: $\mathcal{I}_{sys}$, $\mathcal{I}_{qry_label}$, $\mathcal{I}_{qry}$, and $\mathcal{I}_{tail}$, and redistribute attention from $\mathcal{I}_{sys}$ to $\mathcal{I}_{qry}$.

We report accuracy (ACC.) for all benchmarks. Since the MMLU datasets contain only queries and answers without external knowledge, CAD is not applicable to them.

4.2 Main Analysis

Table 1 reports representative results, while the complete results for all 10 models are provided in Appendix B.1. Based on these results, we summarize the key findings:

Table 1: Comparison between KASR and baselines on five benchmarks.

Backbone	Method	DXY	GMD	StrategyQA	LAW	MED
InternLM2.5-1.8B	Base	44.66	45.61	67.47	30.59	54.84
	CAD	**48.54**	35.98	70.96	-	-
	ACT	34.95	26.78	57.21	28.24	16.13
	PASTA	**48.54**	8.37	70.52	29.41	54.84
	KASR(Ours)	46.6	**58.58**	**71.40**	**33.53**	**61.29**
Llama3.1-8B	Base	50.49	61.09	89.30	48.82	77.42
	CAD	50.49	59.41	90.39	-	-
	ACT	50.49	7.11	46.72	22.35	51.61
	PASTA	50.49	60.67	87.77	53.53	77.42
	KASR(Ours)	**59.22**	**67.78**	**90.61**	**54.71**	**90.32**
Llama3.2-3B	Base	30.10	35.15	78.82	41.76	74.19
	CAD	30.10	36.82	81.22	-	-
	ACT	31.07	7.11	46.29	41.76	19.35
	PASTA	31.07	46.86	76.64	40.59	80.65
	KASR(Ours)	**42.72**	**48.95**	**81.66**	**44.71**	**83.87**
Qwen2.5-3B	Base	41.75	62.34	73.80	41.76	77.42
	CAD	42.72	62.76	77.73	-	-
	ACT	18.45	30.96	49.13	25.29	67.74
	PASTA	47.57	64.02	73.58	42.35	74.19
	KASR(Ours)	**48.54**	**73.22**	**79.26**	**47.65**	**80.65**
Qwen2.5-7B	Base	49.51	58.16	86.24	52.94	74.19
	CAD	49.51	57.74	**90.61**	-	-
	ACT	26.21	27.62	89.30	41.18	61.29
	PASTA	52.43	65.17	88.43	53.53	74.19
	KASR(Ours)	**58.25**	**68.62**	87.99	**55.88**	**77.42**

KASR Consistently Enhances Knowledge Q&A. Across all benchmarks, KASR yields an average improvement of 3.83% over the strongest baseline. These results underscore that redistributing redundant attention from $\mathcal{I}_{sys}$ within mid-to-late layers is an effective strategy for enhancing the model's focus on and processing of knowledge. In contrast, CAD often exhibits unstable performance. We attribute this limitation to its reliance on the discrepancy between model's internal knowledge and external knowledge. When the backbone model already possesses robust domain expertise, the distributions conditioned on correct and perturbed contexts become similar, limiting contrastive decoding to amplify knowledge signals. KASR, instead, directly enhances attention to knowledge, making it less sensitive to the knowledge already stored in model parameters. Similar to CAD, we anticipate KASR provides larger gains on tasks where external knowledge plays a more decisive role in prediction.

KASR Provides a More Principled Attention Intervention Strategy. Compared with the heuristic attention intervention methods ACT and PASTA, KASR consistently achieves better performance, validating the effectiveness of saliency-guided attention redistribution. ACT identifies intervention heads through validation-set search and suppresses tokens whose attention exceeds a predefined threshold. Since high attention does not necessarily indicate redundant attention, this heuristic may also attenuate informative tokens, limiting its effectiveness. PASTA alleviates this issue by explicitly redirecting attention toward the target region and restricting interventions to only the top 1% empirically selected heads, leading to stronger performance than ACT. However, its intervention strategy still relies on validation-set search and remains agnostic to the actual information contribution of different regions. In contrast, KASR derives both the intervention layers and the redistribution strategy directly from the mismatch between attention allocation and information contribution, resulting in a more principled and effective attention intervention strategy.

KASR Generalizes Well Across Models and Tasks. KASR consistently improves performance across diverse knowledge-intensive Q&A and fact-reasoning benchmarks. The gains are largely independent of model architecture and parameter size, demonstrating strong generalizability.

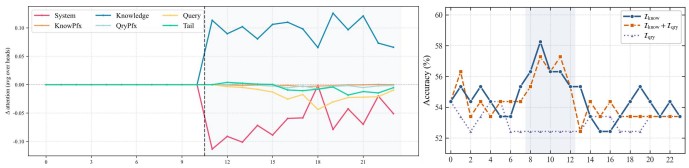


Figure 8: (a) Regional attention changes under the performance-improving KASR intervention setting. (b) Performance comparison when attention redistribution starts from different Transformer layers.

Effectiveness Analysis. To further verify whether KASR actually improves the model's focus on external knowledge, we profile the post-intervention attention distribution across functional input regions in Figure 8(a). KASR yields a prominent surge in attention weights within the knowledge region in the mid-to-late Transformer layers while correspondingly reducing attention to the system region. This observation is fully consistent with the analysis in Section 3.2 and further supports our central hypothesis that increasing attention to high-saliency knowledge regions leads to more effective knowledge utilization.

4.3 Ablations and Discussions

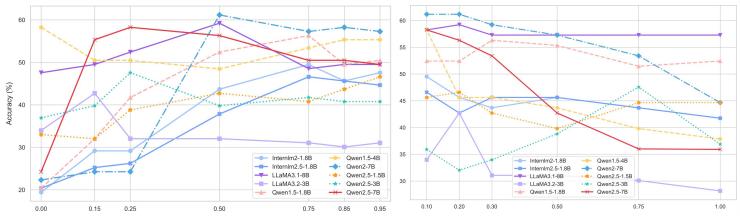


Figure 9: (a) Model performance under different β. (b) Ablation on λ.

Impact of β**.** Figure 9(a) reveals a clear trend: both the excessively small and the excessively large values of β lead to inferior performance. When $\beta = 0$, the selected tokens in the system are completely masked, which tends to create the new attention biases discussed in Section 3.2.2, disrupting the structure of the learned attention and degrading the accuracy. Moreover, an overly large β generates an insufficient budget of liberated attention, failing to effectively amplify the model's focus on knowledge.

Effect of λ**.** In the main experiments, we set $\lambda = 0.1$, redistributing attention from only the top 10% highest-attention system tokens. We investigate the effect of λ, as presented in Figure 9(b). Across different models, $\lambda \in \{0.1, 0.2\}$ consistently outperforms larger values. This phenomenon arises because, although the system region contains considerable redundant attention, it also includes indispensable instructional information, such as the candidate disease list in DXY. Increasing λ inadvertently disrupts more genuinely critical system tokens, thereby degrading model performance.

These findings raise a natural question: which system tokens should be selected for attention reallocation? The current attention-based criterion is only a coarse approximation. We provide a preliminary exploration in *Appendix B.2*, leaving more precise token selection strategies as an essential direction for future work.

Effect of Intervention Stage. Our analyses in Section 3.2.2 and 4.2 suggest that reallocating redundant attention to the knowledge region is most beneficial in the middle-to-late layers, indicating that knowledge enhancement should begin at the intermediate information fusion stage. We therefore ablate the starting layer for attention redistribution and compare three target regions: $\mathcal{I}_{know}$, $\mathcal{I}_{qry}$, and $\mathcal{I}_{know} + \mathcal{I}_{qry}$.

We observe in Figure 8(b) that initiating attention redistribution from intermediate layers consistently outperforms other stages. A preliminary explanation is that knowledge plays a critical role in information-flow interactions within these intermediate layers; delaying attention enhancement until the late stages limits its ability to participate in the information fusion. Conversely, the benefit of amplifying $\mathcal{I}_{qry}$ attention increases in deeper layers. This is likely because the final layers are primarily responsible for prediction, during which the model frequently revisits the query to generate the next token. Strengthening query attention at this stage therefore provides additional gains.

5 Conclusion

We investigate the underlying cause of knowledge unfaithfulness in knowledge-grounded Q&A from Transformer's internal attention mechanism. By quantifying regional attention allocations against their respective contributions to prediction and information flow, we uncover a pronounced Attention-Saliency Imbalance and reveal the critical role of knowledge in mid-to-late Transformer layers. Through causal attention interventions and post-SFT internal state changes, we confirm a stable, positive correlation among knowledge attention, knowledge utilization, and model performance. Driven by these insights, we propose Knowledge-Aware Attention Steering and Redistribution (KASR), a training-free inference method that dynamically reallocates redundant attention from system region to knowledge region, thereby enhancing knowledge utilization. Extensive experiments on 10 models from 3 prominent open-source LLM families and 5 knowledge-intensive benchmarks demonstrate that KASR outperforms strong baselines by 3.83% while introducing negligible inference overhead. Further analyses verify its robustness and generalizability. Our findings reveal the intrinsic interplay between attention allocation and knowledge utilization and offer a principled foundation for designing more interpretable attention intervention methods.

AI USE STATEMENT

In this work, we used generative AI tools solely to assist with translation and language polishing to improve the clarity, grammar, and readability of the manuscript. Generative AI tools were not used for research ideation, methodology or experimental design, implementation, data processing, result analysis or interpretation, or the formulation of scientific claims. All AI-assisted text was carefully reviewed and revised by the authors to ensure its accuracy and consistency with the original intended meaning. We take full responsibility for the final content of this work, including any text produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We ensure the reproducibility of our work by providing all code and data used in our experiments through an anonymous repository at `https://anonymous.4open.science/r/KASR-DF3C`. Detailed experimental settings, including model configurations, datasets, evaluation protocols, and implementation details, are described in the experimental section and Appendix. The Appendix further provides additional analyses, experimental results, and baseline specifications to facilitate reproduction of our findings. All code and data will be made publicly available upon publication.

REFERENCES

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

Baolong Bi, Shenghua Liu, Yiwei Wang, Lingrui Mei, Hongcheng Gao, Yilong Xu, and Xueqi Cheng. Adaptive token biaser: Knowledge editing via biasing key entities. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 11071–11083, 2024.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, et al. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*, 2024.

Yifan Deng, Xingsheng Zhang, He-Yan Huang, and Yue Hu. Towards faithful dialogues via focus learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4554–4566, 2023.

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

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021.

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. When attention sink emerges in language models: An empirical view. In *International Conference on Learning Representations*, volume 2025, pp. 97114–97144, 2025.

Lei Huang, Xiaocheng Feng, Weitao Ma, Yuchun Fan, Xiachong Feng, Yangfan Ye, Weihong Zhong, Yuxuan Gu, Baoxin Wang, Dayong Wu, et al. Improving contextual faithfulness of large language models via retrieval heads-induced optimization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16896–16913, 2025.

Jingchi Jiang, Lian Yan, Haifeng Liu, Zhenbo Xia, Haotian Wang, Yang Yang, and Yi Guan. Knowledge assimilation: Implementing knowledge-guided agricultural large language model. *Knowledge-based systems*, 314:113197, 2025.

Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. See what you are told: Visual attention sink in large multimodal models. *arXiv preprint arXiv:2503.03321*, 2025.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

Wenge Liu, Yi Cheng, Hao Wang, Jianheng Tang, Yafei Liu, Ruihui Zhao, Wenjie Li, Yefeng Zheng, and Xiaodan Liang. "my nose is running.""are you also coughing?": Building a medical diagnosis agent with interpretable inquiry logics. In *Proceedings of the International Conferences on Artificial Intelligence*, 2022.

Xu Liu, Guikun Chen, and Wenguan Wang. Sinktrack: Attention sink based context anchoring for large language models. *arXiv preprint arXiv:2604.10027*, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Youngji Roh, Hyunjin Cho, and Jaehyung Kim. Embracing anisotropy: Turning massive activations into interpretable control knobs for large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29930–29956, 2026.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. Trusting your evidence: Hallucinate less with context-aware decoding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 783–791, 2024.

Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. Massive activations in large language models. *arXiv preprint arXiv:2402.17762*, 2024.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Haotian Wang, Xiyuan Du, Weijiang Yu, Qianglong Chen, Kun Zhu, Zheng Chu, Lian Yan, and Yi Guan. Learning to break: Knowledge-enhanced reasoning in multi-agent debate system. *Neurocomputing*, 618:129063, 2025.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 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*, pp. 9840–9855, 2023.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, 2024.

Ziheng Wang, Zihao Yue, Wenxuan Wang, and Qin Jin. Exploring attention attractors in large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1148–1160, 2026.

Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C Schmidt. A prompt pattern catalog to enhance prompt engineering with chatgpt. *arXiv preprint arXiv:2302.11382*, 2023.

Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. In *International Conference on Learning Representations*, volume 2025, pp. 62143–62156, 2025.

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

Lian Yan, Chen Tang, Yi Guan, Haotian Wang, Songyuan Wang, Haifeng Liu, Yang Yang, and Jingchi Jiang. Rlkgf: Reinforcement learning from knowledge graph feedback without human annotations. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 6619–6633, 2025.

Zihao Yi, Zhenqing Ling, Delong Zeng, Haohao Luo, Zhe Xu, Wei Liu, Jian Luan, Wanxia Cao, and Ying Shen. Attention basin: Why contextual position matters in large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 26086–26106, 2026.

Hao Yin, Guangzong Si, and Zilei Wang. Clearsight: Visual signal enhancement for object hallucination mitigation in multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 14625–14634, 2025.

Xuehui Yu, Mhairi Dunion, Xin Li, and Stefano V Albrecht. Skill-aware mutual information optimisation for zero-shot generalisation in reinforcement learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a. URL https://openreview.net/forum?id=GtbwJ6mruI.

Xuehui Yu, Fucheng Cai, WANG MEIYI, Xiaopeng Fan, and Harold Soh. Conflict-aware additive guidance for flow models under compositional rewards. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=DJtX0Fo7ii.

Zhongzhi Yu, Zheng Wang, Yonggan Fu, Huihong Shi, Khalid Shaikh, and Yingyan Celine Lin. Unveiling and harnessing hidden attention sinks: Enhancing large language models without training through attention calibration. *arXiv preprint arXiv:2406.15765*, 2024b.

Qingru Zhang, Chandan Singh, Liyuan Liu, Xiaodong Liu, Bin Yu, Jianfeng Gao, and Tuo Zhao. Tell your model where to attend: Post-hoc attention steering for llms. In *International Conference on Learning Representations*, volume 2024, pp. 42411–42430, 2024.

A Analysis and Method

A.1 Functional Input Region Partition

In knowledge-grounded Q&A and reasoning tasks, different input tokens serve distinct semantic functions. Specifically, the *System Prompt* specifies model behaviors, the *Knowledge* provides external factual evidence, and the *Query* dictates the target objective. Consequently, we partition the input sequence into functional semantic regions and analyze attention allocation and predictive contribution at the region level rather than the individual-token level. Following the prompt template, we segment the input into six functional regions, as detailed in Table 2.

A.2 Empirical Analysis of Knowledge Attention

In this section, we systematically investigate the internal mechanisms of LLMs in knowledge-grounded Q&A across three analytical dimensions: attention allocation, information-flow contribution, and causal attention intervention. These analyses provide answers to the two questions raised above: (1) Does attention allocation faithfully reflect the predictive importance of different input regions? (2) Which attention should be redistributed? We conduct our study on 10 LLMs spanning 3 mainstream model families, multiple model versions and scales across five benchmarks. The evaluated models are listed in Table 3, dataset statistics are summarized in Table 4, and the prompts used for each benchmark are shown in Figure 10.

A.2.1 Attention-Saliency Imbalance

In the main paper, we present the attention allocation, saliency contribution, and residual-stream contribution of different functional regions for Qwen1.5-1.8B (Bai et al., 2023) on DXY (Yan et al., 2025). In this section, we extend the analysis to larger Qwen models (Qwen2.5-3B (Figure 11) and Qwen2.5-7B (Figure 12)), LLaMA3.2-3B (Touvron et al., 2023) (Figure 13), LLaMA3.1-8B

Table 2: Protocol and examples for input functional regions.

Region	Description	Example
System $\mathcal{I}_{sys}$	System prompt, including special tokens and task instructions.	<|im_start|>system\n{SYSTEM_PROMPT}<|im_end|>\n
Knowledge Label $\mathcal{I}_{know_label}$	Marker indicating the beginning of the knowledge.	<|im_start|>user\nContext:\n
Knowledge $\mathcal{I}_{know}$	External factual knowledge.	sample['knowledge']
Query Label $\mathcal{I}_{qry_label}$	Marker indicating the beginning of the query.	\n\nQuestion:\n
Query $\mathcal{I}_{qry}$	User query.	sample['query']
Tail $\mathcal{I}_{tail}$	Special tokens indicating the start of generation.	<|im_end|>\n<|im_start|>assistant\n

Table 3: LLMs used in our experiments.

Base Model	Variant
Llama3.2	Llama3.2-3B-Instruct
Llama3.1	Llama3.1-8B-Instruct
Qwen2.5	Qwen2.5-7B-Instruct Qwen2.5-3B-Instruct Qwen2.5-1.5B-Instruct
Qwen2	Qwen2-7B-Instruct
Qwen1.5	Qwen1.5-4B-Instruct Qwen1.5-1.8B-Instruct
InternLM2.5	InternLM2.5-1.8B-Chat
InternLM2	InternLM2-1.8B-Chat

(Figure 14), and InternLM2.5-1.8B (Cai et al., 2024) (Figure 15) on DXY. Specifically, we report the attention allocation from both the target prediction token and the query region to different functional regions, together with each region's saliency contribution to prediction, residual-stream updates, and information flow. To further demonstrate the generality of our findings, we additionally present results for InternLM2.5-1.8B (Figure 18), LLaMA3.1-8B (Figure 16), and Qwen2.5-1.5B (Figure 17) on GMD (Liu et al., 2022), LLaMA3.2-3B (Figure 19) on MMLU-LAW (Wang et al., 2024), and Qwen1.5-1.8B (Figure 20) on StrategyQA (Geva et al., 2021). Across all models and benchmarks, the results consistently exhibit three key findings: (1) an Attention–Saliency Imbalance across functional regions; (2) substantial redundant attention in the system region that contributes little to computation; and (3) the increasingly important role of the knowledge region in information flow and prediction during the mid-to-late layers.

We further analyze how attention allocation and saliency vary across different knowledge formats and knowledge relevance (i.e., whether the provided background knowledge contains the disease-specific knowledge required for the current patient). Figures 21 and 22 present the attention allocation and saliency contributions of Qwen2.5-1.5B on GMD with textual background knowledge, comparing cases with and without task-relevant knowledge. Figures 23 and 24 present the corresponding results for Qwen1.5-1.8B on DXY with triplet-based knowledge. Beyond providing further evidence for the three findings above, these results demonstrate that they are consistent across knowledge formats, knowledge relevance, model scales, and architectures. Moreover, when task-relevant knowledge is absent, both the attention allocated to the knowledge region and its saliency increase slightly, suggesting that the model allocates more computational budget to the knowledge region when prediction becomes more challenging.

Table 4: Dataset statistics. # Val. and # Test denote the numbers of validation and test samples, respectively.

Dataset	Language	# Val.	# Test
DXY	Chinese	421	103
GMD	Chinese	1912	239
MMLU-LAW	English	1534	170
MMLU-MED	English	272	31
StrategyQA	English	1832	458

Normal Prompt for DXY & GMD

```
SYSTEM_PROMPT:你是一个疾病诊断专家。请根据患者症状描述和提供的医学知识图谱，从以下疾病中选出最可能的一种，直接输出疾病名称，不要输出任何其他内容。\n可能的疾病：{'、'.join(DISEASE_LIST)}。
(You are a medical diagnosis expert. Given the patient's symptoms and the provided medical knowledge graph, identify the most likely disease from the following candidates. Output only the disease name and nothing else. \n Candidate diseases: {'; '.join(DISEASE_LIST)}.)
msgs = [
    {"role": "system", "content": SYSTEM_PROMPT},
    {"role": "user",   "content":
      f"Context:\n{sample['knowledge_summary_simple']}\n\n"
      f"Question:\n{sample['current_q']}"},
]
```

Disease List Moved to the Query for DXY & GMD

```
SYSTEM_PROMPT:你是一个疾病诊断专家。请根据患者症状描述和提供的医学知识图谱，从以下疾病中选出最可能的一种，直接输出疾病名称，不要输出任何其他内容。
(You are a medical diagnosis expert. Given the patient's symptoms and the provided medical knowledge graph, identify the most likely disease from the candidate diseases. Output only the disease name.)
msgs = [
    {"role": "system", "content": SYSTEM_PROMPT},
    {"role": "user",   "content":
      f"Context:\n{sample['knowledge_summary_simple']}\n\n"
      f"Question:\n{sample['current_q']+candidate diseases}"},
]
```

Prompt for StrategyQA

```
SYSTEM_PROMPT:You are a reasoning assistant. Please answer the yes/no question based on the provided supporting facts. Output only 'Yes' or 'No', with no other content.
msgs = [
    {"role": "system", "content": SYSTEM_PROMPT},
    {"role": "user",   "content":
      f"Facts:\n{format_facts(sample['facts'])}\n\n"
      f"Question:\n{sample['question']}"},
]
```

Prompt for MMLU

```
SYSTEM_PROMPT:You are a xxx expert. Answer the multiple-choice question based on the provided reference materials and your reasoning. Output only one letter: A, B, C, or D, with no other content.
msgs = [
    {"role": "system", "content": SYSTEM_PROMPT},
    {"role": "user", "content": format_user_content(sample)},
]
```

Figure 10: Prompts used for different benchmarks. For DXY and GMD, we additionally present a prompt variant with the disease list moved from the system prompt to the Query.

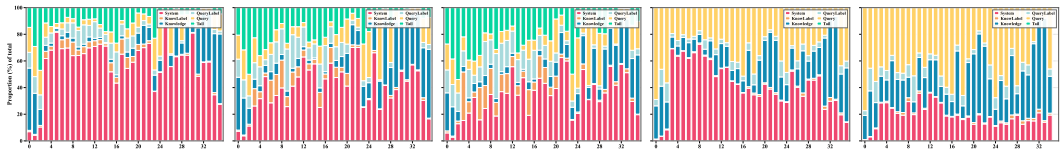


Figure 11: Model: Qwen2.5-3B; Dataset: DXY. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

A.2.2 Attention Interventions

In this section, we present comprehensive attention intervention experiments at the global, layer, and attention-head levels.

Global attention interventions. Figures 25 and 26 show the performance of LLaMA3.2-3B on GMD under global attention attenuation and amplification. For attention attenuation, we intervene on the top 100%, 30%, and 10% highest-attention tokens within a selected region, corresponding to $\lambda \in \{1.0, 0.3, 0.1\}$, with $\beta \in \{0.0, 0.5\}$. For attention amplification, we enhance the top 100%, 50%, 25%, 15%, and 10% highest-attention tokens, using $\beta \in \{1.5, 2.0\}$. As shown in the figures, aside from the widespread failures caused by large-scale attention masking, attenuating attention to the knowledge region leads to a substantially larger performance drop than attenuating other regions, whereas amplifying knowledge attention consistently improves performance on knowledge-intensive tasks.

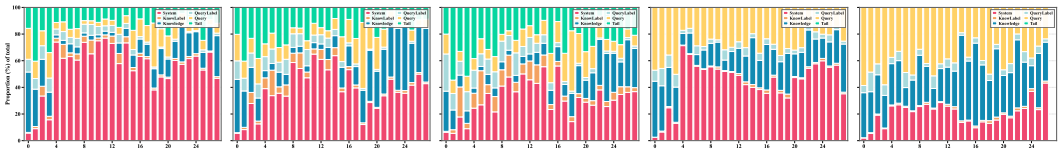


Figure 12: Model: Qwen2.5-7B; Dataset: DXY. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.



Figure 13: Model: LLaMA3.2-3B; Dataset: DXY. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

We further conduct a more extensive study of system attention attenuation by evaluating $\lambda \in \{0.1, 0.2, 0.3, 0.5, 0.75, 1.0\}$ and $\beta \in \{0.0, 0.25, 0.5, 0.75\}$, as shown in Figure 30. We then analyze the internal changes of models exhibiting substantial performance gains or degradations after intervention, yielding consistent conclusions:

Observation. Nearly all performance gains coincide with increased attention to the knowledge region, especially in the mid-to-late Transformer layers. This underscores that reducing system attention does not directly improve performance; instead, it frees attention budget for the knowledge region, enhancing knowledge utilization. Performance degradation, however, is often accompanied by excessive attention to $\mathcal{I}_{know_label}$. For example, masking all system tokens (sys_all_mask) increases $\mathcal{I}_{know_label}$ attention by more than 100×. We attribute this to the loss of architectural attention sinks in $\mathcal{I}_{sys}$, which forces $\mathcal{I}_{know_label}$ to absorb the structural biases inherent to softmax normalization, ultimately disrupting internal Transformer dynamics.

Layer-wise Attention Intervention. Figures 27 and 28 present the performance changes of Qwen1.5-1.8B on DXY and GMD, respectively, under layer-wise attention attenuation across different functional input regions. Figure 29 shows the corresponding results for layer-wise attention amplification on GMD. As shown in the figures, attenuating knowledge attention in the mid-to-late layers causes performance degradation more frequently than attenuating other regions. Conversely, amplifying knowledge attention in these layers consistently improves model performance.

Head-wise interventions. Figures 31 and 32 present the performance of Qwen1.5-1.8B on DXY under head-wise masking of system and knowledge attention, respectively. Figures 33 and 34 show the corresponding results when system and knowledge attention are amplified by 1.5× on a per-head basis. Comparing these results reveals that, in the mid-to-late layers (e.g., after Layer 10), model performance becomes increasingly aligned with changes in knowledge attention, providing preliminary evidence that more attention heads in these layers are involved in knowledge focus and utilization.

A.3 Knowledge-Aware Attention Steering and Redistribution

We rank all layers by their saliency-to-attention ratios and select the top $\lceil 0.5L \rceil$ layers for redistribution. The selection budget is motivated by the layer-wise imbalance profile established in Section 3.2. As Figure 2 shows, the Attention–Saliency Imbalance becomes particularly pronounced in the mid-to-late Transformer layers, where Knowledge saliency rises sharply relative to its allocated

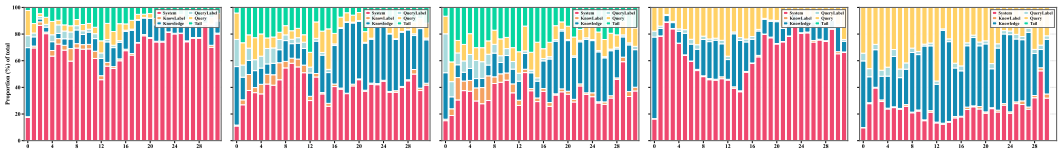


Figure 14: Model: LLaMA3.1-8B; Dataset: DXY. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

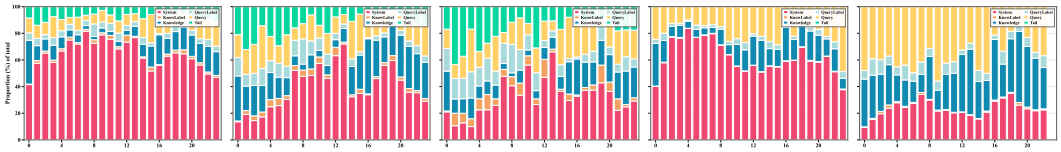


Figure 15: Model: InternLM2.5-1.8B; Dataset: DXY. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

attention. Ranking all layers by their saliency-to-attention ratio and selecting the top 50% therefore operationally targets this high-imbalance regime while excluding the early syntactic layers where the imbalance is negligible. This is consistent with the starting-layer ablation in Section 4.3, which confirms that attention redistribution is most beneficial when initiated from the intermediate layers onward.

B Experiments

B.0.1 Baselines

To comprehensively evaluate KASR, we compare it against training-free contrastive decoding and heuristic attention-intervention baselines: CAD (Shi et al., 2024) amplifies knowledge by computing a contrastive logit distribution between predictions conditioned on correct and perturbed knowledge. Both ACT (Yu et al., 2024b) and PASTA (Zhang et al., 2024) identify effective attention heads on a validation set and reuse them during test-time inference. ACT identifies sink tokens whose attention ratio exceeds $5/N$, redistributes 60% of their attention to the remaining tokens, and retains the heads yielding performance gains. PASTA reallocates attention from non-instruction tokens to instruction tokens and retains the top 1% performing heads. Following its original implementation, we adapt PASTA by treating the knowledge region as the target instruction region. **Notably, both ACT and PASTA require separately evaluating each of the $L \times H$ attention heads over the entire validation set, resulting in substantial computational overhead.**

B.1 Main Analysis

B.1.1 Comparison Results

Table 5 presents the complete comparison of KASR and the baselines across 10 LLMs and 5 benchmarks. KASR consistently outperforms existing training-free contrastive decoding and heuristic attention redistribution methods, demonstrating its effectiveness and strong generalizability.

Efficiency Analysis. As shown in Table 6, KASR is more efficient than CAD. CAD requires two forward passes (conditioned on gold and distractor contexts) to estimate contrastive distributions. In contrast, KASR performs attention redistribution directly within the self-attention computation, introducing no additional inference. Consequently, CAD incurs higher token and time costs. On

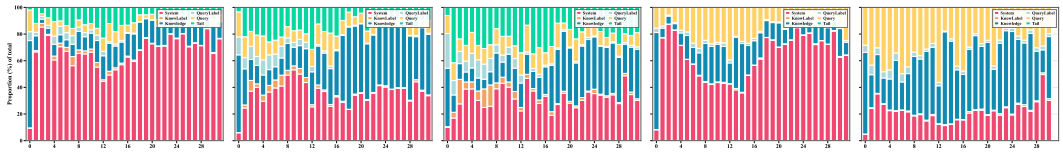


Figure 16: Model: LLaMA3.1-8B; Dataset: GMD. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

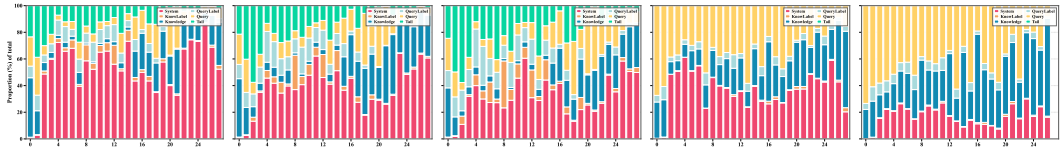


Figure 17: Model: Qwen2.5-1.5B; Dataset: GMD. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

datasets with relatively short knowledge contexts, CAD consumes approximately $1.5\times$ more tokens and $2\times$ longer inference time than KASR. These results validate that KASR achieves performance gains while maintaining superior inference efficiency.

B.2 Further Analysis

B.2.1 Connection between Massive Activation and Attention Redundancy

Prior studies have reported the existence of massive activations in specific hidden dimensions of LLMs and attributed them to biases accumulated during pre-training (Sun et al., 2024). In this section, we analyze the inner connection between this massive activation and the attention redundancy. Specifically, we profile the tokens within the top-10 of both attention weights and hidden-state activations on the GMD dataset, as summarized in Table 7. We can observe:(1) **Several system tokens exhibit severe massive activations**, particularly initial markers such as '<|im_start|>', whose activation values in specific hidden dimensions are more than $1000\times$ those of ordinary tokens, whereas activations of knowledge tokens remain comparatively stable. (2) **Massive activation indeed affect attention allocation.** The outlier magnitudes scale the input representation $\mathbf{X}$ of Transformer blocks, thereby influencing attention computation and propagating their effects through residual connections across layers. This provides a plausible explanation for the persistent attention bias toward the system region. (3) **Corroborating prior findings that LLMs tend to assign excessive attention to uninformative punctuation or special tokens, we observe that this behavior is particularly prevalent in the system region.** Redirecting their redundant attention to semantically informative regions, such as the knowledge region, therefore stands as a mechanistically sound strategy for improving knowledge utilization.

B.2.2 Head-Level Attention Redistribution Analysis

Although our previous analyses confirm the benefits of enhancing knowledge attention in the middle-to-late Transformer layers, Transformer layers are not the finest granularity for attention intervention. We therefore further investigate head-level attention redistribution. Specifically, we record the attention heads whose knowledge's attention weights increase under the intervention operations described in Section 3.2.2, and combine these results with the layer-selection to identify the heads for attention redistribution. The results are reported in Table 8. We observe that head-level redistribution further improves performance over layer-level KASR, suggesting that attention redun-

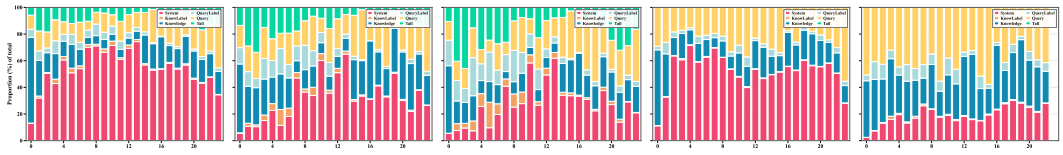


Figure 18: Model: InternLM2-1.8B; Dataset: GMD. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.



Figure 19: Model: LLaMA3.2-3B; Dataset: MMLU-LAW. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

dancy may be concentrated in specific heads rather than being uniformly distributed across an entire layer. These findings highlight the potential of fine-grained attention analysis and redistribution, and provide promising directions for understanding the functional specialization of individual attention heads and designing more precise attention intervention strategies.

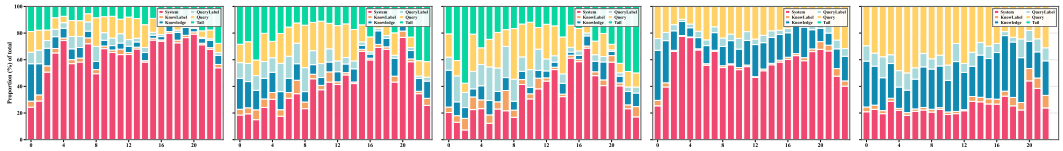


Figure 20: Model: Qwen1.5-1.8B; Dataset: StrategyQA. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

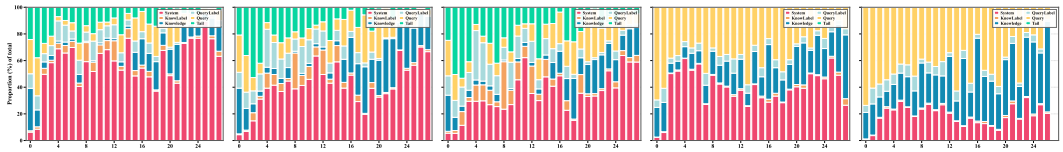


Figure 21: Model: Qwen2.5-1.5B. Dataset: GMD. Knowledge format: Text. Knowledge relevance: Relevant knowledge. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

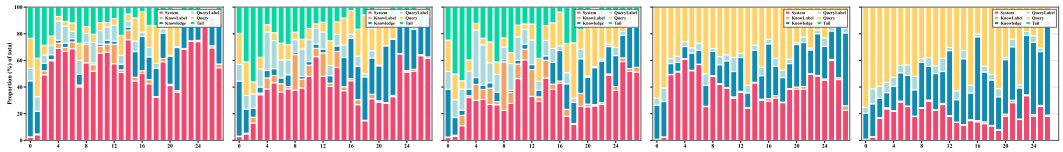


Figure 22: Model:Qwen2.5-1.5B. Dataset:GMD. Knowledge format: Text. Knowledge relevance: No task-relevant knowledge. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.



Figure 23: Model: Qwen1.5-1.8B. Dataset: DXY. Knowledge format: Triplets. Knowledge relevance: Relevant knowledge. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

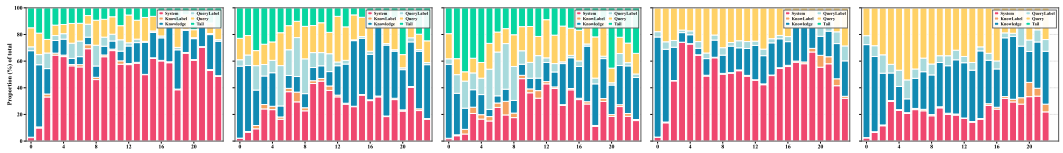


Figure 24: Model: Qwen1.5-1.8B. Dataset: DXY. Knowledge format: Triplets. Knowledge relevance: No task-relevant knowledge. (a) Attention allocation from the target token across functional input regions. (b) Saliency-based contribution of each region to the final prediction. (c) Regional contributions to the residual stream updates. (d) Attention distribution from the query region across input regions. (e) Information saliency propagated from each region to the query region.

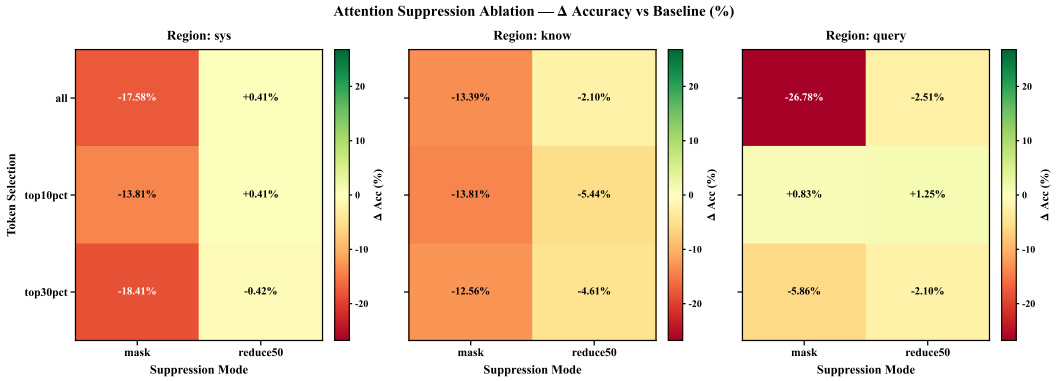


Figure 25: Model: LLaMA3.2-3B. Dataset: GMD. Model performance under global regional attenuation.

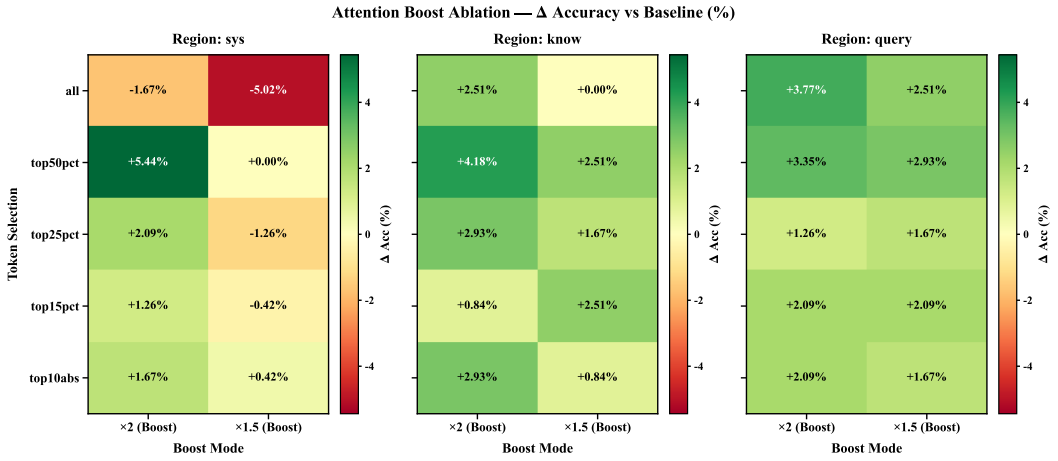


Figure 26: Model: LLaMA3.2-3B. Dataset: GMD. Model performance under global regional amplification.

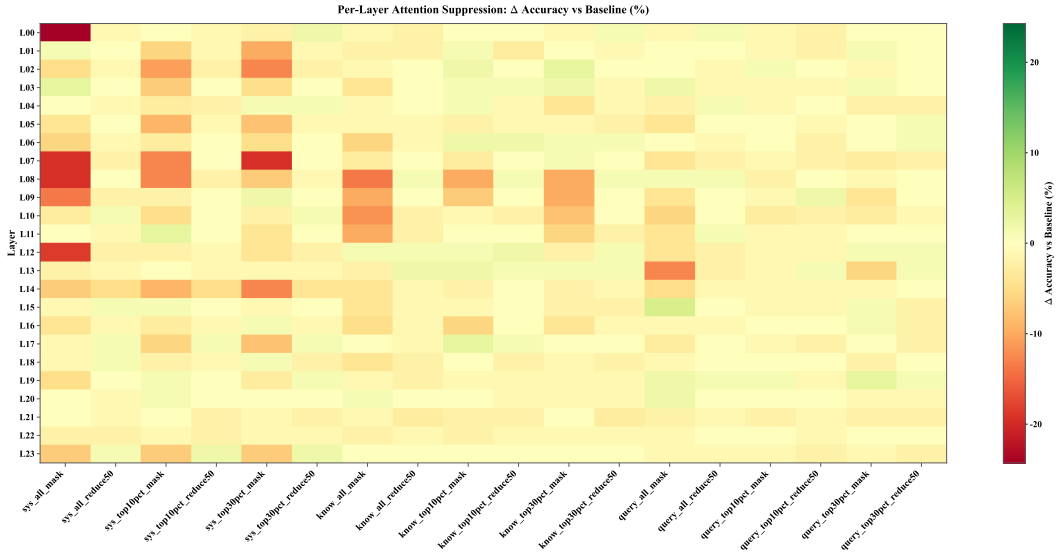


Figure 27: Model: Qwen1.5-1.8B. Dataset: DXY. Heatmap of model performance under layer-wise attention attenuation across different functional input regions. Red indicates performance degradation relative to the baseline, while green indicates performance improvement. The x-axis denotes the intervention setting (e.g., sys_all_reduce50: reduce the attention of all system tokens by 50%), and the y-axis denotes the Transformer layer.



Figure 28: Model: Qwen1.5-1.8B. Dataset: GMD. Heatmap of model performance under layer-wise attention attenuation across different functional input regions.

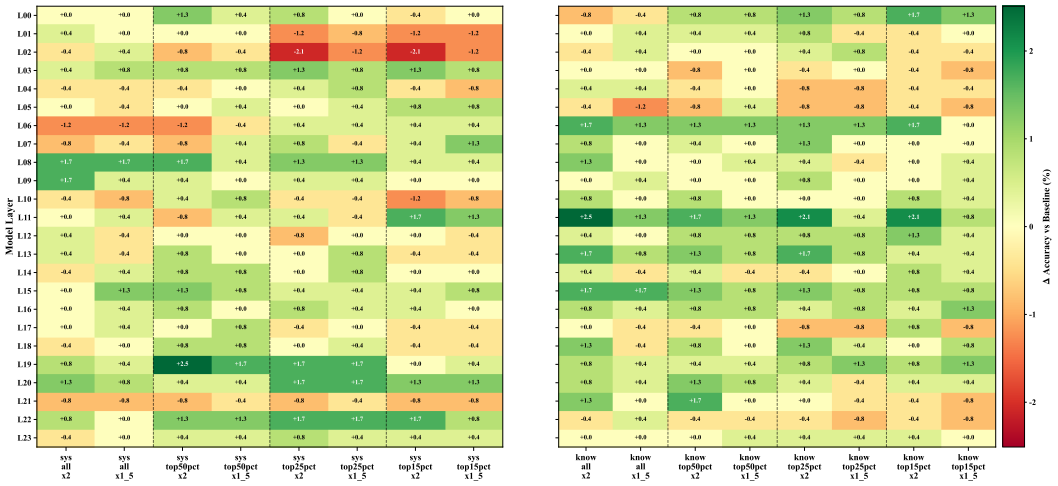


Figure 29: Model: Qwen1.5-1.8B. Dataset: GMD. Heatmap of model performance under layer-wise attention amplification across different functional input regions.

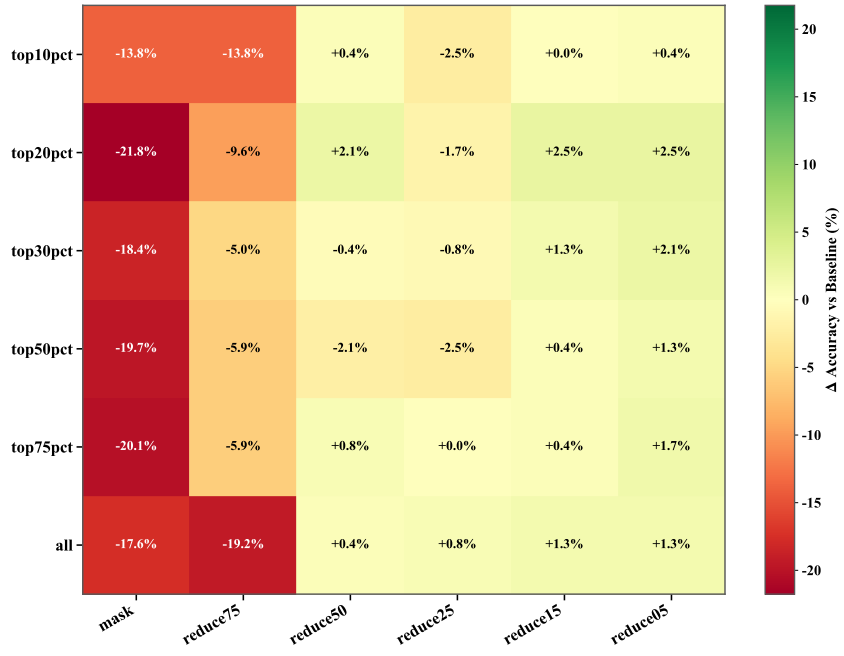


Figure 30: Model: LLaMA3.2-3B. Dataset: GMD. Model performance under global system attention attenuation.

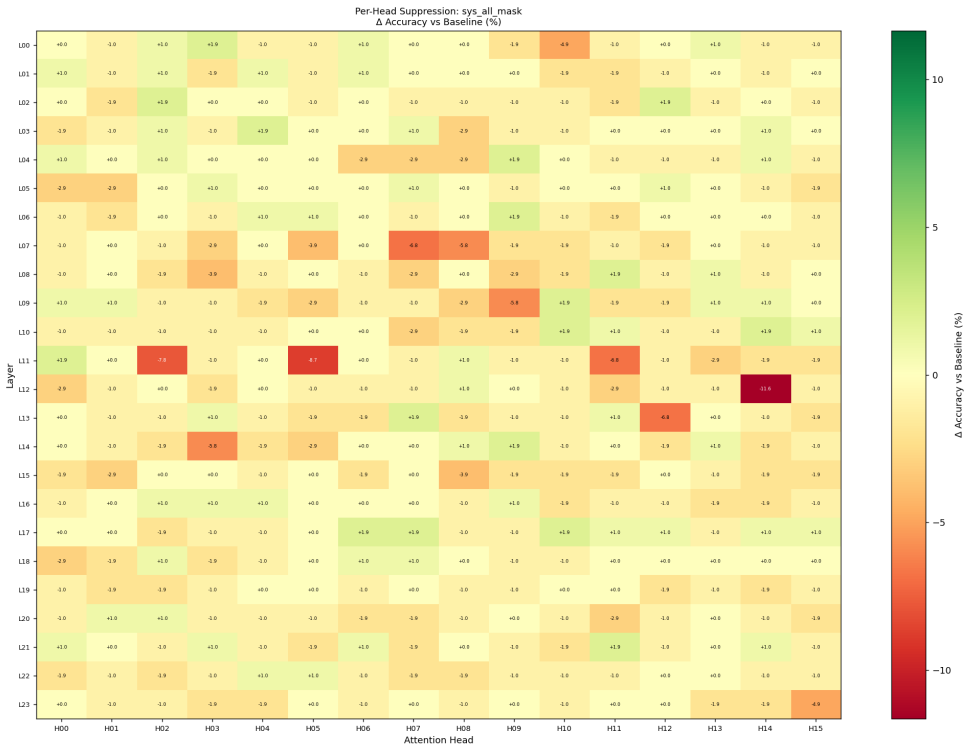


Figure 31: Model: Qwen1.5-1.8B. Dataset: DXY. Performance under head-wise masking of system attention.

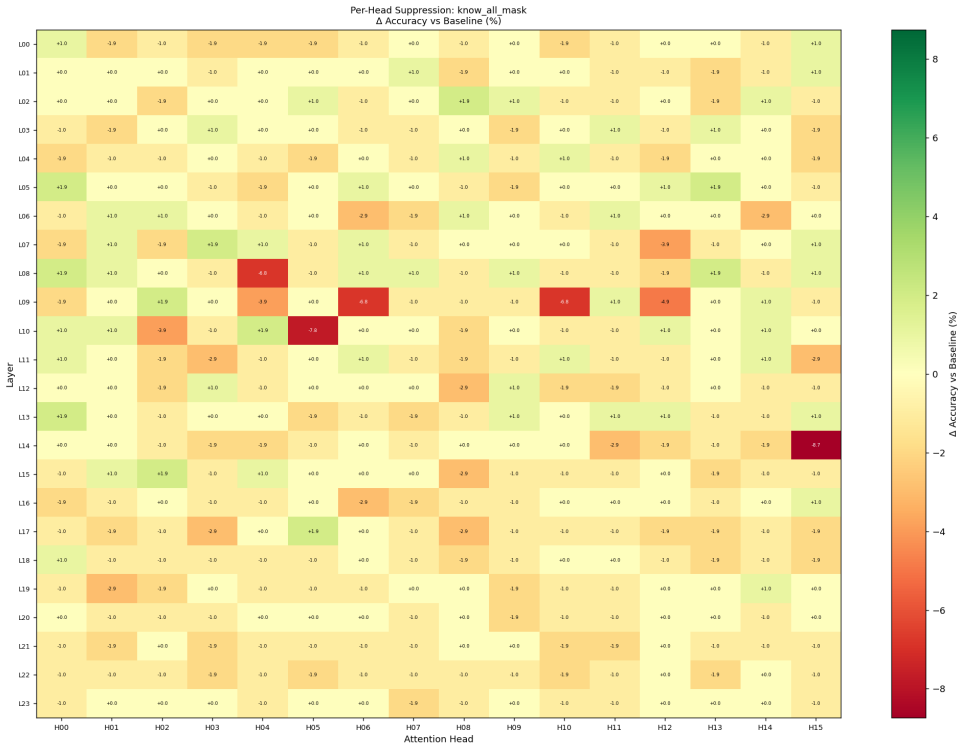


Figure 32: Model: Qwen1.5-1.8B. Dataset: DXY. Performance under head-wise masking of knowledge attention.

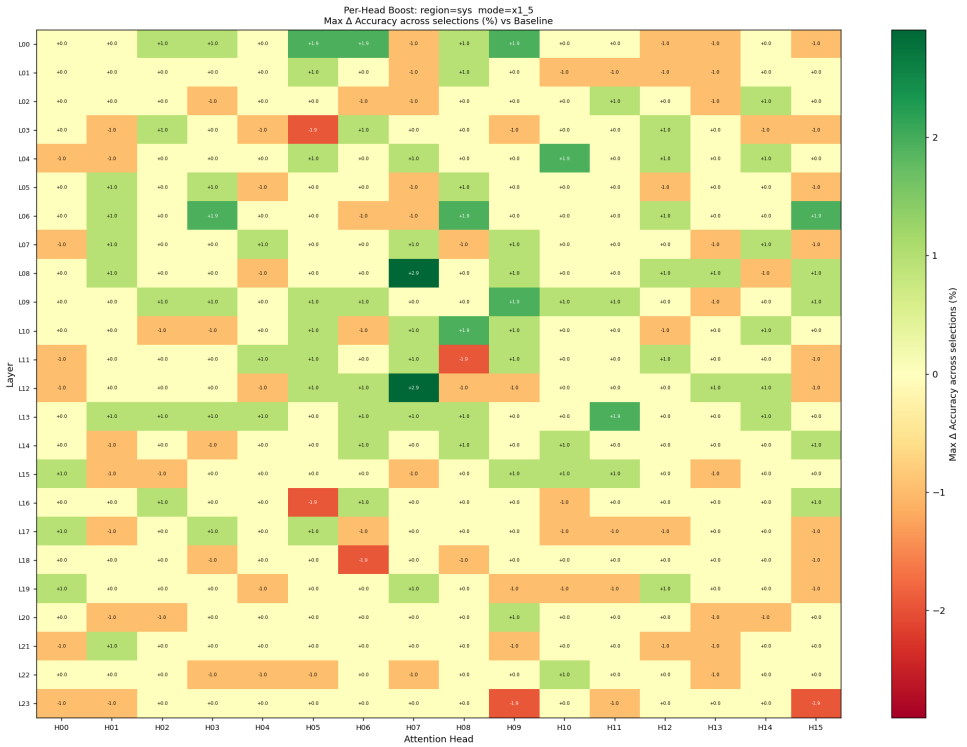


Figure 33: Model: Qwen1.5-1.8B. Dataset: DXY. Performance under head-wise 1.5× amplification of system attention.

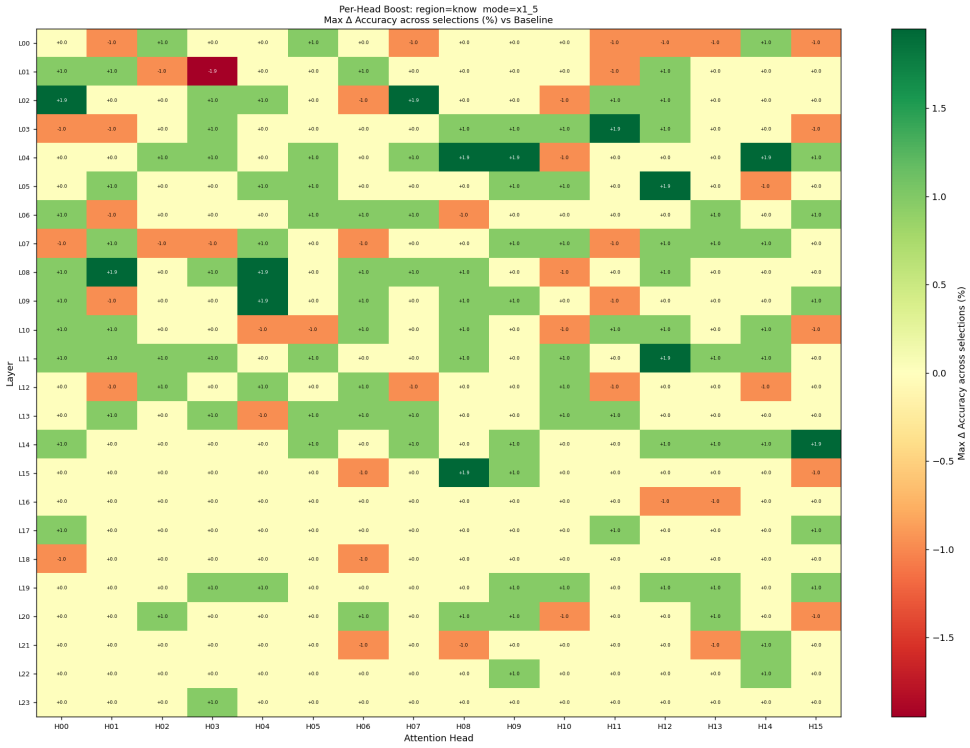


Figure 34: Model: Qwen1.5-1.8B. Dataset: DXY. Performance under head-wise 1.5× amplification of knowledge attention.

Table 5: Comparison between KASR and baselines on five benchmarks.

Backbone	Method	DXY	GMD	StrategyQA	MMLU-LAW	MMLU-MED
InternLM2.5-1.8B	Base	44.66	45.61	67.47	30.59	54.84
	CAD	**48.54**	35.98	70.96	-	-
	ACT	34.95	26.78	57.21	28.24	16.13
	PASTA	**48.54**	8.37	70.52	29.41	54.84
	KASR(Ours)	46.6	**58.58**	**71.40**	**33.53**	**61.29**
InternLM2-1.8B	Base	46.60	67.36	74.02	30.00	61.29
	CAD	48.54	56.49	**76.20**	-	-
	ACT	19.42	8.37	49.34	28.24	48.39
	PASTA	47.57	68.2	74.89	29.41	48.39
	KASR(Ours)	**52.43**	**71.13**	75.76	**34.12**	**64.52**
Llama3.1-8B	Base	50.49	61.09	89.30	48.82	77.42
	CAD	50.49	59.41	90.39	-	-
	ACT	50.49	7.11	46.72	22.35	51.61
	PASTA	50.49	60.67	87.77	53.53	77.42
	KASR(Ours)	**59.22**	**67.78**	**90.61**	**54.71**	**90.32**
Llama3.2-3B	Base	30.10	35.15	78.82	41.76	74.19
	CAD	30.10	36.82	81.22	-	-
	ACT	31.07	7.11	46.29	41.76	19.35
	PASTA	31.07	46.86	76.64	40.59	80.65
	KASR(Ours)	**42.72**	**48.95**	**81.66**	**44.71**	**83.87**
Qwen1.5-1.8B	Base	51.46	59.41	66.38	30.00	54.84
	CAD	52.43	59.41	68.78	-	-
	ACT	31.07	43.51	53.49	32.35	29.08
	PASTA	48.54	66.53	57.42	32.94	54.84
	KASR(Ours)	**58.25**	**69.87**	**71.40**	**34.12**	**58.06**
Qwen1.5-4B	Base	55.34	61.92	78.38	35.29	51.61
	CAD	55.34	61.92	80.57	-	-
	ACT	19.42	23.01	67.90	30.00	38.71
	PASTA	57.28	62.76	**80.79**	35.29	51.61
	KASR(Ours)	**62.14**	**70.71**	80.13	**39.41**	**64.52**
Qwen2-7B	Base	55.34	65.27	87.55	51.76	67.74
	CAD	51.46	66.11	88.21	-	-
	ACT	52.43	**76.15**	72.27	44.12	48.39
	PASTA	56.31	72.38	87.77	51.76	67.74
	KASR(Ours)	**61.17**	74.9	**88.86**	**54.71**	**70.97**
Qwen2.5-1.5B	Base	44.66	65.79	75.98	32.94	54.84
	CAD	40.78	66.44	78.60	-	-
	ACT	21.36	43.1	47.38	30.00	35.48
	PASTA	49.51	67.71	78.60	35.88	54.84
	KASR(Ours)	**54.37**	**68.2**	**79.91**	**41.18**	**61.29**
Qwen2.5-3B	Base	41.75	62.34	73.80	41.76	77.42
	CAD	42.72	62.76	77.73	-	-
	ACT	18.45	30.96	49.13	25.29	67.74
	PASTA	47.57	64.02	73.58	42.35	74.19
	KASR(Ours)	**48.54**	**73.22**	**79.26**	**47.65**	**80.65**
Qwen2.5-7B	Base	49.51	58.16	86.24	52.94	74.19
	CAD	49.51	57.74	**90.61**	-	-
	ACT	26.21	27.62	89.30	41.18	61.29
	PASTA	52.43	65.17	88.43	53.53	74.19
	KASR(Ours)	**58.25**	**68.62**	87.99	**55.88**	**77.42**

Table 6: Efficiency comparison between CAD and KASR. # Avg. Tokens and # Avg. Time denote the average token consumption and time latency, respectively.

Dataset	Method	# Avg. Tokens	#Avg. Time(ms)
DXY	CAD	625.8	114.0
	KASR	535.9	76.5
GMD	CAD	1714.6	193.3
	KASR	1601.3	153.8
StrategyQA	CAD	166.2	72.9
	KASR	106.9	37.3

Table 7: Layer-averaged statistics of attention allocation and hidden-state activations within $\mathcal{I}_{sys}$ and $\mathcal{I}_{know}$ for Qwen2.5-1.5B. # Avg. Attn and # Max denote the average token attention and the maximum hidden-layer activation value per token, respectively.

$\mathcal{I}_{\text{sys}}$				$\mathcal{I}_{\text{know}}$			
Activation Top10	# Max.	Attention Top10	# Avg. Attn	Activation Top10	# Max.	Attention Top10	# Avg. Attn
<|im_start|>	5776.74	<|im_start|>	0.3199	:	44.74	coronary	0.0024
system	43.36	.\n	0.0300	including	43.70	breast	0.0023
Middle	42.78	\n	0.0076	-itis	42.36	asthma	0.0022
;	45.50	asthma	0.0042	disease	41.99	pneumonia	0.0020
patient	42.13	.	0.0041	asthma	41.93	esophag-	0.0020
symptoms	42.10	esophag-	0.0037	-itis	44.29	.	0.0018
and	42.01	name	0.0034	esophag-	41.24	traumatic	0.0016
diagnosis	41.66	pneumonia	0.0033	trauma	41.24	enter-	0.0016
medical	41.12	,	0.0030	pneumonia	40.03	thyroid	0.0013
type	41.10	coronary	0.0030	coronary	39.83	nodules	0.0011

Table 8: Performance of head-level KASR redistribution guided by layer-wise constraints and positively contributing heads filtered from intervention experiments.

Backbone	ACC. (%)
Qwen1.5-1.8B	61.17
Qwen1.5-4B	59.22
Qwen2-7B	61.17
Qwen2.5-1.5B	58.25
Qwen2.5-3B	54.37
Qwen2.5-7B	58.25
LLaMA3.2-3B	51.46