When LLMs Know but Fail to Reason: Injecting Memory for Reasoning Enhancement

Tian Wang[1,*] **Shiyu Hu**[1,*] **Ruiyang Qin**[1] **Hanyue Zhang**[1] **Qingzhuo Wang**[1]
Shuhan Yu[1] **Dongrui Liu**[2] **Zhihua Wei**[1] **Wen Shen**[1,†]

[1] School of Computer Science and Technology, Tongji University, Shanghai, China
[2] Shanghai Artificial Intelligence Laboratory, Shanghai, China

Abstract

Large language models (LLMs) have demonstrated strong capabilities on complex reasoning tasks. However, recent studies [Gekhman et al., 2026, Song et al., 2026, Jin et al., 2025, Cheng et al., 2024] suggest that when handling knowledge-intensive reasoning tasks, LLMs often fail to effectively utilize the knowledge acquired during pre-training, which limits their reasoning performance. To investigate how internal knowledge (also termed memory) is used by an LLM when dealing with a reasoning query, we propose the Memory-to-Reasoning Alignment (MRA) metric to measure the memory engagement level of an LLM on the reasoning query. We empirically verify that failures in reasoning may correspond to insufficient memory engagement. Inspired by this, we propose a training-free Memory-Guided Activation Injection (MGAI) method to increase the memory engagement level of an LLM, in order to improve the reasoning capability of an LLM. Specifically, given a query, MGAI generates an intervention vector and applies it to the reasoning representation to inject the corresponding memory information into the reasoning representation of the LLM. Furthermore, we propose an adaptive intervention strategy to control which queries require intervention and the magnitude of the intervention. Experiments show that our method consistently improves performance across multiple reasoning benchmarks while largely preserving the memory and general capabilities of LLMs.

1 Introduction

Large language models (LLMs) have demonstrated remarkable capabilities on a wide range of complex natural language tasks which require both extensive knowledge and strong reasoning abilities [Jin et al., 2025, Wang et al., 2024, Sheng et al., 2025]. Although LLMs have memorized much factual knowledge in their parameters through pre-training on massive corpora [Mahdavi et al., 2023, Kiyomaru et al., 2024, Zhu et al., 2024], recent studies show that, as illustrated in Figure 1 (top), LLMs often struggle to effectively utilize the knowledge when dealing with complex reasoning queries, which limits their reasoning performance [Gekhman et al., 2026, Song et al., 2026, Jin et al., 2025, Cheng et al., 2024].

To address this issue, some studies leverage Retrieval-Augmented Generation (RAG) [Lewis et al., 2020] or explicitly include structured knowledge in the prompt [Tao et al., 2025, Ko et al., 2024] to inject external factual knowledge directly during reasoning, but they do not directly improve the LLM's ability to utilize internal knowledge. While other studies use Supervised Fine-Tuning

*Tian Wang and Shiyu Hu contributed equally to this work and agree that the order of their names may be exchanged as useful to highlight their contributions in individual professional pursuits.
†Corresponding author. Email: wenshen@tongji.edu.cn.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

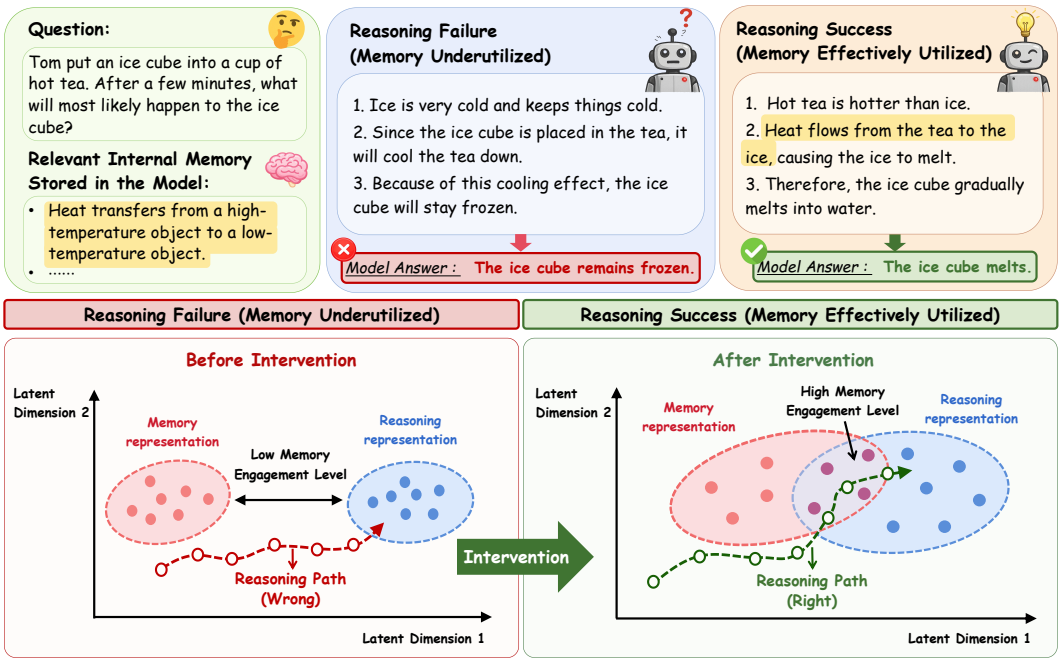


Figure 1: Illustration of memory engagement in LLM reasoning. The top panel presents a reasoning query where the model fails when relevant internal memory is underutilized, but succeeds when the stored knowledge is effectively leveraged. The bottom panel provides a latent-space illustration. The left subfigure shows that reasoning failure may correspond to low memory engagement. In this paper, we propose a training-free method to intervene on the reasoning process to increase the memory engagement level. The reasoning path is only used to schematically indicate whether memory is utilized during reasoning, rather than representing an actual latent trajectory.

(SFT) [Jin et al., 2025] or reinforcement learning (RL) [Chen et al., 2025] to directly improve LLMs' ability to utilize internal memory. However, such training-based methods require additional data annotated with memory-related and reasoning-related signals and introduce extra computational cost.

In this work, as Figure 1 (bottom) shows, we propose a training-free method to intervene in the internal representations of an LLM, aiming to make more effective use of relevant internal memory when dealing with reasoning queries. Specifically, we first propose the **Memory-to-Reasoning Alignment (MRA)** metric to measure the memory engagement level of an LLM. To this end, we identify a sparse set of key neurons most related to the memory task and the reasoning task, respectively, and extract memory representations and reasoning representations based on these key neurons. Then, the MRA metric is defined as the alignment between memory representations and reasoning representations. By comparing MRA scores of correctly and incorrectly answered reasoning queries, we find that failures in reasoning may correspond to insufficient memory engagement, suggesting that relevant internal knowledge is sometimes underutilized during reasoning.

Inspired by the above observations, we propose **Memory-Guided Activation Injection (MGAI)**, a training-free method to enhance the memory engagement level of an LLM during reasoning. MGAI consists of an offline stage and an online stage. Specifically, in the offline stage, we learn a function to map the memory representation to the reasoning representation from correctly answered reasoning queries. Then in the online stage, *i.e.*, during inference on a certain query, we apply the mapping function to the memory representation of the query to obtain an intervention vector. With the intervention vector, we can inject the corresponding memory information into the reasoning representation to guide the model to make more effective use of internal memory.

Furthermore, to achieve more efficient interventions, we develop an adaptive intervention mechanism to determine which samples require intervention and the magnitude of the intervention. Specifically, we train a detection probe based on the extracted memory representations and reasoning representations, so as to determine whether a certain query requires intervention. We then use the confidence scores produced by the detection probe to determine the intervention magnitude for each sample. Experiments show that our method consistently improves performance across five models on multiple

complex reasoning benchmarks while largely preserving performance on benchmarks related to general and memory capabilities of the LLM.

2 Related Work

Enhancing reasoning via knowledge utilization. Large language models (LLMs) have demonstrated strong capabilities in both knowledge acquisition and reasoning [Jin et al., 2025, Wang et al., 2024, Sheng et al., 2025]. Despite memorizing substantial factual knowledge during pre-training [Mahdavi et al., 2023, Kiyomaru et al., 2024, Zhu et al., 2024], recent studies reveal that LLMs often fail to effectively utilize this knowledge when solving complex reasoning tasks [Gekhman et al., 2026, Song et al., 2026, Jin et al., 2025, Cheng et al., 2024]. To address this issue, prior work has explored augmenting LLMs with external knowledge sources. Retrieval-Augmented Generation (RAG) [Li et al., 2025b, Zhou et al., 2025] retrieves relevant information at inference time, while other approaches incorporate structured knowledge directly into prompts [Tao et al., 2025, Ko et al., 2024]. Although these methods can enrich the information available during reasoning, they primarily rely on external knowledge and do not directly improve how LLMs leverage their internal memory. In contrast, our method explicitly enhances the alignment between internal memory and reasoning processes, thereby directly improving reasoning capability without relying on external knowledge sources.

Enhancing reasoning via model adaptation. Another line of research improves reasoning performance by adapting the model itself. Training-based approaches, such as supervised fine-tuning (SFT) [Jin et al., 2025] and reinforcement learning (RL) [Chen et al., 2025, Huang and Wan, 2026, Li et al., 2025a, Liu et al., 2025], aim to enhance the model's ability to integrate memory and reasoning through additional supervision, but typically incur extra computational cost. In parallel, training-free methods have been explored to improve reasoning without modifying model parameters, including approaches at both the decoding level [Yang et al., 2025c] and the representation level [Li et al., 2026, Zhuang et al., 2026, Zhao et al., 2025]. While these methods demonstrate effectiveness in improving reasoning performance or efficiency, they generally do not explicitly model or strengthen the interaction between internal memory and reasoning processes. Our method instead directly targets this interaction by aligning memory and reasoning representations and injecting memory-relevant signals during inference, leading to more principled improvements in reasoning performance.

3 Method

3.1 Analysis of Memory Engagement during the Reasoning of an LLM

Characterizing the memory and reasoning representations. To investigate how the LLM uses its internal memory when dealing with a reasoning query, we first characterize the memory and reasoning representation spaces by locating intermediate-layer neurons that are highly related to the memory process and the reasoning process, respectively. Given a memory dataset $\mathcal{D}_{\text{mem}}$ and a reasoning dataset $\mathcal{D}_{\text{reason}}$, we adopt the gradient-based importance score [Michel et al., 2019, Wang et al., 2023] to estimate the contribution of each neuron to the memory task and the reasoning task. We determine *those neurons that contribute substantially to memory and reasoning tasks as memory-related neurons and reasoning-related neurons, respectively*.

Specifically, for a target module u of layer l with d neurons (*e.g.*, a linear transformation module within the multi-head attention or the multi-layer perceptron) in an LLM, let $\mathbf{h}_{\text{in}}^{(l,u)} \in \mathbb{R}^{d_{\text{in}}}$ denote the input activation vector of the module, $W^{(l,u)} \in \mathbb{R}^{d \times d_{\text{in}}}$ denote the weight matrix of the module, and $\mathbf{h}^{(l,u)} = W^{(l,u)}\mathbf{h}_{\text{in}}^{(l,u)} \in \mathbb{R}^{d}$ denote the output activation vector of the module. The average importance score of the i-th neuron in module u of layer l during inference on all samples $x \in \mathcal{D}$ is computed as follows:

$$s_{\mathcal{D}}^{(l,u)}(i) = \sum_{j=1}^{d_{\text{in}}} I_{\mathcal{D}}^{(l,u)}(i,j), \qquad \text{s.t.} \quad I_{\mathcal{D}}^{(l,u)} = \mathbb{E}_{x\sim\mathcal{D}}\Big[\big|W^{(l,u)} \odot \nabla_{W^{(l,u)}}\mathcal{L}(x)\big|\Big], \tag{1}$$

where $\odot$ denotes element-wise multiplication, and $\mathcal{L}(x)$ denotes the loss computed from the model prediction at the final token of sample x. The corresponding gradients $\nabla_{W^{(l,u)}}\mathcal{L}(x)$ are computed with respect to this last-token loss only. To make importance scores $s_{\mathcal{D}}^{(l,u)}$ comparable across different

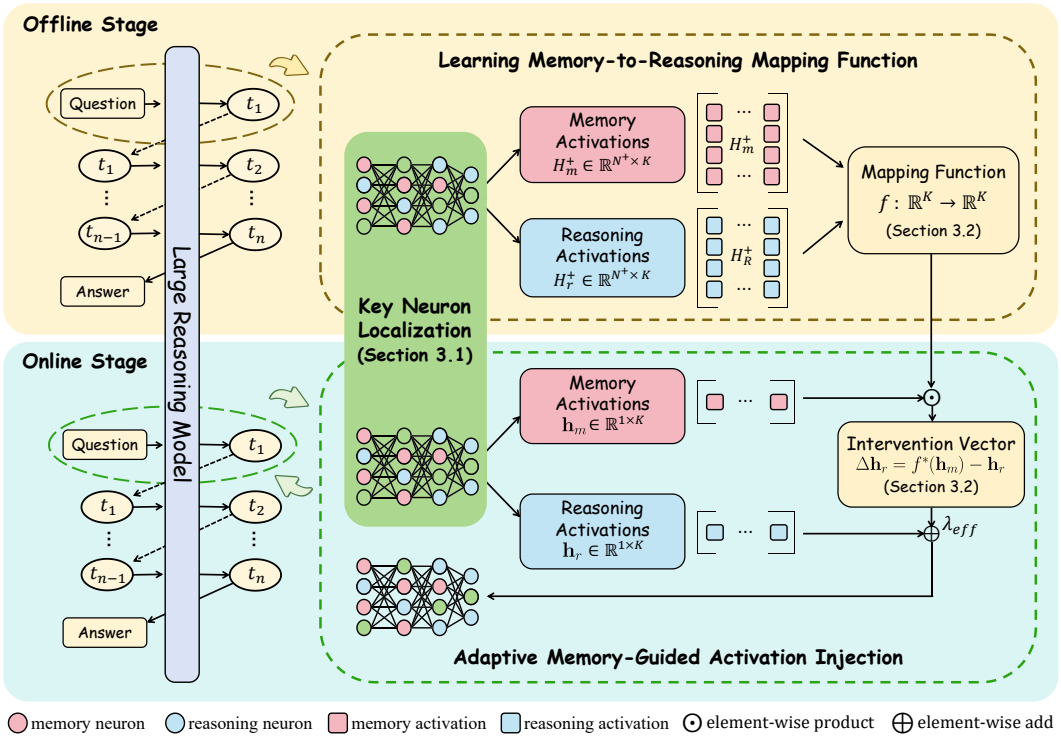


Figure 2: Overview of the proposed MGAI framework. In the offline stage, MGAI identifies key memory-related neurons and reasoning-related neurons and learns a mapping function that captures memory-to-reasoning alignment. In the online stage, MGAI generates a query-specific intervention vector and adaptively injects memory information into reasoning activations to enhance memory engagement, thereby improving the reasoning capability of the LLM.

layers and modules, we further apply min-max normalization within each module, *i.e.*, $\widetilde{s}_{\mathcal{D}}^{(l,u)}(i) = \frac{s_{\mathcal{D}}^{(l,u)}(i) - \min_{i'} s_{\mathcal{D}}^{(l,u)}(i')}{\max_{i'} s_{\mathcal{D}}^{(l,u)}(i') - \min_{i'} s_{\mathcal{D}}^{(l,u)}(i')}$.

We then compute the normalized neuron-level importance scores on the memory dataset $\mathcal{D}_{\text{mem}}$ and the reasoning dataset $\mathcal{D}_{\text{reason}}$, respectively. By globally ranking neurons across all modules u and all layers l in the LLM, we select the Top-K neurons under $\mathcal{D}_{\text{mem}}$ as memory-related neurons, and the Top-K neurons under $\mathcal{D}_{\text{reason}}$ as reasoning-related neurons:

$$\mathcal{N}_{\text{mem}}^{(K)} = \text{TopK}\left(\left\{\widetilde{s}_{\mathcal{D}_{\text{mem}}}^{(l,u)}(i)\right\}_{l,u,i}\right), \quad \mathcal{N}_{\text{reason}}^{(K)} = \text{TopK}\left(\left\{\widetilde{s}_{\mathcal{D}_{\text{reason}}}^{(l,u)}(i)\right\}_{l,u,i}\right). \tag{2}$$

In this way, we characterize the memory representation and the reasoning representation encoded by the LLM based on the identified neuron sets $\mathcal{N}_{\text{mem}}^{(K)}$ and $\mathcal{N}_{\text{reason}}^{(K)}$. Specifically, given an input query x, the memory representation $\mathbf{h}_m(x) \in \mathbb{R}^{1\times K}$ and the reasoning representation $\mathbf{h}_r(x) \in \mathbb{R}^{1\times K}$ of an LLM are denoted as follows:

$$\mathbf{h}_m(x) = \left(\text{Concat}\left[h_i^{(l,u)}(x)\right]_{(l,u,i)\in\mathcal{N}_{\text{mem}}^{(K)}}\right)^\top, \quad \mathbf{h}_r(x) = \left(\text{Concat}\left[h_i^{(l,u)}(x)\right]_{(l,u,i)\in\mathcal{N}_{\text{reason}}^{(K)}}\right)^\top, \tag{3}$$

where $h_i^{(l,u)}(x) \in \mathbb{R}$ denotes the activation value of the i-th neuron in module u of layer l, computed based on the representation at the final token position of the input query x, and the concatenation follows the lexicographic order of (l, u, i) in the LLM architecture.

Memory-to-Reasoning Alignment. Then, we define the Memory-to-Reasoning Alignment (MRA) metric as the similarity between memory representations and reasoning representations encoded by the LLM. Given a query set $\mathcal{Q} = \{x_j\}_{j=1}^N$, the MRA metric is defined as the Singular Vector Canonical Correlation Analysis (SVCCA) [Raghu et al., 2017] between the memory representations $H_m(\mathcal{Q}) = \text{stack}\left(\mathbf{h}_m(x_1), \ldots, \mathbf{h}_m(x_N)\right) \in \mathbb{R}^{N\times K}$ and the reasoning representations $H_r(\mathcal{Q}) =$

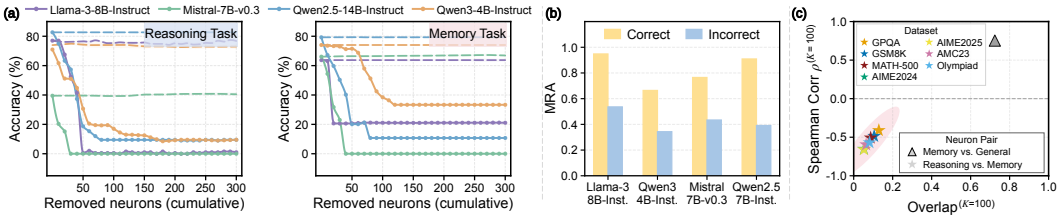


Figure 3: (a) Pruning performance on reasoning and memory tasks. Solid lines denote removing neurons in descending order of their importance scores, while dashed lines denote random pruning. (b) MRA comparison in correct/incorrect queries across models. (c) Separability between memory-related and reasoning-related neurons. Results under other number of selected neurons, including K = 50, 150 and 200, in Figure 5 in Appendix B.1 show the same phenomenon.

$\text{stack}\left(\mathbf{h}_r(x_1), \ldots, \mathbf{h}_r(x_N)\right) \in \mathbb{R}^{N \times K}$, where $\text{stack}(\cdot)$ denotes stacking by rows, as follows:

$$\text{MRA}(\mathcal{Q}) = \frac{1}{c} \sum_{i=1}^{c} \text{CCA}_i\big(\text{SVD}(H_m(\mathcal{Q})), \text{SVD}(H_r(\mathcal{Q}))\big), \tag{4}$$

where $\text{SVD}(\cdot) : \mathbb{R}^K \to \mathbb{R}^c$ denotes SVD-based dimensionality reduction and CCA_i denotes the i-th canonical correlation coefficient. Details of computation of CCA are provided in Appendix A.4.

Verifying the accuracy of neuron localization. To verify whether the localized neurons are truly task-relevant, we conduct a pruning-based validation [Zhou et al., 2018] experiment on the reasoning task dataset GSM8K [Cobbe et al., 2021] and the memory task dataset CSQA [Talmor et al., 2019]. For each task on the dataset $\mathcal{D}$, we progressively suppress the top-ranked memory-related neurons in $\mathcal{N}_{\text{mem}}^{(K)}$ or reasoning-related neurons in $\mathcal{N}_{\text{reason}}^{(K)}$ according to their normalized importance scores $\widetilde{s}_{\mathcal{D}}$ and evaluate the model's performance after each pruning step. Ideally, if the neuron localization is accurate, then the LLM should exhibit a sharp performance drop. We compare this with random pruning under the same pruning budget. As shown in Figure 3(a), pruning top-ranked memory-related neurons or reasoning-related neurons leads to a substantial accuracy degradation on the corresponding benchmark. In most settings, pruning approximately the top 50 localized neurons is sufficient to cause a clear performance drop, and for memory-related neurons in Qwen3-4B-Instruct, the degradation requires removing around 100 neurons. In contrast, random pruning keeps the performance largely stable, verifying the effectiveness of our neuron localization.

Comparing memory engagement level across queries. We further compare the MRA scores between correctly and incorrectly answered reasoning queries on four models. As Figure 3(b) shows, correctly answered queries consistently obtain higher MRA scores than incorrect ones across four models. This indicates higher memory engagement level of the LLM on correctly answered queries than incorrect ones. Based on this, we propose a hypothesis that *insufficient engagement of internal memory is closely related to reasoning failures,* which serves as a basis of learning mapping from memory representations to reasoning representations on correctly answered queries.

Verifying separability of memory and reasoning representations. However, if memory representations and reasoning representations are highly coupled, then the learned mapping function may merely capture their shared components, rather than an effective mapping from memory representations to reasoning representations. Therefore, we need to verify the separability of memory and reasoning representations. To this end, we use the Spearman Correlation and the overlap ratio to measure the separability. We compute the Spearman correlation between the two ranked neuron lists:

$$\rho^{(K)} = \text{Pearson}\left(\{\text{rank}_{\downarrow}(s_{\mathcal{D}_{\text{mem}}}(i))\}_{i \in N}, \{\text{rank}_{\downarrow}(s_{\mathcal{D}_{\text{reason}}}(i))\}_{i \in N}\right), \tag{5}$$

where $N = \mathcal{N}_{\text{mem}}^{(K)} \cup \mathcal{N}_{\text{reason}}^{(K)}$, $\text{rank}_{\downarrow}(\cdot)$ denotes the descending rank induced by neuron importance scores, and Pearson computes the correlation between the two rank lists. In addition, the overlap ratio is computed as follows:

$$\text{Overlap}^{(K)} = \left|\mathcal{N}_{\text{mem}}^{(K)} \cap \mathcal{N}_{\text{reason}}^{(K)}\right| / \left|\mathcal{N}_{\text{mem}}^{(K)} \cup \mathcal{N}_{\text{reason}}^{(K)}\right|. \tag{6}$$

Figure 3(c) reports the correlation of neurons localized on 7 reasoning task datasets and 1 memory task dataset, as well as those between 1 memory task dataset and 1 general task dataset. The general-related neurons are localized on the dataset MMLU [Hendrycks et al., 2020] using the

same method. The reasoning-vs-memory neuron pairs show both low overlap ratios $\mathrm{Overlap}^{(K)}$ and negative Spearman correlations $\rho^{(K)}$, while the memory-vs-general pairs show much higher overlap and positive correlation. It suggests that reasoning-related neurons and memory-related neurons are well separated. Results under other number of selected neurons, including K = 50, 150 and 200, in Figure 5 in Appendix B.1 show the same phenomenon. Such separability of memory and reasoning representations ensures that we can learn a mapping which captures a meaningful memory-to-reasoning alignment.

3.2 Training-Free Memory-Guided Activation Injection

Inspired by the above observations, as Figure 2 shows, we propose the training-free Memory-Guided Activation Injection (MGAI) method to learn a mapping function from memory representations to reasoning representations on correctly answered queries (**see the Offline stage**), and then use the learned mapping function to guide inference-time activation injection (**see the Online stage**).

Offline stage: learning memory-to-reasoning mapping function. Given a set of correctly answered queries $\mathcal{Q}^+ = \{x_i\}_{i=1}^{N^+}$, we extract the memory representation matrix $H_m^+ \in \mathbb{R}^{N^+ \times K}$ and the reasoning representation matrix $H_r^+ \in \mathbb{R}^{N^+ \times K}$ and learn a linear mapping function $f : \mathbb{R}^K \to \mathbb{R}^K$ by minimizing the least-squares objective:

$$f^* = \arg\min_f \left\| f(H_m^+) - H_r^+ \right\|_F^2 . \tag{7}$$

We verify that the linear function f is sufficient to capture the memory-to-reasoning alignment; detailed analyses are provided in Appendix B.2.

Besides, we train a detection probe to determine which queries require stronger intervention. For a given query x, we concatenate the memory activation vector $\mathbf{h}_m(x) \in \mathbb{R}^{1 \times d_m}$ and the reasoning activation vector $\mathbf{h}_r(x) \in \mathbb{R}^{1 \times d_r}$ as the input feature representations, denoted by $\mathbf{h}(x) = [\mathbf{h}_m(x), \mathbf{h}_r(x)] \in \mathbb{R}^{1 \times (d_m + d_r)}$. Then, we train an XGBoost binary probe $g : \mathbb{R}^{d_m + d_r} \to [0, 1]$ on the dataset $\left\{ \left(\mathbf{h}(x), \tilde{y}_j \right) \right\}_{j=1}^{N}$, where $y_j = 1$ indicates the LLM answers correctly on query x_j, and $y_j = 0$ indicates the LLM fails:

$$p_{\text{fail}}(x) = g\left(\mathbf{h}(x) \right), \tag{8}$$

where $p_{\text{fail}}(x)$ predicts the failure probability of the LLM on query x. In this way, during the inference time, a larger $p_{\text{fail}}(x)$ suggests a higher probability that the LLM will answer incorrectly on query x.

Online stage: adaptive memory-guided activation injection. During online inference, given a test reasoning query x, we first extract its memory representation $\mathbf{h}_m(x)$ and reasoning representation $\mathbf{h}_r(x)$, both derived from the final token position of the input query x. We then use the learned mapping function f^* to generate a memory-guided reasoning target:

$$\hat{\mathbf{h}}_r(x) = f^*\big(\mathbf{h}_m(x)\big). \tag{9}$$

The target $\hat{\mathbf{h}}_r(x)$ represents an ideal reasoning representation where the LLM exhibits more effective memory engagement during reasoning. In this way, the difference between $\hat{\mathbf{h}}_r(x)$ and $\mathbf{h}_r(x)$, *i.e.*, $\Delta\mathbf{h}_r(x) = \hat{\mathbf{h}}_r(x) - \mathbf{h}_r(x)$ denotes the intervention direction from the current representation toward the ideal reasoning representation. Thus, to move the actual reasoning vector $\mathbf{h}_r(x)$ closer to the ideal reasoning vector $\hat{\mathbf{h}}_r(x)$, we intervene on the reasoning representation $\mathbf{h}_r$ as follows:

$$\mathbf{h}_r'(x) = \mathbf{h}_r(x) + \lambda_{\text{eff}}(x)\, \Delta\mathbf{h}_r(x), \tag{10}$$

where $\lambda_{\text{eff}}(x)$ represents the intervention strength, which is computed based on the probability $p_{\text{fail}}(x)$ that the probe predicts that query x will be answered incorrectly, as follows:

$$\lambda_{\text{eff}}(x) = \begin{cases} \lambda \cdot p_{\text{fail}}(x), & \text{if } p_{\text{fail}}(x) > \tau, \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

where τ is a threshold. We set $\lambda = 11$ and $\tau = 0.5$ in this paper.

In this way, for queries that are likely to be answered incorrectly, we inject additional relevant memory into their reasoning representation, thereby improving the reasoning performance of the LLM.

Table 1: Accuracy (%) (↑) on reasoning, memory and general tasks across multiple backbone models. The best and second-best results are highlighted in **bold** and underlined, respectively. In the "Δ vs. Vanilla" row, **red** indicates performance gains of MGAI over vanilla, while **green** indicates performance drops. Compared with existing baseline methods for improving reasoning performance, the proposed MGAI achieves the largest gains across all backbone LLMs on reasoning tasks.

	Reasoning Task							Memory Task	General Task
Method	GPQA-D	MATH-500	AIME25	AIME24	GSM8K	AMC23	Olympiad	CSQA	MMLU
LLaMA3-8B-instruct									
Vanilla [Grattafiori et al., 2024]	31.31	26.60	6.67	6.67	76.95	25.00	4.89	63.64	60.55
REBALANCE [Li et al., 2026]	33.84	26.20	3.33	3.33	80.29	30.00	5.04	63.96	61.00
DEER [Yang et al., 2025c]	32.83	28.40	6.67	3.33	77.71	27.50	5.48	65.76	59.87
MOI [Zhuang et al., 2026]	33.33	28.00	**10.00**	6.67	68.01	22.5	5.19	65.36	59.70
EELo-CoT [Zhao et al., 2025]	32.32	28.60	6.67	**10.00**	79.23	27.50	5.48	64.37	59.05
MGAI (ours)	**36.87**	**31.20**	**10.00**	**10.00**	**81.50**	**35.00**	**6.52**	**65.93**	**61.59**
Δ vs. Vanilla	+5.56	+4.60	+3.33	+3.33	+4.55	+10.00	+1.63	+2.29	+1.04
Qwen2.5-14B-instruct									
Vanilla [Yang et al., 2025b]	37.88	72.40	13.33	20.00	83.09	37.50	15.11	79.28	70.23
REBALANCE [Li et al., 2026]	38.89	74.00	16.67	20.00	84.23	42.50	16.15	79.03	71.08
DEER [Yang et al., 2025c]	39.39	71.40	13.33	16.67	84.84	40.00	15.41	78.87	69.27
MOI [Zhuang et al., 2026]	37.37	73.80	13.33	**23.33**	84.08	40.00	17.04	79.33	70.36
EELo-CoT [Zhao et al., 2025]	39.90	68.80	16.67	20.00	83.70	37.50	17.48	**79.36**	71.01
MGAI (ours)	**42.93**	**75.20**	**20.00**	**23.33**	**85.22**	**45.00**	**18.67**	79.11	**72.05**
Δ vs. Vanilla	+5.05	+2.80	+6.67	+3.33	+2.13	+7.50	+3.56	-0.17	+1.82
Qwen3-4B-instruct									
Vanilla [Yang et al., 2025a]	56.06	72.80	23.33	33.33	70.96	27.50	2.07	74.04	67.41
REBALANCE [Li et al., 2026]	61.11	72.00	20.00	33.33	73.54	30.00	4.15	74.12	66.87
DEER [Yang et al., 2025c]	59.60	74.00	16.67	26.67	72.25	30.00	3.41	73.87	67.63
MOI [Zhuang et al., 2026]	62.12	73.80	23.33	30.00	73.01	32.50	3.55	72.89	67.20
EELo-CoT [Zhao et al., 2025]	60.60	73.40	20.00	26.67	71.04	32.50	4.30	72.40	67.10
MGAI (ours)	**66.16**	**75.60**	**26.67**	**36.67**	**79.98**	**35.00**	**4.89**	**74.45**	**67.99**
Δ vs. Vanilla	+10.10	+2.80	+3.34	+3.34	+9.02	+7.50	+2.82	+0.41	+0.58
Mistral-7B-v0.3									
Vanilla [Jiang et al., 2023]	18.18	14.60	6.67	6.67	39.50	17.50	3.26	66.09	60.35
REBALANCE [Li et al., 2026]	24.24	14.80	6.67	6.67	43.97	20.00	4.00	66.34	60.29
DEER [Yang et al., 2025c]	25.76	16.20	10.00	3.33	40.11	17.50	3.85	66.09	60.94
MOI [Zhuang et al., 2026]	24.75	15.40	6.67	3.33	42.53	22.50	3.70	66.42	60.74
EELo-CoT [Zhao et al., 2025]	21.72	15.20	10.00	10.00	40.86	22.50	4.00	65.77	60.48
MGAI (ours)	**26.77**	**18.20**	**13.33**	**13.33**	**44.50**	**25.00**	**5.19**	**66.50**	**61.13**
Δ vs. Vanilla	+8.66	+3.60	+6.66	+6.66	+5.00	+7.50	+1.93	+0.41	+0.78
DeepSeek-R1-Distill-Qwen-1.5B									
Vanilla [DeepSeek-AI, 2025]	17.17	79.60	20.00	23.33	75.97	62.50	41.19	24.00	30.13
REBALANCE [Li et al., 2026]	21.70	83.00	**26.67**	**33.33**	77.33	72.50	43.85	25.60	30.30
DEER [Yang et al., 2025c]	16.16	67.00	23.33	20.00	69.22	57.50	35.41	25.53	30.24
MOI [Zhuang et al., 2026]	18.69	80.20	20.00	26.67	76.80	65.00	42.52	25.93	30.70
EELo-CoT [Zhao et al., 2025]	19.19	81.60	23.33	23.33	78.31	70.00	41.33	24.87	30.57
MGAI (ours)	**22.22**	**83.80**	**26.67**	**33.33**	**80.06**	**75.00**	**44.00**	**27.44**	**31.60**
Δ vs. Vanilla	+5.05	+4.20	+6.67	+10.00	+4.09	+12.5	+2.81	+3.44	+1.47

4 Experiments

4.1 Experimental Setup

Models. To verify the effectiveness of the proposed MGAI, we apply it to 5 widely used large language models, including LLaMA3-8B-Instruct [Grattafiori et al., 2024], Qwen2.5-14B-Instruct [Yang et al., 2025b], Qwen3-4B-Instruct [Yang et al., 2025a], Mistral-7B-v0.3 [Jiang et al., 2023] and DeepSeek-R1-Distill-Qwen-1.5B [DeepSeek-AI, 2025].

Baselines. As a training-free method, we compare the proposed MGAI with 4 existing training-free reasoning approaches and vanilla inference. Vanilla performs direct inference without intervention. DEER [Yang et al., 2025c] employs dynamic early exit based on confidence. REBALANCE [Li et al., 2026] steers reasoning by modulating hidden states. MOI [Zhuang et al., 2026] incorporates token distributions into autoregressive inputs to preserve uncertainty. EELo-CoT [Zhao et al., 2025] manipulates activations to elicit longer chain-of-thought reasoning.

Datasets. We evaluate our method on 9 benchmarks, including 7 reasoning datasets, 1 memory dataset and 1 general dataset. For reasoning tasks, we use 6 mathematical reasoning datasets: MATH-500 [Hendrycks et al., 2021], AIME2025 [OpenCompass, 2025], AIME2024 [AI-MO, 2024a],

GSM8K [Cobbe et al., 2021], AMC2023 [AI-MO, 2024b], OlympiadBench [He et al., 2024] and one scientific reasoning dataset: GPQA Diamond [Rein et al., 2023]. For memory and general tasks, we use CommonsenseQA [Talmor et al., 2019] and MMLU [Hendrycks et al., 2020] respectively.

Extraction of memory and reasoning representations as well as mapping function. The proposed characterization of memory and reasoning representations, as well as the mapping function that captures memory-to-reasoning alignment, are learned through a small set of correctly answered queries from the training splits of GSM8K and CSQA, corresponding to reasoning and memory tasks, respectively. The learned components are computed once in the offline stage for each backbone and are held fixed during inference across all unseen evaluation benchmarks in the online stage. *As shown in Table 1, representations extracted from a given dataset, along with the corresponding mapping function, generalize effectively across diverse benchmarks, indicating the robustness of our method.*

Implementation details. Unless otherwise specified, we select the Top-$K = 150$ memory-related neurons and reasoning-related neurons for all experiments, set the intervention strength to $\lambda = 11$, and use $c = 8$ for the SVD-based dimensionality reduction in the MRA metric. The data used to learn the mapping function and train the detection probe are kept strictly separate from the evaluation set.

More details about datasets, baselines and implementation details are listed in Appendix A.

4.2 Main Results

Main results on reasoning tasks. Table 1 presents the performance of the proposed MGAI across diverse reasoning benchmarks. Compared to vanilla inference and existing efficient reasoning approaches, MGAI attains higher accuracy on nearly all reasoning datasets and model backbones. The improvements are consistent across all the reasoning tasks, demonstrating the robustness of our method. For instance, on LLaMA3-8B-Instruct, MGAI improves performance on AMC23 from 25.00 to 35.00 and on GPQA-D from 31.31 to 36.87. Even on smaller models such as Qwen3-4B-Instruct, MGAI yields a remarkable improvement on GSM8K from 71.00 to 80.00. *Moreover, despite learning the memory-to-reasoning mapping function from only a small set of correctly answered queries, the proposed MGAI achieves strong performance across a wide range of unseen benchmarks, showing its strong cross-task generalization ability.* This indicates that the learned mapping captures a general alignment between memory representations and reasoning representations.

MGAI preserves previous memory and general capabilities after intervention. Beyond reasoning tasks, we evaluate the proposed MGAI on CSQA and MMLU to assess its impact on memory and general capabilities. Results in Table 1 show that MGAI maintains performance on memory and general tasks. Specifically, MGAI improves over vanilla inference on CSQA for four out of five models and on MMLU for all five models, indicating that gains in reasoning do not come at the cost of degrading memory or general capabilities. This is consistent with the design of MGAI, which performs targeted interventions on reasoning representations by injecting memory-aligned signals, rather than globally altering the model's parameters.

4.3 Ablation Study

Sensitivity analysis of the number of selected neurons. We further ablate the number of selected top-ranked neurons. Specifically, we vary the Top-K from 50 to 300 and apply our MGAI method using LLaMA3-8B-Instruct on diverse benchmarks. As shown in Figure 4(a), Top-$K = 150$ achieves the best reasoning performance and preserves the memory and general capabilities of the LLM. Note that, although Figure 3(a) indicates that approximately 50 top-ranked reasoning-related neurons are sufficient to characterize the reasoning representations, the results in Figure 4(a) indicate that selecting $K = 50$ or $K = 100$ leads to relatively weaker reasoning performance. This discrepancy might be because additional neurons also contribute to basic semantic understanding and provide useful contextual information for reasoning, so we adopt a moderately larger Top-$K = 150$.

Verifying the effectiveness of the intervention direction. To examine whether the performance gain of our MGAI method indeed comes from the learned intervention direction $\Delta\mathbf{h}$, we compare the effects of applying the learned direction and five randomly sampled directions orthogonal to it on the reasoning performance of LLaMA3-8B-Instruct on the GPQA-D dataset. As shown in Figure 4(b), applying the learned direction consistently outperforms orthogonal random directions in improving

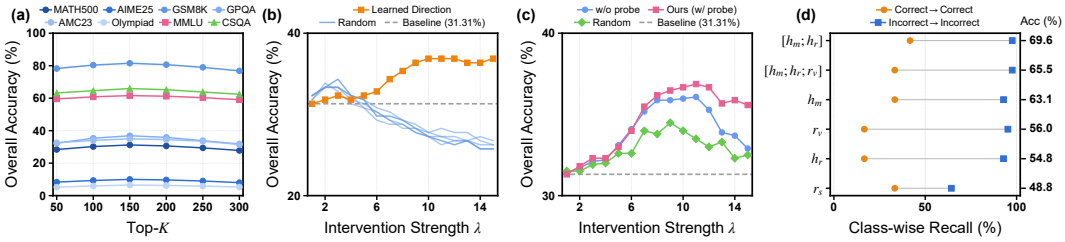


Figure 4: (a) Sensitivity analysis of the number of selected neurons (Top-K) across multiple benchmarks. Results show that $K = 150$ achieves the best performance on all benchmarks. (b) Verifying the effectiveness of the learned intervention direction by comparing overall accuracy under the learned direction and five random directions orthogonal to it. Results show that the learned intervention direction consistently improves reasoning under different intervention strengths. (c) Verifying the effectiveness of the detection probe. Results show that our probe-based gating outperforms intervening without probe or on random samples across different intervention strengths. (d) Feature ablation for the detection probe. Results show that using $[\mathbf{h}_m; \mathbf{h}_r]$ as probe input achieves the best accuracy.

Table 2: Ablation on datasets used for neuron localization. Column headers indicate the datasets for localizing reasoning-related neurons, and rows indicate the evaluation datasets. All results are evaluated with LLaMA3-8B-Instruct.

Dataset	Method	GPQA-D	MATH-500	AIME25	AIME24	GSM8K	AMC23	Olympiad
GPQA-D	Vanilla [Grattafiori et al., 2024]	31.31	31.31	31.31	31.31	31.31	31.31	31.31
	MGAI(ours)	**34.34**	**37.37**	**38.89**	**38.38**	**36.87**	**35.86**	**36.36**
MATH-500	Vanilla [Grattafiori et al., 2024]	60.40	60.40	60.40	60.40	60.40	60.40	60.40
	MGAI(ours)	**62.20**	**66.00**	**66.40**	**66.80**	**65.20**	**64.60**	**62.80**
GSM8K	Vanilla [Grattafiori et al., 2024]	76.95	76.95	76.95	76.95	76.95	76.95	76.95
	MGAI(ours)	**79.00**	**81.27**	**81.88**	**81.42**	**81.50**	**79.76**	**80.44**

overall accuracy as intervention strength increases. These results indicate that the performance improvement is specifically attributed to the learned memory-guided intervention direction.

Verifying the effectiveness of the detection probe. To examine the necessity of the detection probe and the sensitivity to the intervention strength, we vary the intervention strength λ and compare two strategies using LLaMA3-8B-Instruct on the GPQA-D dataset: (1) **w/o probe**, which applies the intervention to all samples x, *i.e.*, $p_{\text{fail}}(x) > \tau, \tau = 0$, (2) **Ours(w/ probe)**, gating based on the probe-predicted failure probability, *i.e.*, $p_{\text{fail}}(x) > \tau, \tau = 0.5$. As shown in Figure 4(c), our probe-based gating outperforms intervening on all or random samples across different intervention strengths λ, indicating the effectiveness of selecting failure-prone samples. These results demonstrate that the detection probe is essential for robust intervention.

Feature ablation for the detection probe. We ablate the input features used to train the detection probe using LLaMA3-8B-Instruct on the GPQA-D dataset. As shown in Figure 4(d), using the concatenation of memory and reasoning activations $[\mathbf{h}_m, \mathbf{h}_r]$ achieves the best accuracy. In contrast, using only $\mathbf{h}_m$, $\mathbf{h}_r$, the vector residual $\mathbf{r}_v = W\mathbf{h}_m - \mathbf{h}_r$, or the scalar residual $r_s = \|\mathbf{r}_v\|_2$ leads to lower performance. These results indicate that $\mathbf{h}_m$ and $\mathbf{h}_r$ provide complementary signals for distinguishing correct and incorrect queries, supporting our choice of $[\mathbf{h}_m, \mathbf{h}_r]$ as the probe input.

Verifying robustness across datasets for localizing neurons. To verify the effectiveness of the reasoning-related neurons localized from different reasoning datasets, we use seven reasoning datasets to independently localize Top-K reasoning-related neurons of LLaMA3-8B-Instruct, then apply our MGAI method on three datasets, including GPQA-D, MATH-500 and GSM8K. As Table 2 shows, applying interventions based on each reasoning-related neuron set consistently improves reasoning performance, which suggests robustness of datasets used for localizing reasoning-related neurons.

5 Conclusion and Limitation

In this paper, we study how an LLM leverages its internal knowledge during reasoning and find that failures are often associated with insufficient memory engagement. We introduce the Memory-to-Reasoning Alignment (MRA) metric and propose a training-free Memory-Guided Activation

Injection (MGAI) approach to enhance the alignment between memory representation and reasoning representation at inference time. Experimental results show that MGAI improves reasoning performance with strong cross-task generalization while maintaining the model's memory and general capabilities. This provides a general, plug-and-play intervention framework for enhancing memory engagement during reasoning without additional training or architectural changes.

Limitations. The limitations of our MGAI framework are as follows: (1) it requires access to internal activations, so it cannot be directly applied to closed-source LLMs that only provide black-box API access; (2) the detection probe used to determine the query-specific intervention strength is still relatively opaque, and future work could explore more interpretable probe designs to better explain why and how much each query should be intervened.

Acknowledgments and Disclosure of Funding

This work is partially supported by the Shanghai Science and Technology Commission (No. 25511102900), the National Nature Science Foundation of China (Nos. 62576249, 62376199, 62576247), and the Shanghai Municipal Education Commission (No. 24CGA20).

References

AI-MO. Aime 2024. `https://huggingface.co/datasets/AI-MO/aimo-validation-aime`, July 2024a.

AI-MO. Amc 2023. `https://huggingface.co/datasets/AI-MO/aimo-validation-amc`, July 2024b.

Mingyang Chen, Linzhuang Sun, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z Pan, Wen Zhang, Huajun Chen, et al. Learning to reason with search for llms via reinforcement learning. *arXiv preprint arXiv:2503.19470*, 2025.

Sitao Cheng, Liangming Pan, Xunjian Yin, Xinyi Wang, and William Yang Wang. Understanding the interplay between parametric and contextual knowledge for large language models. *arXiv preprint arXiv:2410.08414*, 2024.

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

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL `https://arxiv.org/abs/2501.12948`.

Zorik Gekhman, Roee Aharoni, Eran Ofek, Mor Geva, Roi Reichart, and Jonathan Herzig. Thinking to recall: How reasoning unlocks parametric knowledge in llms. *arXiv preprint arXiv:2603.09906*, 2026.

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

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3828–3850, 2024.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Baizhou Huang and Xiaojun Wan. Pros: Towards compute-efficient rlvr via rollout prefix reuse. In *The Fourteenth International Conference on Learning Representations*, 2026.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL `https://arxiv.org/abs/2310.06825`.

Mingyu Jin, Weidi Luo, Sitao Cheng, Xinyi Wang, Wenyue Hua, Ruixiang Tang, William Yang Wang, and Yongfeng Zhang. Disentangling memory and reasoning ability in large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1681–1701, 2025.

Hirokazu Kiyomaru, Issa Sugiura, Daisuke Kawahara, and Sadao Kurohashi. A comprehensive analysis of memorization in large language models. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 584–596, 2024.

Miyoung Ko, Sue Hyun Park, Joonsuk Park, and Minjoon Seo. Hierarchical deconstruction of llm reasoning: A graph-based framework for analyzing knowledge utilization. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4995–5027, 2024.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Gang Li, Yan Chen, Ming Lin, and Tianbao Yang. Drpo: Efficient reasoning via decoupled reward policy optimization. *arXiv preprint arXiv:2510.04474*, 2025a.

Yulin Li, Tengyao Tu, Li Ding, Junjie Wang, Huiling Zhen, Yixin Chen, Yong Li, and Zhuotao Tian. Efficient reasoning with balanced thinking. *arXiv preprint arXiv:2603.12372*, 2026.

Yuqing Li, Jiangnan Li, Zheng Lin, Ziyan Zhou, Junjie Wu, Weiping Wang, Jie Zhou, and Mo Yu. Mindscape-aware retrieval augmented generation for improved long context understanding. *arXiv preprint arXiv:2512.17220*, 2025b.

Wei Liu, Ruochen Zhou, Yiyun Deng, Yuzhen Huang, Junteng Liu, Yuntian Deng, Yizhe Zhang, and Junxian He. Learn to reason efficiently with adaptive length-based reward shaping. *arXiv preprint arXiv:2505.15612*, 2025.

Sadegh Mahdavi, Renjie Liao, and Christos Thrampoulidis. Memorization capacity of multi-head attention in transformers. *arXiv preprint arXiv:2306.02010*, 2023.

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

OpenCompass. Aime 2025. `https://huggingface.co/datasets/opencompass/AIME2025`, February 2025.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *Advances in neural information processing systems*, 30, 2017.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Yu Sheng, Linjing Li, and Daniel Dajun Zeng. Learning theorem rationale for improving the mathematical reasoning capability of large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 15151–15159, 2025.

Peiyang Song, Pengrui Han, and Noah Goodman. Large language model reasoning failures. *arXiv preprint arXiv:2602.06176*, 2026.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, 2019.

Zhengwei Tao, Zhi Jin, Yifan Zhang, Xiancai Chen, Haiyan Zhao, Jia Li, Bin Liang, Chongyang Tao, Qun Liu, and Kam-Fai Wong. A comprehensive evaluation on event reasoning of large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25273–25281, 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*, pages 9840–9855, 2023.

Xinyi Wang, Antonis Antoniades, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. Generalization vs memorization: Tracing language models' capabilities back to pretraining data. *arXiv preprint arXiv:2407.14985*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

An Yang, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoyan Huang, Jiandong Jiang, Jianhong Tu, Jianwei Zhang, Jingren Zhou, Junyang Lin, Kai Dang, Kexin Yang, Le Yu, Mei Li, Minmin Sun, Qin Zhu, Rui Men, Tao He, Weijia Xu, Wenbiao Yin, Wenyuan Yu, Xiafei Qiu, Xingzhang Ren, Xinlong Yang, Yong Li, Zhiying Xu, and Zipeng Zhang. Qwen2.5-1m technical report, 2025b. URL `https://arxiv.org/abs/2501.15383`.

Chenxu Yang, Qingyi Si, Yongjie Duan, Zheliang Zhu, Chenyu Zhu, Qiaowei Li, Minghui Chen, Zheng Lin, and Weiping Wang. Dynamic early exit in reasoning models. *arXiv preprint arXiv:2504.15895*, 2025c.

Zekai Zhao, Qi Liu, Kun Zhou, Zihan Liu, Yifei Shao, Zhiting Hu, and Biwei Huang. Activation control for efficiently eliciting long chain-of-thought ability of language models. *arXiv preprint arXiv:2505.17697*, 2025.

Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Revisiting the importance of individual units in cnns via ablation. *arXiv preprint arXiv:1806.02891*, 2018.

Chulun Zhou, Chunkang Zhang, Guoxin Yu, Fandong Meng, Jie Zhou, Wai Lam, and Mo Yu. Improving multi-step rag with hypergraph-based memory for long-context complex relational modeling. *arXiv preprint arXiv:2512.23959*, 2025.

Tongyao Zhu, Qian Liu, Liang Pang, Zhengbao Jiang, Min-Yen Kan, and Min Lin. Beyond memorization: The challenge of random memory access in language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3373–3388, 2024.

Yufan Zhuang, Liyuan Liu, Chandan Singh, Jingbo Shang, and Jianfeng Gao. Mixture of inputs: Text generation beyond discrete token sampling. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL `https://openreview.net/forum?id=l6C6Pw30Gl`.

A More Experimental Details

A.1 Benchmark

We evaluate our method on nine benchmarks, including seven reasoning task datasets, one memory task dataset and one general task dataset.

GPQA Diamond (198 problems) [Rein et al., 2023] is a challenging subset of GPQA, consisting of graduate-level, Google-proof questions mainly covering scientific domains such as biology, physics, and chemistry. The Diamond subset contains questions carefully validated by domain experts and is widely used to evaluate advanced reasoning ability. In our experiments, we evaluate models on GPQA Diamond based on multiple-choice accuracy.

MATH-500 (500 problems) [Hendrycks et al., 2021] is a representative subset of the MATH benchmark, containing 500 competition-level mathematical problems across diverse topics, including algebra, geometry, number theory, counting and probability, and precalculus. This benchmark requires models to perform multi-step mathematical reasoning rather than simple pattern matching.

AIME2025 (30 problems) [OpenCompass, 2025] consists of problems from the 2025 American Invitational Mathematics Examination, which are high-school-level competition math problems requiring non-trivial symbolic and numerical reasoning. The answers are typically integers, making the benchmark suitable for evaluating precise mathematical reasoning.

AIME2024 (30 problems) [AI-MO, 2024a] contains problems from the 2024 American Invitational Mathematics Examination. Similar to AIME2025, it focuses on challenging competition-style mathematical reasoning and requires models to derive a precise final integer answer.

GSM8K (1319 problems) [Cobbe et al., 2021] is a grade-school math word-problem benchmark that requires models to solve multi-step arithmetic reasoning problems expressed in natural language. Although the mathematical operations are elementary, the benchmark tests whether models can correctly follow the reasoning chain and compute the final numerical result.

AMC2023 (40 problems) [AI-MO, 2024b] consists of problems from the 2023 American Mathematics Competitions, covering diverse mathematical topics such as algebra, geometry, combinatorics, and number theory. Compared with grade-school arithmetic benchmarks, AMC2023 requires stronger competition-level problem-solving ability.

OlympiadBench (675 problems) [He et al., 2024] is a benchmark designed to evaluate advanced olympiad-level reasoning ability, covering challenging problems in domains such as mathematics and science. The problems usually require complex reasoning and precise answer derivation.

CommonsenseQA (1221 problems) [Talmor et al., 2019] is a multiple-choice question answering benchmark designed to evaluate commonsense reasoning over everyday concepts and relations. Correctly answering these questions often requires the model to use implicit world knowledge rather than relying solely on surface-level text patterns.

MMLU (1531 problems) [Hendrycks et al., 2020] is a broad multi-task benchmark that evaluates models across 57 subjects, including STEM, humanities, social sciences, and professional knowledge. It is commonly used to measure the general knowledge and problem-solving capabilities of large language models.

A.2 Baseline Methods

REBALANCE [Zhao et al., 2025]: REBALANCE is a training-free framework that balances overthinking and underthinking using confidence-based signals. It monitors reasoning dynamics and applies a steering vector to adjust inference behavior, enabling adaptive control over reasoning depth and improving both efficiency and answer accuracy.

DEER [Yang et al., 2025c]: DEER is a training-free inference method that reduces overthinking in reasoning models by dynamically truncating chain-of-thought generation. It detects high-confidence intermediate states and halts further reasoning, improving efficiency while preserving accuracy through adaptive early-exit decisions without requiring additional training or architectural changes.

MOI [Zhuang et al., 2026]: MOI is a training-free generation method that improves reasoning by preserving token-level uncertainty during decoding. It combines sampled tokens with their original probability distributions to form continuous inputs, allowing the model to maintain alternative hypotheses and generate more robust, accurate outputs.

EELo-CoT [Zhao et al., 2025]: EELo-CoT is a training-free approach that enhances long-form reasoning by manipulating key activations in final model layers. By amplifying reasoning-related signals and introducing auxiliary tokens, it elicits longer, more structured chain-of-thought outputs, improving reasoning performance without fine-tuning or supervised data.

A.3 Implementation details

All experiments are conducted on four RTX 5090 32GB GPUs. Unless otherwise specified, we select the Top-$K = 150$ memory-related neurons and reasoning-related neurons for all experiments, set the intervention strength to $\lambda = 11$, and use $c = 8$ for the SVD-based dimensionality reduction in the MRA metric.

The data used to learn the mapping function and train the detection probe are kept strictly separate from the evaluation set. The mapping function that captures memory-to-reasoning alignment, is learned through a small set of correctly answered queries from the training splits of GSM8K and CSQA, corresponding to reasoning and memory tasks, respectively. The detection probe is trained on a small set of the training splits of GSM8K. The learned components are computed once in the offline stage for each backbone and held fixed during inference across all unseen evaluation benchmarks in the online stage.

A.4 Computation Details of Canonical Correlation Analysis (CCA)

Canonical Correlation Analysis (CCA) measures the similarity between two sets of representations by finding linear projections that maximize their correlation. Given two centered activation matrices

$$X \in \mathbb{R}^{N \times d_x}, \qquad Y \in \mathbb{R}^{N \times d_y}, \tag{12}$$

where each row corresponds to one input sample, CCA seeks projection vectors $\mathbf{a}$ and $\mathbf{b}$ such that the correlation between $X\mathbf{a}$ and $Y\mathbf{b}$ is maximized:

$$\rho = \max_{\mathbf{a},\mathbf{b}} \frac{\mathbf{a}^\top \Sigma_{XY} \mathbf{b}}{\sqrt{\mathbf{a}^\top \Sigma_{XX} \mathbf{a}} \sqrt{\mathbf{b}^\top \Sigma_{YY} \mathbf{b}}}, \tag{13}$$

where Σ_{XX} and Σ_{YY} are the covariance matrices of X and Y, and Σ_{XY} is their cross-covariance matrix. Solving this optimization yields a sequence of canonical correlations

$$\rho_1, \rho_2, \ldots, \rho_q, \qquad q = \min(d_x, d_y), \tag{14}$$

which quantify the alignment between the two representations along different canonical directions. Following the common practice in representation similarity analysis, we use the average canonical correlation as the overall CCA similarity:

$$\mathrm{CCA}(X, Y) = \frac{1}{q} \sum_{i=1}^{q} \rho_i. \tag{15}$$

A larger CCA value indicates stronger alignment between the two representations.

B Additional Experimental Results

B.1 Separability under Additional Top-K Settings.

We further evaluate the separability between memory-related and reasoning-related neurons under additional Top-K settings. Specifically, we vary the number of selected neurons with $K \in \{50, 100, 150, 200\}$ and measure Spearman correlation and Overlap$^{(K)}$ between neuron sets. As shown in Figure 5, the reasoning-related and memory-related neuron sets remain weakly overlapped and negatively correlated across all K values. Meanwhile, memory-related neurons consistently have much higher overlap and stronger positive correlation with general-capability neurons. This suggests that the separability between memory-related and reasoning-related neurons is stable across different top neuron selection.

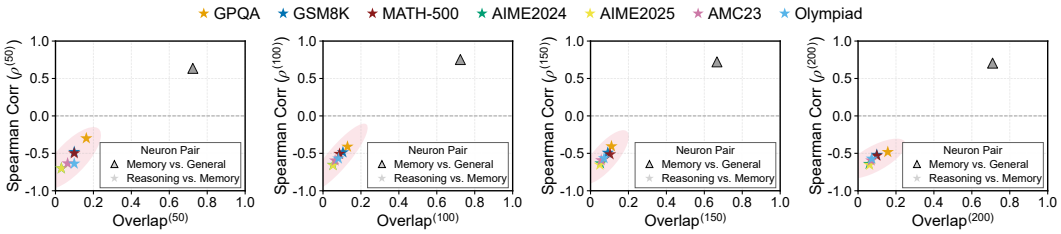


Figure 5: Separability between memory-related and reasoning-related neurons under different selection of Top-K neurons.

B.2 Justification of the Linear Memory-to-Reasoning Mapping Function

In Eq. (7), we adopt a linear least-squares objective to learn the mapping from memory representations to reasoning representations. Since the representation space of large language models is highly non-linear, one may question whether a linear mapping is sufficient for capturing the memory-to-reasoning alignment.

To address this concern, we provide both theoretical motivation and empirical verification. Specifically, we compare the proposed linear mapping with two non-linear mapping alternatives based on shallow multi-layer perceptrons (MLPs) on three datasets using LLaMA3-8B-Instruct. Experimental results show that the linear mapping already achieves comparable or even better performance than more expressive non-linear mappings on reasoning, memory, and general tasks, despite using significantly fewer parameters. These results suggest that the memory-to-reasoning alignment learned in our framework mainly lies in a compact and approximately linear subspace after neuron selection.

B.2.1 Experimental Setup

To empirically verify whether a linear mapping is sufficient, we compare the proposed linear mapping with two non-linear mapping alternatives on 3 benchmarks, including GPQA-Diamond, CommonsenseQA, and MMLU, using LLaMA3-8B-Instruct. All experiments use the same neuron localization strategy and intervention pipeline as the main experiments.

Mapping Functions. Following the main experimental setup, the mapping functions are learned in the offline stage using correctly answered samples from GSM8K and CSQA, corresponding to reasoning and memory tasks, respectively. We strictly ensure that the samples used for learning the mapping functions do not overlap with any downstream evaluation samples, thereby avoiding potential data leakage.

Specifically, we compare three mapping functions: (1) **Linear**, the linear least-squares mapping used in Equation 7; (2) **MLP-**d_h, a 2-layer MLP with hidden dimension $d_h = 150$; and (3) **MLP-**$2d_h$, a 2-layer MLP with hidden dimension $2d_h = 300$. Given a query x, the architecture used by two MLP variants is as follows:

$$\hat{\mathbf{y}}_r(x) = f_{\text{MLP}}(\mathbf{h}_m(x)) = W_2\,\text{GELU}(W_1\mathbf{h}_m(x) + b_1) + b_2, \tag{16}$$

where $\mathbf{h}_m(x) \in \mathbb{R}^{1\times K}$ denotes the memory representation, $W_1 \in \mathbb{R}^{K\times d_h}$ and $W_2 \in \mathbb{R}^{d_h\times K}$ denote the learnable weight matrices of the two linear layers, $b_1 \in \mathbb{R}^{1\times d_h}$ and $b_2 \in \mathbb{R}^{1\times K}$ denote the corresponding bias vectors, $\text{GELU}(\cdot)$ denotes the Gaussian Error Linear Unit activation function, and $\hat{\mathbf{y}}_r(x) \in \mathbb{R}^{1\times K}$ denotes the predicted reasoning representation.The parameter counts of Linear, MLP-d_h, and MLP-$2d_h$ are 22,500, 45,300, and 90,450, respectively.

Regression Metrics. To evaluate the fitting quality of different mapping functions, we compute the Mean Squared Error (MSE) and the Explained Variance on the validation set.

Given a validation set $\mathcal{Q} = \{x_j\}_{j=1}^N$, the MSE is defined as

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N} \|\hat{\mathbf{y}}_r(x) - \mathbf{y}_r(x)\|_2^2, \tag{17}$$

Table 3: MSE (↓) and ExplainedVar (↑) of different mapping functions on reasoning, memory, and general tasks on LLaMA3-8B-Instruct.

Task	Dataset	Mapping Function	Parameter Counts	MSE ↓	ExplainedVar ↑
Reasoning	GPQA	Linear	22,500	0.03775	0.99674
		MLP-d_h	45,300	0.03534	0.99694
		MLP-$2d_h$	90,450	0.03437	0.99703
Memory	CSQA	Linear	22,500	0.02685	1.00000
		MLP-d_h	45,300	0.02877	1.00000
		MLP-$2d_h$	90,450	0.02823	1.00000
General	MMLU	Linear	22,500	0.04525	1.00000
		MLP-d_h	45,300	0.04123	1.00000
		MLP-$2d_h$	90,450	0.03874	1.00000

Table 4: Accuracy(%) (↑) on reasoning, memory, and general tasks after applying activation interventions generated by different mapping functions on LLaMA3-8B-Instruct.

Task	Dataset	Mapping Function	Vanilla Acc.(%)	MGAI Acc.(%) ↑
Reasoning	GPQA	Linear	31.31	36.87
		MLP-d_h		33.84
		MLP-$2d_h$		33.33
Memory	CSQA	Linear	63.64	65.93
		MLP-d_h		65.77
		MLP-$2d_h$		65.60
General	MMLU	Linear	60.55	61.59
		MLP-d_h		61.59
		MLP-$2d_h$		60.81

and the Explained Variance is computed as

$$\text{ExplainedVar} = 1 - \frac{\text{Var}(\hat{\mathbf{y}}_r(x) - \mathbf{y}_r(x))}{\text{Var}(\mathbf{y}_r(x))}, \tag{18}$$

where $\hat{\mathbf{y}}_r(x)$ denotes the predicted reasoning representation and $\mathbf{y}_r(x)$ represents the target reasoning representation.

We also evaluate the accuracy of LLaMA3-8B-Instruct after applying activation interventions generated by the three mapping functions on reasoning, memory, and general tasks.

Training data and evaluation protocol. Following the main experimental setup, the mapping functions are learned in the offline stage using correctly answered samples from GSM8K and CSQA, corresponding to reasoning and memory tasks, respectively. The learned mappings are then evaluated on downstream benchmarks including GPQA, CSQA and MMLU. We strictly ensure that the samples used for learning the mapping functions do not overlap with any downstream evaluation samples. In particular, the training data used for fitting both the linear mapping and the non-linear MLP variants are fully disjoint from the evaluation datasets, thereby avoiding potential data leakage. Therefore, the downstream improvements reflect genuine cross-dataset generalization of the learned memory-to-reasoning alignment.

B.2.2 Results

Table 3 reports the regression fitting quality of different mapping functions. Although the two MLP-based mappings slightly reduce the validation MSE on some datasets, the improvements are marginal. In all settings, the explained variance is already extremely high for the linear mapping, reaching values close to 1.0. This indicates that the memory representations and reasoning representations already exhibit an approximately linear relationship in the selected subspace.

More importantly, Table 4 shows that better regression fitting quality does not translate into better downstream reasoning performance. On GPQA, the proposed linear mapping achieves the best downstream intervention accuracy (36.87%), outperforming both MLP-d_h (33.84%) and MLP-$2d_h$ (33.33%). Similar observations hold on CSQA and MMLU, where the linear mapping consistently achieves comparable or better performance than the non-linear alternatives.

These results suggest that increasing the expressive capacity of the mapping function does not improve the quality of the intervention direction. Instead, the additional non-linearity may introduce unnecessary complexity and overfitting to local activation patterns, which weakens the cross-task generalization ability of the learned intervention direction.

Furthermore, the proposed linear mapping is substantially more parameter-efficient. Compared with MLP-$2d_h$, the linear mapping uses only approximately one-fourth of the parameters while achieving consistently stronger or comparable downstream performance.

Overall, these results empirically verify that the linear mapping used in Eq. (7) is already sufficient for capturing the memory-to-reasoning alignment required by our intervention framework.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction (Section 1) clearly state our contributions: (i) the Memory-to-Reasoning Alignment (MRA) metric for measuring memory engagement, (ii) the training-free Memory-Guided Activation Injection (MGAI) method, and (iii) the adaptive intervention strategy. These claims are supported by the analyses in Section 3 and the experimental results in Section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our work in Section 5 (Conclusion and Limitation). (i) Our method requires access to internal model activations, which limits its direct applicability to closed-source LLMs that only provide black-box API access; (ii) the detection probe used to determine query-specific intervention strength remains relatively opaque, motivating future work on more interpretable probe designs to better explain why and how much each query should receive intervention.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover

limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: This paper does not include formal theoretical results or theorems requiring proofs. The proposed method is supported by empirical analyses and experiments rather than theoretical derivations.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 3 fully describes the MGAI method (neuron localization, mapping function, detection probe, and activation injection), and Section 4.1 specifies the models, baselines, datasets, and implementation details (e.g., Top-$K = 150$, intervention strength $\lambda = 11$, threshold $\tau = 0.5$). All datasets used are publicly available benchmarks.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: All datasets used in this paper are publicly available, and all evaluated backbone models are open-weight. However, due to copyright constraints on parts of the implementation, we are unable to publicly release the full codebase and all experimental scripts at submission time.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4.1 details the experimental setting, including the five backbone models, baselines, datasets, and key hyperparameters (Top-$K = 150$ neurons, intervention strength $\lambda = 11$, gating threshold $\tau = 0.5$).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Following the common practice in evaluating LLMs on these reasoning benchmarks, we report accuracy on the standard test sets. We do not report error bars due to the high computational cost of repeatedly running large models on these benchmarks. The consistent gains observed across five different models and nine benchmarks help mitigate concerns about variance.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: As MGAI is a training-free method, it only requires inference. All experiments are conducted on RTX 5090 32GB GPUs. Detailed compute information is provided in the supplemental material.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and our research fully conforms to it. Our work uses publicly available models and benchmarks, does not involve human subjects, and preserves anonymity in submission.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This work is foundational research on improving the reasoning ability of LLMs by better leveraging their internal memory. It does not introduce new models or datasets and is not tied to any particular deployment. We do not foresee direct negative societal impacts beyond those that are common to the broader field of LLM research.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release any new pre-trained models, generative models, or scraped datasets. MGAI is a training-free intervention method applied to existing publicly released LLMs and thus poses no additional misuse risks beyond those of the underlying models.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly cite the original papers for all models (LLaMA3, Qwen2.5, Qwen3, Mistral, DeepSeek-R1-Distill-Qwen) and benchmarks (MATH-500, AIME2024/2025, GSM8K, AMC2023, OlympiadBench, GPQA Diamond, CommonsenseQA, MMLU). All assets are used in accordance with their licenses and terms of use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not introduce new datasets or pre-trained models. We only provide our anonymized implementation code in the supplemental material, which is documented with a README and configuration files.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects, so IRB approval is not required.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The LLM is used only for writing, editing, or formatting purposes.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.