LongAct: Harnessing Intrinsic Activation Patterns for Long-Context Reinforcement Learning

Anonymous ACL submission

Abstract

Reinforcement Learning (RL) has emerged as a critical driver for enhancing the reasoning capabilities of Large Language Models (LLMs). While recent advancements have focused on reward engineering or data synthesis, few studies exploit the model's intrinsic representation characteristics to guide the training process. In this paper, we first observe the presence of high-magnitude activations within the query and key vectors when processing long contexts. Drawing inspiration from model quantization—which establishes the criticality of such high-magnitude activations—and the insight that long-context reasoning inherently exhibits a sparse structure, we hypothesize that these weights serve as the pivotal drivers for effective model optimization. Based on this insight, we propose LongAct, a strategy that shifts from uniform to saliency-guided sparse updates. By selectively updating only the weights associated with these significant activations, LongAct achieves an approximate 8% improvement on LongBench v2 and enhances generalization on the RULER benchmark. Furthermore, our method exhibits remarkable universality, consistently boosting performance across diverse RL algorithms such as GRPO and DAPO. Extensive ablation studies suggest that focusing on these salient features is key to unlocking long-context potential.

1 Introduction

Reinforcement Learning (RL) has been proven to be a catalyst for eliciting the reasoning capabilities of Large Language Models (LLMs) (Guo et al., 2025; Team et al., 2025a). This capability is particularly pivotal in long-context scenarios. Real-world long-context tasks, such as long-dialogue history understanding and long structured data analysis, are characterized not only by their extensive input lengths but also by the necessity for deep comprehension and complex reasoning over the content. In addition, enhancing long-context understanding is instrumental for LLM agents in managing extended trajectories.

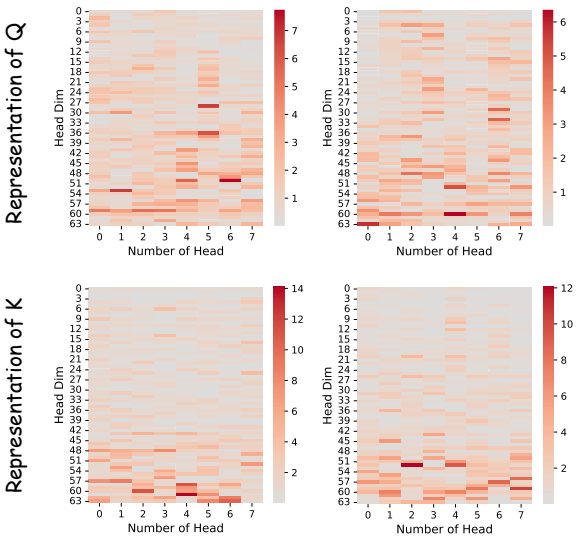

Figure 1: Visualization of the query/key (Q/K) representation magnitudes in Qwen3-8B on the RULER benchmark (Common Words Extraction subset). We show the first 8 attention heads and the first 64 dimensions within each head for clarity. The x-axis denotes the head index and the y-axis denotes the head dimension.

Recently, researchers have begun to explore the application of RL in long-context scenarios (Bai et al., 2025; Zeng et al., 2025). Existing efforts primarily optimize external supervision signals or training curricula. For instance, some approaches focus on synthesizing high-quality reasoning data (Wang et al., 2025) or employing granular reward functions to mitigate sparse feedback (Anonymous, 2025), while others adopt progressive context scaling strategies (Wan et al., 2025). Parallel efforts have even explored modifying the model architecture itself to handle extended sequences (QwenTeam, 2025; Team et al., 2025b). However, these methods effectively treat the model's internal computation as a black box.

Previous work (Hao et al., 2024; Deng et al., 2025) suggests that complex deduction relies on continuous "thought trajectories" within the hidden state space rather than merely surface-level token generation. Yet, current long-context RL paradigms largely overlook the features embedded in these latent representations.

To bridge this gap, we propose LongAct, a method that leverages the model's intrinsic activation patterns to guide the training process. Our intuition is grounded in two complementary insights. First, prior studies (Lin et al., 2024; Jin et al., 2025) demonstrate that hidden dimensions are not equally important—high-magnitude activations often encode disproportionately critical information compared to the rest. Second, long-context inherently exhibits a sparse structure, as evidenced by methods that achieve full-context performance utilizing only a subset of selected tokens (Xiao et al., 2024; Zhao et al., 2025). We hypothesize that this sparsity extends beyond the token level to the hidden state dimension. As illustrated in Figure 1, we empirically observe this phenomenon as sparse, high-magnitude activations within the query and key vectors. Identifying these activations as the structural "anchors" for long-context reasoning, LongAct adopts a sparse, saliency-guided strategy, selectively updating only the weights linked to these significant features. This targeted approach yields an approximate 8% improvement on LongBench v2 (Bai et al., 2025), demonstrating that focusing on intrinsic saliency is key to unlocking long-context potential.

LongAct exhibits remarkable universality, enhancing generalization on generic long-context tasks (e.g., RULER and InfiniteBench (Hsieh et al., 2024; Zhang et al., 2024))—evidenced by a 4% gain on 128K RULER in Table 2—while consistently boosting performance across a diverse spectrum of RL algorithms, including GRPO, DAPO, and KL-Cov (Shao et al., 2024; Yu et al., 2025b; Cui et al., 2025) shown in Table 4. Furthermore, ablation studies indicate that updating weights associated with high-magnitude activations is critical for these improvements. Specifically, our strategy achieves an overall score of 36.73 on LongBench v2, significantly outperforming methods that update low-magnitude (29.82) shown in Table 5. Finally, case-level analysis in Figure 6 illustrates that disrupting high-magnitude activations triggers immediate model collapse (e.g., repetitive loops), whereas neutralizing low-magnitude counterparts preserves reasoning coherence.

Our contributions can be summarized as follows:

- We propose LongAct that leverages intrinsic high-magnitude activations to guide sparse reinforcement learning.
- We conduct extensive experiments (using LongBench v2, RULER, etc.) to validate the effectiveness of LongAct. For instance, LongAct achieves an improvement of 8% on LongBench v2.
- We provide in-depth experiments and analysis to elucidate the efficacy of LongAct, identifying high-magnitude activations is critical for model reasoning in long-context scenarios.

2 Related work

Reinforcement Learning in Long-context Scenario Reinforcement Learning (RL) is widely used in tasks such as mathematics (Yu et al., 2025b; Zheng et al., 2025). Recently, researchers have begun to explore the application of RL in long-context scenarios. Many researchers modify model architectures, adopting methods such as linear attention and sparse attention (QwenTeam, 2025; Team et al., 2025b,c; Gao et al., 2025) which require pre-training. Wan et al. (2025) use progressive context scaling during RL. LongRLVR employs a carefully designed reward function (Anonymous, 2025). Wang et al. (2025) propose to synthesize better long-context reasoning data. In contrast to prior work, our method leverages the model's internal mechanisms and is complementary to existing approaches.

Import Values in Attention Modules Numerous studies have investigated high-magnitude activation (Dettmers et al., 2022; Ahmadian et al., 2023; Guo et al., 2024; Xu et al., 2024). Many methods have proven effective; more specifically, Lin et al. (2024) preserve weights related to high-magnitude activations with high precision during quantization, while Liu et al. (2024) employ asymmetric quantization guided by the distribution of high-magnitude activations in the KV cache. Several other studies have investigated the impact of RoPE on model activations (Barbero et al., 2024; Jin et al., 2025). Departing from previous research, we analyze how high-magnitude activations influence performance in long-context reasoning tasks.

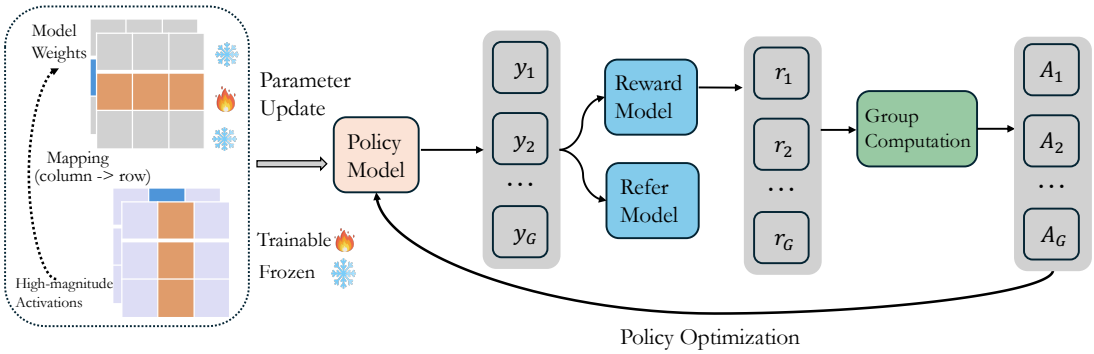

Figure 2: Overview of the LongAct framework. The left panel illustrates the dynamic saliency-guided sparse update mechanism: distinct high-magnitude activations (Orange/Blue columns) in the projections (e.g., Query/Key) dynamically map to their corresponding weight rows for sparse updates, while keeping other parameters frozen. Given the projection weight shape $\mathbf{W} \in \mathbb{R}^{d_{out} \times d_{in}}$, high-magnitude outliers in the activation output channels (columns) correspond directly to specific rows in $\mathbf{W}$. The right panel depicts the standard group-based policy optimization loop.

3 Method

3.1 Preliminary

High-magnitude Activations We focus on the attention layers, which serve as the core components for context modeling. Let H_Q and H_{KV} denote the number of attention heads for the query and key/value branches, respectively.

Let $W_Q \in \mathbb{R}^{H_Q \times D \times d_{\text{model}}}$, and $W_K \in \mathbb{R}^{H_{KV} \times D \times d_{\text{model}}}$ represent the learnable projection weights, where D is the head dimension. Given an input hidden state $H_{\text{in}} \in \mathbb{R}^{B \times S \times d_{\text{model}}}$, the query and key activations are computed via linear projections:

$$Q = H_{\text{in}} W_Q^\top, \quad K = H_{\text{in}} W_K^\top, \tag{1}$$

where $Q \in \mathbb{R}^{B \times S \times H_Q \times D}$ and $K \in \mathbb{R}^{B \times S \times H_{KV} \times D}$, with B denoting the batch size and S the sequence length.

To quantify activation patterns, we compute the ℓ_2-norm across the sequence dimension. We define the global magnitude matrix M as the expectation over the batch. Since the number of heads may differ for Q and K, we denote their magnitude matrices as $M^Q \in \mathbb{R}^{H_Q \times D}$ and $M^K \in \mathbb{R}^{H_{KV} \times D}$, respectively. For the query representation Q, the magnitude for head h at feature dimension d is given by:

$$M^Q_{h,d} = \frac{1}{B} \sum_{i=1}^{B} \sqrt{\sum_{s=1}^{S} (Q^{(i)}_{s,h,d})^2}. \tag{2}$$

A similar calculation applies to K using H_{KV} heads. As illustrated in Figure 1, high-magnitude activations (outliers) consistently appear in specific dimensions. Following insights from quantization (Lin et al., 2024; Jin et al., 2025), we identify the specific rows in W_Q and W_K corresponding to these outlier dimensions as the critical parameters for update.

Reinforcement Learning in Long-context Scenarios The standard reinforcement learning objective in language modeling seeks to optimize an expected reward, regularized by KL divergence. (Schulman et al., 2018):

$$\max_{\pi_\theta} \mathbb{E}_{x \sim \mathcal{D}, y \sim \pi_\theta(\cdot|x)} \left[r_\phi(x, y) \right] - \beta \mathbb{D}_{\text{KL}} \left[\pi_\theta(y \mid x) \,\|\, \pi_{\text{ref}}(y \mid x) \right] \tag{3}$$

where $r_\phi(x, y)$ denotes the reward for output y given input x from the policy model π_θ, and π_{ref} represents the reference model for $\mathbb{D}_{\text{KL}}$ regularization.

Unlike prior works that rely on the parametric knowledge of the policy model π_θ to generate an output y from a typically short question x, we extend the formulation by incorporating an additional long-context c. This requires π_θ to first ground relevant information in c before producing reasoning chains to solve x:

$$\max_{\pi_\theta} \mathbb{E}_{x,c \sim \mathcal{D}, y \sim \pi_\theta(\cdot|x,c)} \left[r_\phi(x, c, y) \right] - \beta \mathbb{D}_{\text{KL}} \left[\pi_\theta(y \mid x, c) \,\|\, \pi_{\text{ref}}(y \mid x, c) \right] \tag{4}$$

Reinforcement Learning with Variable Reward (RLVR) significantly enhances the reasoning capabilities of models. We build our training framework upon Group Relative Policy Optimization

(GRPO) (Shao et al., 2024), which eliminates the need for an external critic model by normalizing rewards within a group of outputs. Given a context c and question x, the old policy $\pi_{\theta_{\text{old}}}$ generates a group of G outputs $\{y_i\}_{i=1}^{G}$, with rewards $\{r_i\}_{i=1}^{G}$. The optimization objective is formulated as:

$$\mathcal{J}(\theta) = \mathbb{E}\Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\min\Big(\rho_{i,t}A_{i,t}, \text{clip}\left(\rho_{i,t}, 1-\varepsilon, 1+\varepsilon\right)A_{i,t}\Big) - \beta\mathbb{D}_{\text{KL}}\Bigg] \tag{5}$$

where the expectation is over $x, c \sim \mathcal{D}$ and $\{y_i\} \sim \pi_{\theta_{\text{old}}}$. Here, $\rho_{i,t} = \frac{\pi_\theta(y_{i,t}|x,c,y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t}|x,c,y_{i,<t})}$ denotes the probability ratio. Crucially, $A_{i,t}$ is the advantage term derived from group-relative normalization:

$$A_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^{G})}{\text{std}(\{r_j\}_{j=1}^{G})} \tag{6}$$

Since recent studies have identified the limitations of naive GRPO (Yu et al., 2025b; Zheng et al., 2025), we employ DAPO in our implementation for more stable training.

3.2 Supervised Fine-tuning (Cold Start)

The first stage of our pipeline is a Supervised Fine-tuning (SFT) phase, which initializes the base model with a robust policy prior to reinforcement learning. In this stage, we optimize the model parameters θ by minimizing the standard Cross-Entropy (CE) loss over the gold reasoning trajectories:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{SFT}}}\sum_{t=1}^{|y|}\log\pi_\theta(y_t \mid x, y_{<t}), \tag{7}$$

where x denotes the input context, $y = (y_1, \ldots, y_T)$ represents the target chain-of-thought sequence, and $\pi_\theta(y_t \mid x, y_{<t})$ is the probability of the t-th token given the context and preceding tokens. This phase ensures the model adapts to the specific output format (e.g., enclosing reasoning processes within `<think>` tags) required for the subsequent RL stage.

3.3 LongAct Training Framework

We propose a sparsity-aware training framework that dynamically adapts to the model's intrinsic activation patterns. Our approach consists of a rule-based reward mechanism and a saliency-guided parameter update strategy.

Dynamic Saliency-guided Updates. The core of LongAct is to restrict gradient updates to the "load-bearing" parameters identified in the Preliminary. We focus on the projection weights W_Q and W_K. Taking the Query as an example (the Key follows the same logic):

The query projection weight $W_Q \in \mathbb{R}^{H_Q \times D \times d_{\text{model}}}$ maps the hidden state to the concatenated head outputs. Structurally, W_Q is organized by heads, where the r-th **row** generates the specific feature dimension for a corresponding head. The mapping from a specific head $h \in \{0, \ldots, H_Q - 1\}$ and its internal dimension $d \in \{0, \ldots, D-1\}$ to the global row index j in W_Q is defined as:

$$j(h, d) = h \cdot D + d. \tag{8}$$

At each training step, we utilize the pre-computed global magnitude matrix $M^Q \in \mathbb{R}^{H_Q \times D}$ (Eq. 2). Instead of a global top-k selection, we perform intra-head selection to preserve the multi-head structure. For each head h, we identify the subset of critical local dimensions $\mathcal{K}_h$:

$$\mathcal{K}_h = \left\{ d \mid d \in \arg\max_k \{M^Q_{h,d'}\}_{d'=0}^{D-1} \right\}, \tag{9}$$

where $k = \lfloor \lambda D \rfloor$ is determined by the sparsity ratio λ (e.g., 0.3).

We then define a binary gradient mask $\mathbf{G}^Q \in \{0,1\}^{H_Q \times D \times d_{\text{model}}}$ for the weight matrix W_Q. A row r in W_Q is trainable if and only if it corresponds to a selected high-magnitude feature in its respective head:

$$\mathbf{G}^Q_{r,:} = \begin{cases} \mathbf{1} & \text{if } r \in \{j(h,d) \mid \forall h, d \in \mathcal{K}_h\} \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{10}$$

where $\mathbf{1}$ and $\mathbf{0}$ denote row vectors of size d_{model}. We provide an example in Appendix A.

During backward propagation, we apply these masks to the gradients:

$$\nabla W_Q \leftarrow \nabla W_Q \odot \mathbf{G}^Q, \quad \nabla W_K \leftarrow \nabla W_K \odot \mathbf{G}^K. \tag{11}$$

The computational overhead of generating the dynamic mask is negligible. The saliency calculation using Equation (2) and Top-k selection are performed on the collapsed head-dimension tensors ($H \times D$), not the full sequence, rendering the cost minimal compared to the full forward-backward pass.

It is important to emphasize that the sparse mask is applied **only to the gradients** of W_Q and W_K during training. All other parameters (i.e., W_V, W_O, and MLP layers) receive standard full updates. Detailed visualizations of these activation distributions are provided in Figures 3 to 5.During inference, the mask is discarded, and the model operates as a standard dense Transformer with zero additional latency or architectural modifications.

Reward Formulation. We employ a rule-based reward function $r(y)$ composed of a format reward and an answer reward:

$$r(y) = r_{\text{fmt}}(y) + r_{\text{ans}}(y). \tag{12}$$

Specifically, $r_{\text{fmt}}(y)$ is set to 1 if the response y contains the tags `<think>`, `</think>`, `<answer>`, and `</answer>`, and 0 otherwise. $r_{\text{ans}}(y)$ is set to 1 if the answer matches the ground truth, and 0 otherwise.

4 Experimental Results

4.1 Set Up

Training Setting We use Qwen3-8B-Base [1] as backbone. All experiments are conducted on 8 NVIDIA H800 80GB GPUs.

- **Supervised Fine-tuning** We conduct our training using LLaMA-Factory [2]. For the cold-start phase, we utilize 20k open-source instruction-following samples from AM-DeepSeek-R1-0528-Distilled [3] and we employ a cosine learning rate scheduler with a peak rate of 2e-5 and a warmup ratio of 0.03. We set the maximum sequence length to 16,384, with sequence packing and training for 900 steps.

- **Reinforcement Learning** We use verl [4] for RL and use DAPO (Yu et al., 2025b). We curate a training dataset by mixing DocQA-RL-1.6K [5] and randomly 1K samples from MemAgent (Yu et al., 2025a). We set the learning rate to 1e-6, with a maximum sequence length of 32,768, a batch size of 8, a rollout number of 16, a temperature of 1.0, and a maximum output length of 4096 for sampling. Unless otherwise specified, we set the default sparsity ratio λ for LongAct to **0.3** (i.e., updating only the top 30% of high-magnitude weights) based on our ablation studies.

Evaluation Setting We employ a comprehensive suite of long-context benchmarks to rigorously evaluate model performance:

- **LongBench v2** (Bai et al., 2025): A challenging benchmark focused on realistic long-context deeper understanding. It employs a multiple-choice question format to facilitate rigorous and objective evaluation.

- **RULER** (Hsieh et al., 2024): A benchmark designed to evaluate the effective context window size through diverse synthetic tasks.

- **InfiniteBench** (Zhang et al., 2024): A dataset targeting diverse long-context capabilities, covering heterogeneous tasks such as retrieval (Re.Pa, Re.Nu), summarization (En.Sum), QA (En.QA, Zh.QA), and multi-choice reasoning (En.MC).

For inference, we utilize vLLM [6] with a decoding temperature of 1.0. All evaluations are conducted using official scripts with a fixed random seed of 0 to ensure reproducibility.

Baselines We conduct comparisons using the `Qwen3` family at both 4B and 8B scales. The baselines include: (1) the officially released `Qwen3-Base` models; (2) our reproduced `Qwen3-SFT` models obtained from the Cold Start phase; and (3) the `Qwen3-SFT w/ DAPO` variants, which undergo standard full-parameter RL training.

4.2 Main Results

Results on LongBench v2. Table 1 summarizes our LongBench v2 results. Overall, `Qwen3-8B-SFT w/ LongAct` achieves the best performance at **36.73**, improving by **+3.93** over `Qwen3-8B-SFT w/ DAPO` (32.80) and **+9.69** over the cold-start `Qwen3-8B-SFT` model (27.04). Notably, our method also outperforms the officially released `Qwen3-8B*` by **+3.13** (33.60 → 36.73), indicating that `LongAct` provides gains beyond standard post-training recipes. We observe a consistent trend on the smaller model: `Qwen3-4B-SFT w/ LongAct` reaches **34.24**, yielding **+3.82** over `Qwen3-4B-SFT`

[1] https://huggingface.co/Qwen/Qwen3-8B-Base
[2] https://github.com/hiyouga/LLaMA-Factory
[3] https://huggingface.co/datasets/a-m-team/AM-DeepSeek-R1-0528-Distilled
[4] https://github.com/volcengine/verl
[5] https://huggingface.co/datasets/Tongyi-Zhiwen/DocQA-RL-1.6K
[6] https://github.com/vllm-project/vllm

Model	Overall	Difficulty		Length		
		Easy	Hard	Short	Medium	Long
Qwen3-8B*	33.60	39.58	29.90	39.44	28.37	34.26
Qwen3-8B-Base	20.68	19.27	21.54	28.33	17.67	13.89
Qwen3-8B-SFT	27.04	28.65	26.05	32.22	24.65	23.15
Qwen3-8B-SFT w/ DAPO	32.80	**40.10**	28.30	38.33	28.37	32.41
Qwen3-8B-SFT w/ LongAct	**36.73**	38.02	**35.93**	**41.94**	**33.37**	**34.72**
Qwen3-4B*	31.41	**34.90**	29.26	35.56	27.91	31.48
Qwen3-4B-Base	17.10	19.79	15.43	21.11	13.49	17.59
Qwen3-4B-SFT	25.65	21.35	28.30	33.33	22.33	19.44
Qwen3-4B-SFT w/ DAPO	30.42	28.12	31.83	33.33	31.16	24.07
Qwen3-4B-SFT w/ LongAct	**34.24**	33.07	**34.97**	**37.92**	**31.63**	**33.33**

Table 1: Evaluation results (%) on LongBench v2. Qwen3-8B* and Qwen3-4B* are officially released models available on HuggingFace. Short, Medium, and Long refer to length ranges of <32k, 32-128k, and >128k, respectively. The evaluation is conducted using the official code, and we have set the random seed to 0 to ensure reproducibility.

Model	Ruler-128k					Ruler-64k				
	NIAH-sub	NIAH	VT	QA	Avg	NIAH-sub	NIAH	VT	QA	Avg
Qwen3-8B-Base	30.63	51.88	47.68	17.60	36.95	37.93	61.76	28.08	39.70	41.87
Qwen3-8B-SFT	33.57	61.64	51.96	30.50	44.42	41.85	62.96	31.00	38.80	43.65
Qwen3-8B-SFT w/ DAPO	33.92	57.64	**76.56**	30.40	49.63	41.98	66.13	32.52	**40.40**	45.26
Qwen3-8B-SFT w/ LongAct	**34.55**	**63.70**	75.24	**31.10**	**51.15**	**45.97**	**69.64**	**34.96**	34.90	**46.37**
Qwen3-4B-Base	22.23	46.74	45.64	14.00	32.15	42.72	60.55	56.40	19.30	44.74
Qwen3-4B-SFT	26.60	53.02	**91.44**	27.20	49.57	40.92	66.20	90.92	20.20	54.56
Qwen3-4B-SFT w/ DAPO	31.08	54.57	90.48	28.70	51.21	42.60	67.74	**94.96**	30.90	59.05
Qwen3-4B-SFT w/ LongAct	**31.50**	**56.29**	90.64	**32.10**	**52.63**	**44.70**	**69.68**	94.52	**36.30**	**61.30**

Table 2: Evaluation results (%) on Ruler-128K and Ruler-64K. NIAH-sub is derived from the multi-key level 2, multi-key level 3 and multi-value tasks from NIAH.

w/ DAPO (30.42) and surpassing Qwen3-4B* by **+2.83** (31.41 → 34.24). These results suggest that our approach scales robustly across model sizes.

Difficulty Breakdown. LongAct excels on *hard* instances, which are highly sensitive to error accumulation. On the 8B model, it achieves **35.93**, substantially surpassing SFT w/ DAPO (**+7.63**) and SFT (**+9.88**). Similarly, the 4B model reaches **34.97**, outperforming all baselines. While DAPO performs well on easy splits, LongAct offers a more balanced profile, significantly lifting performance on hard tasks without compromising overall capability.

Length Breakdown. We further analyze robustness across input lengths. LongAct achieves the best results across *short/medium/long* categories. For 8B model, LongAct attains **41.94** (Short), **33.37** (Medium), and **34.72** (Long), with particularly strong gains compared to SFT w/ DAPO across input lengths (38.33 → 41.94 (**+3.61**), 28.37 → 33.37 (**+5.00**), 32.41 → 34.72 (**+2.31**) for Short, Medium and Long respectively). On 4B model, LongAct significantly improves the long split to **33.33**, which is **+9.26** over SFT w/ DAPO (24.07) and **+1.85** over Qwen3-4B* (31.48). These improvements indicate that LongAct enhances stability as context length increases, rather than overfitting to shorter-context behaviors.

4.3 Ablation on More Long-context Benchmarks

Results on RULER-128K and RULER-64K. Table 2 reports ablations on RULER under two context lengths (128K and 64K). Overall, LongAct consistently improves long-context performance across both model sizes. For 8B model, Qwen3-8B-SFT w/ LongAct achieves the best average score on RULER-128K (**51.15**), improving over Qwen3-8B-SFT (44.42) and Qwen3-8B-SFT w/ DAPO (49.63) by **+6.73** and **+1.52**, respectively. On

Methods	Re.Pa	Re.Nu	En.Sum	En.QA	Zh.QA	En.MC	Avg.
Qwen3-8B-SFT	85.76	**86.44**	14.18	25.09	14.47	54.15	46.68
Qwen3-8B-SFT w/ DAPO	85.93	83.56	11.86	**27.45**	**23.54**	56.33	48.11
Qwen3-8B-SFT w/ LongAct	**86.44**	85.76	**15.83**	27.33	21.61	**59.39**	**49.39**
Qwen3-4B-SFT	86.78	83.44	12.76	12.68	9.65	53.28	43.10
Qwen3-4B-SFT w/ DAPO	**87.63**	84.75	**17.78**	14.83	10.39	54.59	44.99
Qwen3-4B-SFT w/ LongAct	86.61	**86.95**	15.57	**21.06**	**11.61**	**56.33**	**46.36**

Table 3: Evaluation results (%) on InfiniteBench.

		Difficulty		Length		
Model	Overall	Easy	Hard	Short	Medium	Long
Qwen3-8B-SFT	27.04	28.65	26.05	32.22	24.65	23.15
+ DAPO	**36.73**	38.02	**35.93**	**41.94**	33.37	**34.72**
+ GRPO	35.04	**41.15**	31.27	37.92	**36.05**	28.24
+ CLIP-conv	35.04	40.76	31.51	37.08	34.42	32.87
+ KL-conv	34.24	37.37	32.32	40.00	32.44	28.24
Qwen3-4B-SFT	25.65	21.35	28.30	33.33	22.33	19.44
+ DAPO	**34.24**	33.07	34.97	**37.92**	31.63	33.33
+ GRPO	33.45	**34.64**	32.72	37.64	28.49	36.34
+ CLIP-conv	32.11	32.55	31.83	33.06	30.00	34.72
+ KL-conv	33.90	32.03	**35.05**	32.92	**33.02**	**37.27**

Table 4: Evaluation results (%) on LongBench v2 under different RL algorithms.

RULER-64K, LongAct also yields the highest performance with an average of **46.37**, outperforming Qwen3-8B-SFT (43.65) and Qwen3-8B-SFT w/ DAPO (45.26). We observe the same trend for the 4B model, where LongAct delivers the best overall averages at both 128K and 64K, suggesting that the gains are robust and scale across model sizes.

Results on InfiniteBench. Table 3 summarizes results on InfiniteBench. LongAct achieves the best average performance for both model sizes. For 8B, Qwen3-8B-SFT w/ LongAct attains the highest average score (**49.39**), improving over Qwen3-8B-SFT (46.68) by **+2.71** and over Qwen3-8B-SFT w/ DAPO (48.11) by **+1.28**. The gains are driven by consistent improvements on En.Sum (14.18 → 15.83) and En.MC (54.15 → 59.39), while remaining competitive on retrieval and QA tasks (e.g., Re.Pa: 85.76 → 86.44). For 4B, Qwen3-4B-SFT w/ LongAct also yields the best average (**46.36**), outperforming Qwen3-4B-SFT (43.10) and Qwen3-4B-SFT w/ DAPO (44.99) by **+3.26** and **+1.37**, respectively. Notably, LongAct substantially boosts long-context QA for smaller models (En.QA: 12.68 → 21.06; Zh.QA: 9.65 → 11.61) and improves En.MC (53.28 → 56.33). Overall, these results indicate that LongAct provides consistent and general improvements across heterogeneous long-context tasks.

4.4 Ablation on More Reinforcement Learning Methods

Table 4 evaluates our training recipe under different RL algorithms on top of the cold-start *SFT* models. Across both 4B and 8B models, applying RL consistently improves long-context performance, and our method maintains stable gains under all RL algorithms, including DAPO, GRPO, CLIP-conv, and KL-conv on LongBench v2. Among them, DAPO achieves the best overall results on both 8B and 4B and yields the most balanced improvements across difficulty and length splits, while other algorithms remain competitive but exhibit stronger trade-offs (e.g., favoring easier or medium-length subsets over the longest-context subset). These results suggest that the effectiveness of our method does not rely on a particular RL algorithm and is generalizable. Unless otherwise specified, we adopt DAPO as the

		Difficulty		Length		
Model	**Overall**	**Easy**	**Hard**	**Short**	**Medium**	**Long**
Qwen3-8B-SFT	27.04	28.65	26.05	32.22	24.65	23.15
+ random	28.63	30.21	27.65	28.33	30.70	25.00
+ min values	29.82	32.81	27.97	35.00	26.98	26.85
+ massive values	**36.73**	**38.02**	**35.93**	**41.94**	**33.37**	**34.72**
Qwen3-4B-SFT	27.04	28.65	26.05	32.22	24.65	23.15
+ random	29.03	30.21	28.30	30.00	28.84	27.78
+ min values	30.22	23.96	34.08	27.78	31.16	32.41
+ massive values	**34.24**	**33.07**	**34.97**	**37.92**	**31.63**	**33.33**

Table 5: Evaluation results (%) on LongBench v2. Ablation on activation selection strategies.

default in subsequent experiments.

4.5 Ablation on Selecting Activations

Table 5 evaluates different activation selection strategies during training. Across both backbones, selecting *massive values* consistently yields the best performance, delivering large gains over the SFT baseline and clearly outperforming *random* selection and *min values*. In contrast, *random* and *min values* provide only modest improvements and are less consistent across difficulty and length splits. These results suggest that focusing updates on salient (high-magnitude) activations is crucial for effectively improving long-context capability.

4.6 Ablation on the Sparsity in RL Parameter Updates

Table 6 studies how the update sparsity (i.e., the percentage of selected massive values) affects performance. We observe that using a moderate sparsity consistently yields the best overall results. For both 8B and 4B backbones, selecting 30% massive values achieves the highest overall score and the most balanced improvements across difficulty and length splits, especially on the *Hard* and *Long* subsets. Increasing the ratio to 40% does not further improve overall performance and can introduce regressions on some partitions, suggesting that overly dense updates may weaken the benefit of targeting the most salient parameters. Unless otherwise specified, we set the default sparsity ratio to 30% in subsequent experiments.

5 Analysis

To investigate the underlying mechanism of long-context capabilities, we conduct a perturbation analysis on Qwen3-8B using real-world cases from LongBench v2. As illustrated in Figure 6, to be specific, we isolate the impact of activation magnitude by selectively disrupting the top 30% ("Qwen3-8B w/o High-magnitude Activations") versus the bottom 30% ("Qwen3-8B w/o Non High-magnitude Activations") of activations. Specifically, the selected activations are clamped to the global mean value, calculated by averaging the entire query (or key) tensor across all heads, sequence lengths, and feature dimensions.

Comparing these responses, we observe that "Qwen3-8B w/o Non High-magnitude Activations" retains its logical coherence. As shown in Figure 6, the generated CoT maintains a structured flow—correctly utilizing logical connectors like "Alternatively" and "However"—and successfully derives the correct answer. This suggests that the core reasoning process is less dependent on these "quiet" activations. Conversely, disrupting the top 30% of high-magnitude activations leads to immediate model collapse. The output degrades into repetitive loops (e.g., the "3333..." pattern visible in Figure 6).

The stark contrast between these outcomes underscores that high-magnitude activations are pivotal for maintaining reasoning in long-context scenarios. This observation corroborates prior findings (Lin et al., 2024; Liu et al., 2024; Jin et al., 2025). Crucially, our empirical results extend this insight, demonstrating that leveraging these critical activations to guide the training process is highly effective for enhancing long-context capabilities.

6 Conclusion

We propose LongAct, a robust method that leverages the model's internal representations to enhance long-context reasoning.

Limitations

Limited by computing resources, we could not use larger models for reinforcement learning. We will explore the scaling effects of our approach in future work.

References

Arash Ahmadian, Saurabh Dash, Hongyu Chen, Bharat Venkitesh, Zhen Stephen Gou, Phil Blunsom, Ahmet Üstün, and Sara Hooker. 2023. Intriguing properties of quantization at scale. *Advances in Neural Information Processing Systems*, 36:34278–34294.

Anonymous. 2025. Longrlvr: Overcoming the long-context bottleneck in reinforcement learning with verifiable rewards. OpenReview. Under review as a conference paper at ICLR 2026.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, and 1 others. 2025. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3639–3664.

Federico Barbero, Alex Vitvitskyi, Christos Perivolaropoulos, Razvan Pascanu, and Petar Veličković. 2024. Round and round we go! what makes rotary positional encodings useful? *arXiv preprint arXiv:2410.06205*.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, and 1 others. 2025. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*.

Jingcheng Deng, Liang Pang, Zihao Wei, Shichen Xu, Zenghao Duan, Kun Xu, Yang Song, Huawei Shen, and Xueqi Cheng. 2025. Latent reasoning in llms as a vocabulary-space superposition. *Preprint*, arXiv:2510.15522.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Advances in neural information processing systems*, 35:30318–30332.

Yizhao Gao, Shuming Guo, Shijie Cao, Yuqing Xia, Yu Cheng, Lei Wang, Lingxiao Ma, Yutao Sun, Tianzhu Ye, Li Dong, and 1 others. 2025. Seerattention-r: Sparse attention adaptation for long reasoning. *arXiv preprint arXiv:2506.08889*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Tianyu Guo, Druv Pai, Yu Bai, Jiantao Jiao, Michael I Jordan, and Song Mei. 2024. Active-dormant attention heads: Mechanistically demystifying extreme-token phenomena in llms. *arXiv preprint arXiv:2410.13835*.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. 2024. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. Ruler: What's the real context size of your long-context language models? *arXiv preprint arXiv:2404.06654*.

Mingyu Jin, Kai Mei, Wujiang Xu, Mingjie Sun, Ruixiang Tang, Mengnan Du, Zirui Liu, and Yongfeng Zhang. 2025. Massive values in self-attention modules are the key to contextual knowledge understanding. *arXiv preprint arXiv:2502.01563*.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of machine learning and systems*, 6:87–100.

Zirui Liu, Jiayi Yuan, Hongye Jin, Shaochen Zhong, Zhaozhuo Xu, Vladimir Braverman, Beidi Chen, and Xia Hu. 2024. Kivi: A tuning-free asymmetric 2bit quantization for kv cache. *arXiv preprint arXiv:2402.02750*.

QwenTeam. 2025. Qwen3-next: Towards ultimate training & inference efficiency. https://qwen.ai/blog?id=4074cca80393150c248e508aa62983f9cb7d27cd. Accessed: 2025-10.

John Schulman, Xi Chen, and Pieter Abbeel. 2018. Equivalence between policy gradients and soft q-learning. *Preprint*, arXiv:1704.06440.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, and 1 others. 2025a. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*.

Kimi Team, Yu Zhang, Zongyu Lin, Xingcheng Yao, Jiaxi Hu, Fanqing Meng, Chengyin Liu, Xin Men, Songlin Yang, Zhiyuan Li, and 1 others. 2025b. Kimi linear: An expressive, efficient attention architecture. *arXiv preprint arXiv:2510.26692*.

MiniCPM Team, Chaojun Xiao, Yuxuan Li, Xu Han, Yuzhuo Bai, Jie Cai, Haotian Chen, Wentong Chen, Xin Cong, Ganqu Cui, and 1 others. 2025c. Minicpm4: Ultra-efficient llms on end devices. *arXiv preprint arXiv:2506.07900*.

Fanqi Wan, Weizhou Shen, Shengyi Liao, Yingcheng Shi, Chenliang Li, Ziyi Yang, Ji Zhang, Fei Huang, Jingren Zhou, and Ming Yan. 2025. Qwenlong-l1: Towards long-context large reasoning models with reinforcement learning. *arXiv preprint arXiv:2505.17667*.

Siyuan Wang, Gaokai Zhang, Li Lyna Zhang, Ning Shang, Fan Yang, Dongyao Chen, and Mao Yang. 2025. Loongrl: Reinforcement learning for advanced reasoning over long contexts. *arXiv preprint arXiv:2510.19363*.

Chaojun Xiao, Pengle Zhang, Xu Han, Guangxuan Xiao, Yankai Lin, Zhengyan Zhang, Zhiyuan Liu, and Maosong Sun. 2024. Infllm: Training-free long-context extrapolation for llms with an efficient context memory. *Advances in Neural Information Processing Systems*, 37:119638–119661.

Wujiang Xu, Qitian Wu, Zujie Liang, Jiaojiao Han, Xuying Ning, Yunxiao Shi, Wenfang Lin, and Yongfeng Zhang. 2024. Slmrec: Distilling large language models into small for sequential recommendation. *arXiv preprint arXiv:2405.17890*.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, and 1 others. 2025a. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent. *arXiv preprint arXiv:2507.02259*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, and 1 others. 2025b. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, and 1 others. 2025. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models. *arXiv preprint arXiv:2508.06471*.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Hao, Xu Han, Zhen Thai, Shuo Wang, Zhiyuan Liu, and 1 others. 2024. ∞ bench: Extending long context evaluation beyond 100k tokens. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15262–15277.

Weilin Zhao, Zihan Zhou, Zhou Su, Chaojun Xiao, Yuxuan Li, Yanghao Li, Yudi Zhang, Weilun Zhao, Zhen Li, Yuxiang Huang, and 1 others. 2025. Infllm-v2: Dense-sparse switchable attention for seamless short-to-long adaptation. *arXiv preprint arXiv:2509.24663*.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, and 1 others. 2025. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*.

A Appendix

Illustrative Example. To clarify the indexing mechanism, consider a simplified configuration with $H = 2$ heads, head dimension $D = 4$, and a selection ratio $\lambda = 0.3$, resulting in $k = \lfloor 0.3 \times 4 \rfloor = 1$ active dimension per head. Suppose the computed importance scores for the two heads are:

$$\mathbf{M}_0 = [0.8, 0.2, \mathbf{0.9}, 0.5] \quad \rightarrow \quad \text{idx}_0 = 2$$
$$\mathbf{M}_1 = [0.3, \mathbf{0.7}, 0.6, 0.4] \quad \rightarrow \quad \text{idx}_1 = 1$$

The system then maps these local head indices to the global row indices of the projection weight $\mathbf{W} \in \mathbb{R}^{(H \cdot D) \times d_{in}}$:

- **Head 0:** Global Row $g_0 = 0 \times 4 + 2 = \mathbf{2}$.
- **Head 1:** Global Row $g_1 = 1 \times 4 + 1 = \mathbf{5}$.

Consequently, only rows $\{2, 5\}$ of $\mathbf{W}$ (2 out of 8 total rows) are updated, while the rest remain frozen.

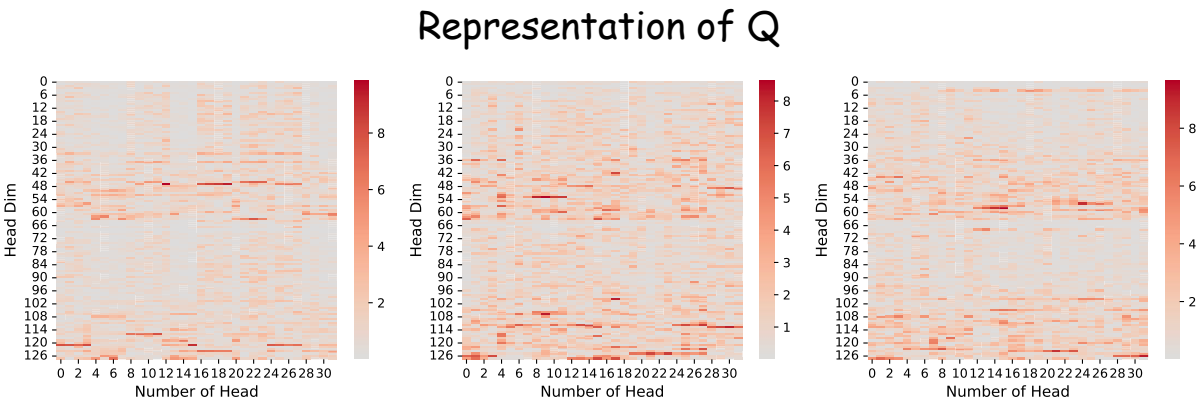

Figure 3: Visualization of the query representation magnitudes in Qwen3-8B on the RULER benchmark.

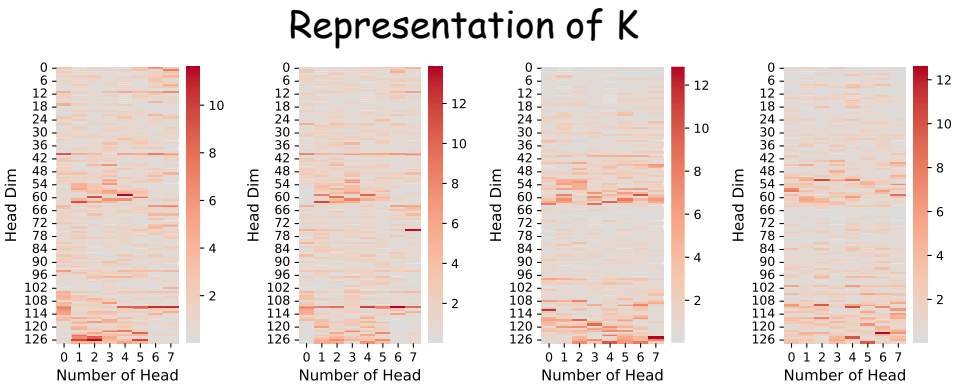

Figure 4: Visualization of the key representation magnitudes in Qwen3-8B on the RULER benchmark.

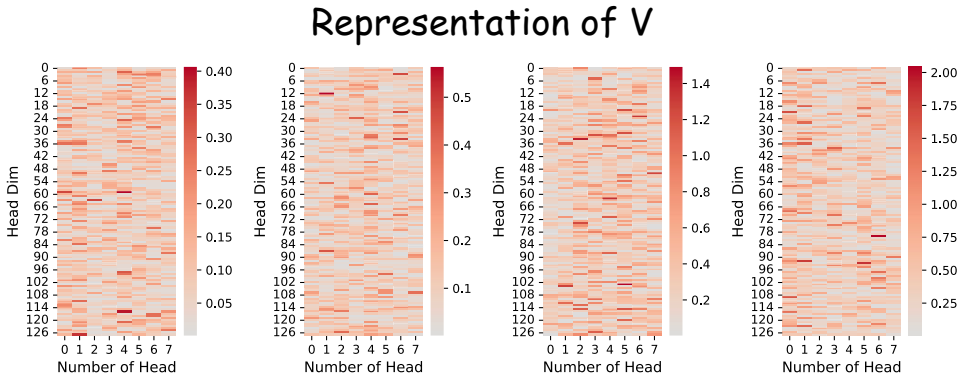

Figure 5: Visualization of the value representation magnitudes in Qwen3-8B on the RULER benchmark.

		Difficulty		Length		
Model	**Overall**	**Easy**	**Hard**	**Short**	**Medium**	**Long**
Qwen3-8B-SFT	27.04	28.65	26.05	32.22	24.65	23.15
+ 20% massive values	32.41	39.06	28.30	39.44	30.23	25.00
+ 30% massive values	**36.73**	38.02	**35.93**	**41.94**	33.37	**34.72**
+ 40% massive values	35.98	**40.62**	33.12	38.33	**34.88**	34.26
Qwen3-4B-SFT	27.04	28.65	26.05	32.22	24.65	23.15
+ 20% massive values	30.42	30.21	30.55	29.44	30.23	32.41
+ 30% massive values	**34.24**	**33.07**	**34.97**	37.92	**31.63**	**33.33**
+ 40% massive values	32.80	31.77	33.44	**41.11**	28.37	27.78

Table 6: Ablation on sparsity (percentage of selected massive values). Results are reported in %. Bold indicates the best value within each model (8B / 4B) for each column.

Question: Which of the following suggestions for the new year's initiatives is correct?

Qwen3-8B:
<think>\nOkay, let's tackle this question. The user is asking which of the given choices . . . . . \n\n First, I need to recall. . . . . but the option says looseness, which could be incorrect. Therefore, the correct answer should be D.\n</think>\n\nThe correct answer is (D).

Qwen3-8B w/o Non High-magnitude Activations:
<think>\nOkay, let's tackle this question. The user provided a text. . . . . \n\nFirst, I need to understand. . . . . \n\nAlternatively, the mention of coordination. . . . . However, the original text doesn't. . . . . So option D could be correct\n</think>\n\nThe correct answer is (D).

Qwen3-8B w/o High-magnitude Activations:
333333. 3333333333333

Figure 6: This is an example taken from the LongBench v2 dataset with the ID 66ec41d3821e116aacb1c874, whose ground truth is **D**. The context is too long to be displayed in the image. "Qwen3-8B w/o High-magnitude Activations" refers to disrupting the top 30% of high-magnitude activations, while "Qwen3-8B w/o Non High-magnitude Activations" refers to disrupting the smallest 30% of activations. Numerous cases in LongBench v2 exhibit similar patterns.

Setting	**Correct / Total**
Qwen3-8B	173 / 503
Qwen3-8B w/o Non High-magnitude Activations	108 / 503
Qwen3-8B w/o High-magnitude Activations	0 / 503

Table 7: Accuracy of Qwen3-8B on LongBench v2. We clamp selected activations to the global mean and report the number of correct predictions.