RANKING AND PARTITIONING OF ATTENTION HEADS FOR SPARSE AND CONTINUAL REINFORCEMENT LEARNING FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

We study sparse and continual reinforcement learning fine-tuning of language models through attention-head ranking and partitioning. For sparse learning, we rank attention heads by gradient norm to identify a small subset of heads that can be trained effectively. For continual learning, we partition heads across tasks to reduce cross-task interference and preserve previously learned capabilities. Theoretically, we relate gradient norm to training effectiveness, and further characterize why sparse, separated head updates can remain stable. Empirically, across math, instruction-following, and coding tasks, tuning higher-ranked heads consistently outperforms tuning lower-ranked while often matching denser baselines, and head partitioning improves retention while preserving adaptation to new tasks.

1 INTRODUCTION

Reinforcement-learning (RL) fine-tuning has become an important approach for improving the capabilities of language models (LMs). However, RL fine-tuning is expensive, especially when a large fraction of model parameters is updated, which increases gradient memory and synchronization costs in distributed training (Miahi & Belilovsky, 2026; Fireworks.ai, 2026). Recent works suggest that effective adaptation can often be achieved by updating only restricted parts of the LM, such as attention modules or individual transformer layers (Jin et al., 2025; Zhang et al., 2026). This motivates sparse RL fine-tuning under a limited trainable-parameter budget.

In this work, we focus on attention heads as the basic units for sparse RL fine-tuning. Classical pruning studies show that many attention heads can be removed with little performance degradation, while a smaller subset plays more important roles (Michel et al., 2019; Voita et al., 2019). More recent studies further suggest that attention heads can specialize toward different latent signals during training (Sagitova et al., 2026), and that high-level LM capabilities can be localized to sparse, task-specific subsets of heads (Bair et al., 2026). This motivates us to design a criterion to identify those heads that are most effective to update during training.

Sparse RL fine-tuning is also appealing for continual learning, where sequential updates can interfere with previously learned capabilities and lead to catastrophic forgetting. A common approach is to constrain different tasks to parameter or gradient subspaces (Chaudhry et al., 2020; Doan et al., 2020; Wang et al., 2023; Liang & Li, 2024; Qiu et al., 2026). Sparse updates can therefore serve to separate task-specific adaptations. Such parameter isolation avoids direct overwriting, yet indirect functional interference may remain (Lanzillotta et al., 2023), requiring careful selection of the trainable parameters such as Iurada et al. (2025); Lin et al. (2025; 2026).

Contributions. We study attention-head ranking and partitioning for sparse and continual RL fine-tuning. Our main contributions are summarized as follows:

- We propose gradient norm as a simple criterion for ranking attention heads. Across several tasks, tuning higher-ranked heads consistently outperforms tuning lower-ranked heads, while often matching or exceeding denser baselines.
- We propose partitioning attention heads across sequential tasks and study several head-allocation strategies. Our experiments show that head partitioning with these strategies effectively prevents forgetting while preserving adaptation to new tasks.

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

- We provide a theoretical analysis to guide both designs. We characterize the role of gradient norm in the optimization dynamics of sparse and continual learning, and further analyze the second-order structure of multi-head self-attention, revealing favorable properties of attention-head blocks for parameter and functional isolation.

2 Related Works

Sparse optimization in deep learning and LMs. Sparse optimization has been studied through subset selection (Dan et al., 2019), sparse reward-model structure (Yao et al., 2025; Shi et al., 2026), and sparse task-specific parameter updates (Guo et al., 2021). For LMs, recent works show that updating only attention modules (Jin et al., 2025) or a single layer (Zhang et al., 2026) can obtain competitive performance, while sparse updates can improve communication efficiency in distributed RL (Miahi & Belilovsky, 2026; Fireworks.ai, 2026).

Subspace continual learning. A common approach to continual learning is to reduce interference by constraining new-task updates to subspaces that avoid directions important for previous tasks. Early methods construct such subspaces from past gradients or representations (Farajtabar et al., 2020; Saha et al., 2021; Chaudhry et al., 2020; Doan et al., 2020). Recent works extend this idea to LoRA adaptation (Wang et al., 2023; Liang & Li, 2024; Zhang et al., 2025; Qiu et al., 2026), full fine-tuning (Nayak et al., 2026), and model merging (Qiao & Mahdavi, 2026; Tang et al., 2025; Yang et al., 2025). Iurada et al. (2025) rank parameters by sensitivity measured using Fisher matrix, and fine-tune only the least-sensitive ones. Lin et al. (2025) update only parameters that are activated more strongly by new data than by pretraining data, reducing interference with prior knowledge. Lin et al. (2026) dynamically update parameter-level masks based on accumulated gradient magnitudes, whereas we use static head-level gradient norms.

Theoretical understanding of continual learning. Neural tangent kernel (NTK)-based analyses relate catastrophic forgetting to overlap between task-induced tangent subspaces (Doan et al., 2020; Zheng et al., 2026), while kernel-based views have also been developed for LM fine-tuning (Malladi et al., 2023). Parameter isolation has been studied through Hessian-based arguments (Lanzillotta et al., 2023). In RL, Luo et al. (2026) characterize loss of plasticity through degradation of spectral plasticity, while Shenfeld et al. (2026) show that online RL tends to forget less by staying closer to the base policy, and Chen et al. (2026) attribute this advantage to the use of on-policy data. We connect the stability–plasticity trade-off in continual learning directly to gradient norms.

Mechanism of attention heads. Prior work has shown that attention heads are highly heterogeneous, with many heads being redundant while a smaller subset exhibits specialized behavior (Michel et al., 2019; Voita et al., 2019). More directly related to this work, Bair et al. (2026) show that small subsets of attention heads can be critical to performance on particular LM capabilities. Sagitova et al. (2026) theoretically characterize how attention heads specialize toward distinct latent signals during training. Ormaniec et al. (2025) further analyzes the second-order structure of self-attention. Together, these results motivate us to focus on attention heads as structured units.

3 Preliminaries

Notations. Let $[H] := \{1, \dots, H\}$ index the attention heads. We partition the trainable attention parameters into disjoint head blocks $\theta = (\theta_1, \dots, \theta_H)$, where θ_h denotes the parameters associated with head h. For $S \subseteq [H]$, let $\theta_S := (\theta_h)_{h \in S}$ and $S^c := [H] \backslash S$. More generally, we write x_h and x_S for the corresponding vector blocks, and $D_{hh'}$ and D_{ST} for the corresponding matrix blocks, for $h, h' \in [H], S, T \subseteq [H]$. For an update $\Delta = (\Delta_1, \dots, \Delta_H)$, define its head support by $\mathrm{supp}_{\mathrm{head}}(\Delta) := \{h \in [H] : \|\Delta_h\|_2 > 0\}$. Let $\{\xi_i\}_{i=1}^N$ denote the training data and $\mathcal{L}(\theta) := \frac{1}{N} \sum_{i=1}^N \mathcal{L}(\theta; \xi_i)$ the empirical training loss. In the RL fine-tuning setting, each ξ_i is a prompt, and $\mathcal{L}(\theta; \xi_i)$ denotes the corresponding empirical RL loss induced by rollouts from that prompt. We use $\|\cdot\|_2$ for the Euclidean norm and $\|A\|_{\mathrm{op}}$ for the operator norm of a matrix A.

Gradient norm. Given a model with parameters θ, we define the gradient norm of a head set S as

$$\mathcal{G}(S; \theta) := \|\nabla_{\theta_S} \mathcal{L}(\theta)\|_2 = \sqrt{\sum_{h \in S} \|\nabla_{\theta_h} \mathcal{L}(\theta)\|_2^2} \, .$$

For two tasks A and B, we write $\mathcal{G}_A(S;\theta) := \|\nabla_{\theta_S}\mathcal{L}_A(\theta)\|_2$ and $\mathcal{G}_B(S;\theta) := \|\nabla_{\theta_S}\mathcal{L}_B(\theta)\|_2$. We assume that these losses are twice continuously differentiable on the relevant training regions.

Second-order terms of loss Hessian. We study the directional second-order derivative:

$$\frac{\left|\nabla_\theta\mathcal{L}(\theta)^\top\nabla^2_\theta\mathcal{L}_(\theta)\nabla_\theta\mathcal{L}(\theta)\right|}{\|\nabla_\theta\mathcal{L}(\theta)\|_2^2},$$

which measures how rapidly the gradient norm changes when moving along its own normalized gradient direction. For two objectives $\mathcal{L}_A$ and $\mathcal{L}_B$, we also study the mixed directional second-order derivative:

$$\frac{\left|\nabla_\theta\mathcal{L}_A(\theta)^\top\nabla^2_\theta\mathcal{L}_A(\theta)\nabla_\theta\mathcal{L}_B(\theta)\right|}{\|\nabla_\theta\mathcal{L}_A(\theta)\|_2\|\nabla_\theta\mathcal{L}_B(\theta)\|_2},$$

which measures how rapidly the task-A gradient norm changes when moving along the normalized task-B gradient direction.

Multi-head self-attention. Consider a multi-head self-attention layer with input $X \in \mathbb{R}^{L\times d}$ and H heads. For head $h \in [H]$, let $W_h^Q, W_h^K \in \mathbb{R}^{d\times d_k}$ and $W_h^V \in \mathbb{R}^{d\times d}$, and define

$$Q_h = XW_h^Q, \quad K_h = XW_h^K, \quad V_h = XW_h^V, \quad A_h = \mathrm{softmax}\left(\frac{Q_hK_h^\top}{\sqrt{d_k}}\right).$$

The layer output is $F(\theta) = \sum_{h=1}^H A_hV_h \in \mathbb{R}^{L\times d}$.

4 Ranking the Attention Heads

4.1 Theoretical understandings

We now study gradient-norm ranking for attention heads. Given the base model θ_0, let $\mathcal{G}(S) := \mathcal{G}(S;\theta_0)$ denote the gradient norm defined in Section 3, and let S_k be the set of the k heads with largest individual gradient norms at θ_0. Omitted proofs are deferred to Appendix A.1. We first characterize what this quantity measures under gradient flow.

Proposition 1. *Let $\theta(t;S)$ follow Euclidean gradient flow restricted to S, initialized at θ_0, and suppose the flow exists on $[0,T]$. Then $-\frac{d}{dt}\mathcal{L}(\theta(t;S)) = \mathcal{G}(S;\theta(t))^2$.*

Proof. The gradient flow satisfies $\dot\theta_S(t;S) = -\nabla_{\theta_S}\mathcal{L}(\theta(t;S))$ and $\dot\theta_{S^c}(t;S) = 0$, so the chain rule gives $\frac{d}{dt}\mathcal{L}(\theta(t;S)) = -\|\nabla_{\theta_S}\mathcal{L}(\theta(t;S))\|_2^2 = -\mathcal{G}(S;\theta(t))^2$. □

Corollary 2. *We have $S_k \in \arg\max_{|S|\leqslant k}\left\{-\frac{d}{dt}\mathcal{L}(\theta(t;S))\big|_{t=0}\right\}$, which maximizes the initial loss decrease rate.*

Corollary 3. *For any head set S and any $t > 0$ for which the restricted flow exists, define the time-averaged gradient norm by $\overline{\mathcal{G}}(S;t) := \sqrt{\frac{1}{t}\int_0^t \mathcal{G}(S;\theta(s;S))^2\,ds}$. Then we have*

$$\mathcal{L}(\theta_0) - \mathcal{L}(\theta(t;S_k)) \geqslant \frac{\overline{\mathcal{G}}(S_k;t)^2}{\max_{|S|\leqslant k}\overline{\mathcal{G}}(S;t)^2}\max_{|S|\leqslant k}\left[\mathcal{L}(\theta_0) - \mathcal{L}(\theta(t;S))\right].$$

Remark 1. This simple gradient-flow analysis shows that gradient norm characterizes training efficiency under a fixed head budget. It determines both the initial loss-decrease rate and finite-time progress through the time-averaged gradient norm.

Proposition 4. *Suppose that, for some $\beta > 0$,*

$$\frac{\left|(\nabla_{\theta_S}\mathcal{L}(\theta(t;S)))^\top\nabla^2_{\theta_S}\mathcal{L}(\theta(t;S))\nabla_{\theta_S}\mathcal{L}(\theta(t;S))\right|}{\|\nabla_{\theta_S}\mathcal{L}(\theta(t;S))\|_2^2} \leqslant \beta$$

for all $t \in [0,T]$ with $\mathcal{G}(S;\theta(t;S)) > 0$. Then, for every $t \in (0,T]$,

$$\sqrt{\frac{1-e^{-2\beta t}}{2\beta t}}\,\mathcal{G}(S) \leqslant \overline{\mathcal{G}}(S;t) \leqslant \sqrt{\frac{e^{2\beta t}-1}{2\beta t}}\,\mathcal{G}(S).$$

Remark 2. Under the second-order condition, the time-averaged gradient norm remains controlled by its initial value, so the ranking need not be recomputed during training as in Lin et al. (2026). In our experiments in Section 4.2, this ranking remains instructive for hundreds of training steps. See Section 6 for discussions on why the second-order assumption on attention blocks is reasonable.

Adam's perspective. We now give a complementary optimizer-level interpretation based on Adam. Let m and v denote the moment estimates used in the Adam update at θ_0, and let

$$D := \mathrm{diag}(\sqrt{v} + \epsilon_{\mathrm{adam}}) \succ 0, \qquad \eta > 0 \ ,$$

where $\epsilon_{\mathrm{adam}} > 0$. The Adam step $\Delta^{\mathrm{adam}} = -\eta D^{-1} m$ minimizes the objective (Duchi et al., 2011):

$$\mathcal{L}_{\mathrm{adam}}(\Delta) := m^\top \Delta + \frac{1}{2\eta} \Delta^\top D \Delta \ ,$$

where η is the learning rate. To induce head-level sparsity, we add a group-lasso penalty and consider

$$\mathcal{L}_{\mathrm{adam}_\lambda}(\Delta) := m^\top \Delta + \frac{1}{2\eta} \Delta^\top D \Delta + \lambda \sum_{h=1}^{H} \|\Delta_h\|_2 \ , \qquad \lambda \geqslant 0 \ .$$

Proposition 5. *The unique minimizer $\Delta^\star(\lambda)$ of $\mathcal{L}_{\mathrm{adam}_\lambda}(\Delta)$ satisfies that $\exists k$, s.t.*

$$\mathrm{supp}_{\mathrm{head}}(\Delta^\star(\lambda)) = \arg \operatorname*{TopK}_h \left(\|m_h\|_2^2 \right) \ .$$

Proposition 6. *Define $\Delta_S^\star$ as a minimizer of $\mathcal{L}_{\mathrm{adam}}(\Delta)$ w.r.t. $\mathrm{supp}_{\mathrm{head}}(\Delta) \subseteq S$, and $S_k^{(m)}$ be the set of the k heads with largest $\|m_h\|_2$. Then we have*

$$\mathcal{L}_{\mathrm{adam}}(\Delta^\star_{S_k^{(m)}}) \leqslant \min_{|S| \leqslant k} \frac{\min_{i \in \theta_S} \sqrt{v_i} + \epsilon_{\mathrm{adam}}}{\max_{i \in \theta_{S_k^{(m)}}} \sqrt{v_i} + \epsilon_{\mathrm{adam}}} \cdot \mathcal{L}_{\mathrm{adam}}(\Delta_S^\star) \ .$$

Remark 3 (Connection to practical optimization)**.** Proposition 5 shows that a group-sparse Adam update selects heads according to the first-moment magnitudes $\|m_h\|_2$, while Proposition 6 shows that the top-k heads under this ranking approximately optimize Adam's local quadratic objective. Since m_h is the exponential moving average of past gradients, the gradient norm $\mathcal{G}(\{h\}) = \|\nabla_{\theta_h} \mathcal{L}(\theta_0)\|_2$ provides a natural surrogate for $\|m_h\|_2$. This provides an optimizer-level motivation beyond the gradient-flow analysis.

4.2 Head-sparse learning experiments

Experimental setup. We evaluate head-sparse RL fine-tuning on Math, Instruction-following (IF), and Coding using **Qwen3-1.7B**, **Qwen3-4B**, and **Mistral-7B-Instruct-v0.3**. For each task, we rank attention heads by gradient norms at the initial checkpoint and compare training the top 50% and bottom 50% heads, together with full-attention tuning and full tuning baselines. We also include random 50% attention-head tuning, where 50% head blocks are randomly selected per layer, as a baseline. All training jobs share the same training configuration, except that, given the substantially larger number of trainable parameters, we use smaller learning rates for full tuning of the **Qwen3** series and for experiments on **Mistral-7B-Instruct-v0.3**. Implementation details are provided in Appendix C.

Main results. Tables 1 and 2 show a consistent advantage of gradient-based head selection. Top-50% tuning outperforms bottom-50% tuning across all model–task pairs and evaluation splits, and also generally outperforms the random-50% baseline. While random selection occasionally performs better on some tasks for **Qwen3-1.7B**, the advantage of top-ranked heads is more robust across various settings. Despite updating only half of the attention heads, top-50% tuning often matches or exceeds full-attention and full tuning, with the same trend extending to **Mistral-7B-Instruct-v0.3**. As shown in Table 10 in Appendix B.2, head-sparse tuning also reduces the gradients to be stored and synchronized at each step, offering a significant communication-efficiency advantage in distributed training Fireworks.ai (2026); Miahi & Belilovsky (2026). These results support gradient norm as an effective criterion for identifying heads useful for sparse RL tuning, consistent with the theoretical claims in Section 4.1. See supplemental results in Appendix B.1.

Comments on learning rates. Readers may wonder whether the weaker bottom 50% results simply arise because their smaller gradient scales require larger learning rates. We note that, under the fixed training configuration and the same trainable-parameter budget, this already reflects a nontrivial training-efficiency gap; moreover, if the difference were only a uniform gradient rescaling, Adam's normalization would cancel this effect. We further sweep the learning rate for bottom 50% attention-head tuning in Appendix B.4. It reveals that larger learning rates help in some cases, but the gap to top 50% attention-head tuning generally remains.

Head-ranking consistency. Table 11 in Appendix B.2 reports Spearman distances between independently computed head rankings. Within-task distances are consistently small, indicating stable rankings across subsets of the same task. Cross-task distances are larger, reflecting task-specific differences in head importance, but remain small overall, suggesting a shared intrinsic hierarchy across tasks. In practice, this consistency also suggests that, when the training set is large, head rankings can be estimated from a small subset of the data to reduce the computation cost.

Layer-ranking consistency. We further compare gradient-norm ranking with the candidate-layer rankings obtained by direct performance evaluation. As shown in Table 3, the two rankings are highly consistent across the **QWEN3-BASE** series, and exactly match for **QWEN3-4B-BASE**. This provides additional evidence that gradient norm can serve as a useful proxy for identifying trainable attention components without exhaustively evaluating each candidate. The layer distributions of top heads are visualized in Appenix B.3.

Sparsity–performance trade-off. Beyond the fixed 50% sparsity setting, we further vary the number of trainable heads to examine how performance changes with the head budget. All configurations use the same training setup, differing only in the number of trainable attention heads. Figure 1 shows that performance generally improves as more top-ranked heads are included, while the major performance can already be achieved with a relatively small head budget.

Table 1: Sparse RL accuracy (%) of **QWEN3** series. Parenthesized arrows are the performance drops compared with top 50%. The top-2 results are highlighted in pink. The IF performance of **QWEN3-4B** top/bottom 50% is reported as mean$\pm$std over three random rollout-seeds $\{32, 42, 52\}$.

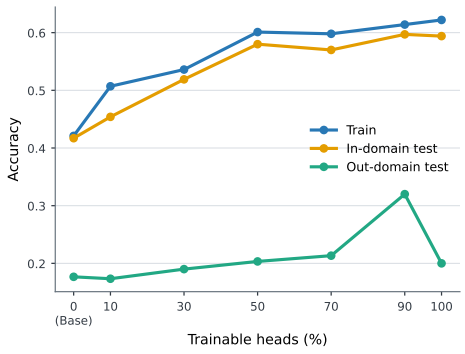
Model	Task	Eval split	Base	Top 50%	Bottom 50%	Rand 50%	Full-attn	Full
QWEN3-1.7B	Math	In-dist test	26.6	37.5	30.9 ($\downarrow$ 6.6)	38.6	33.1	35.3
		Out-dist test	70.2	75.6	73.0 ($\downarrow$ 2.6)	79.2	74.2	73.6
QWEN3-1.7B	IF	In-dist test	41.7	58.0	46.9 ($\downarrow$ 11.1)	51.6	59.4	54.6
		Out-dist test	17.7	20.3	15.0 ($\downarrow$ 5.3)	19.7	20.0	19.7
QWEN3-1.7B	Coding	In-dist test	53.7	58.5	56.9 ($\downarrow$ 1.6)	58.5	58.2	61.6
		Out-dist test	33.6	35.3	33.4 ($\downarrow$ 1.9)	35.6	34.8	34.4
QWEN3-4B	Math	In-dist test	42.4	59.3	53.6 ($\downarrow$ 5.7)	53.3	55.3	58.2
		Out-dist test	81.4	88.4	84.8 ($\downarrow$ 3.6)	83.6	86.0	88.2
QWEN3-4B	IF	In-dist test	50.5	60.7 $\pm$3.8	52.5 $\pm$1.0 ($\downarrow$ 8.2)	56.3	56.2	59.2
		Out-dist test	21.0	25.6 $\pm$1.3	25.4 $\pm$1.0 ($\downarrow$ 0.1)	25.3	26.0	24.0
QWEN3-4B	Coding	In-dist test	64.6	69.1	66.4 ($\downarrow$ 2.7)	66.7	65.1	66.4
		Out-dist test	46.3	51.4	43.1	37.9	47.9	43.6

Table 2: Sparse RL accuracy (%) of **MISTRAL-7B-INSTRUCT-V0.3**. Parenthesized arrows are the performance drops compared with top 50%. The highest results are highlighted in pink.

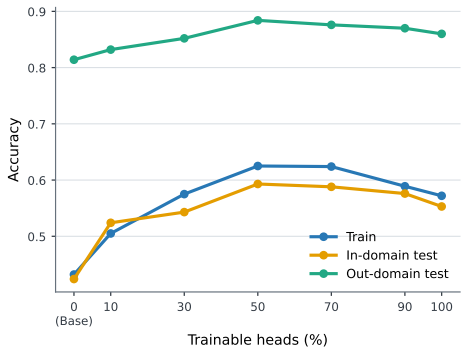
Model	Task	Eval split	Base	Top 50%	Bottom 50%	Full-attn
MISTRAL-7B-INSTRUCT-V0.3	IF	In-dist test	27.4	69.9	53.5 ($\downarrow$ 16.4)	56.9
		Out-dist test	18.3	21.0	20.3 ($\downarrow$ 0.7)	25.3

Table 3: Math candidate-layer rankings for **QWEN3-BASE** series. Rankings are obtained using the gradient norms of attention heads in each layer, computed on a 3k-subset of `DAPO-Math-17k`.

Model	Performance ranking (Zhang et al., 2026)	Gradient norm ranking
QWEN3-1.7B-BASE	$10 > 12 > 1 > 7 > 24$	$12 > 10 > 7 > 24 > 1$
QWEN3-4B-BASE	$16 > 14 > 11 > 24 > 2$	$16 > 14 > 11 > 24 > 2$
QWEN3-8B-BASE	$16 > 15 > 8 > 2 > 0$	$15 > 16 > 0 > 8 > 2$



(a) **QWEN3-1.7B**, IF



(b) **QWEN3-4B**, Math

Figure 1: Sparsity–performance trade-off on training, in-distribution test, and out-of-distribution test sets. Each panel includes top 10, 30, 50, 70, and 90% attention-head tuning plus full-attention tuning. All jobs share exactly the same training configuration.

5 PARTITIONING THE ATTENTION HEADS

5.1 THEORETICAL UNDERSTANDINGS

We consider sequential learning of two tasks A and B. Starting from a base model θ_0, task A is trained on a head set S_A to obtain θ_A, after which task B is trained on a head set S_B from $\theta(0) = \theta_A$. Recall that Section 4 motivates using high-gradient-norm heads for task A.

Updating the same parameters for task B can overwrite the parameter changes induced by task A. This concern is particularly pronounced when the trainable parameters are restricted to a sparse subset, motivating the use of disjoint head sets across tasks. Sometimes we may assume an exact partition, $S_B \subseteq S_A^c$.

We now characterize the trade-off between learning task B and preserving task A, *i.e.*, plasticity and stability. Omitted proofs are deferred to Appendix A.2.

Proposition 7. *Let $\theta(t)$ follow task-B Euclidean gradient flow restricted to S_B, initialized at $\theta(0) = \theta_A$. Then $-\frac{d}{dt}\mathcal{L}_B(\theta(t)) = \mathcal{G}_B(S_B;\theta(t))^2$, while the change in the task-A loss satisfies $\left|\frac{d}{dt}\mathcal{L}_A(\theta(t))\right| \leqslant \mathcal{G}_A(S_B;\theta(t))\mathcal{G}_B(S_B;\theta(t))$.*

Proof. The first equation is by Proposition 1. The gradient flow satisfies $\dot{\theta}_{S_B}(t) = -\nabla_{\theta_{S_B}}\mathcal{L}_B(\theta(t))$ and $\dot{\theta}_{S_B^c}(t) = 0$, so the chain rule gives $\frac{d}{dt}\mathcal{L}_A(\theta(t)) = -\langle \nabla_{\theta_{S_B}}\mathcal{L}_A(\theta(t)), \nabla_{\theta_{S_B}}\mathcal{L}_B(\theta(t))\rangle$. The desired bound follows from Cauchy–Schwarz inequality. □

Corollary 8. *Define the contraharmonic time-averaged gradient norm by*

$$\widetilde{\mathcal{G}}_B(S_B;t) := \frac{\int_0^t \mathcal{G}_B(S_B;\theta(s))^2\,ds}{\int_0^t \mathcal{G}_B(S_B;\theta(s))\,ds}, \quad \widetilde{\mathcal{G}}_{A|B}(S_B;t) := \frac{\int_0^t \mathcal{G}_A(S_B;\theta(s))\mathcal{G}_B(S_B;\theta(s))\,ds}{\int_0^t \mathcal{G}_B(S_B;\theta(s))\,ds}.$$

Then $|\mathcal{L}_A(\theta(t)) - \mathcal{L}_A(\theta_A)| \leqslant \frac{\widetilde{\mathcal{G}}_{A|B}(S_B;t)}{\widetilde{\mathcal{G}}_B(S_B;t)}[\mathcal{L}_B(\theta_A) - \mathcal{L}_B(\theta(t))]$.

Remark 4. For task B, we prefer heads with small task-A but large task-B gradient norm. When $S_B \subseteq S_A^c$, since heads in S_A^c are not directly updated during task-A training, their task-A gradient norms may remain controlled by their initial values when the gradient changes smoothly. This suggests balancing the allocation by assigning some high-task-A-gradient heads to task A and some high-task-B-gradient heads to task B, motivating our later allocation strategy design.

Proposition 9. *Suppose the task-B restricted gradient flow exists on $[0,T]$, and $\exists \beta_A, \beta_B \geqslant 0$ s.t.*

$$-\frac{\nabla_{\theta_{S_B}}\mathcal{L}_A(\theta(t))^\top \nabla^2_{\theta_{S_B}}\mathcal{L}_A(\theta(t))\nabla_{\theta_{S_B}}\mathcal{L}_B(\theta(t))}{\|\nabla_{\theta_{S_B}}\mathcal{L}_A(\theta(t))\|_2\|\nabla_{\theta_{S_B}}\mathcal{L}_B(\theta(t))\|_2} \leqslant \beta_A, \quad \frac{\nabla_{\theta_{S_B}}\mathcal{L}_B(\theta(t))^\top \nabla^2_{\theta_{S_B}}\mathcal{L}_B(\theta(t))\nabla_{\theta_{S_B}}\mathcal{L}_B(\theta(t))}{\|\nabla_{\theta_{S_B}}\mathcal{L}_B(\theta(t))\|_2^2} \leqslant \beta_B,$$

whenever the corresponding gradient norms > 0. *Then, for every* $t \in (0, T]$,

$$|\mathcal{L}_A(\theta(t)) - \mathcal{L}_A(\theta_A)| \leqslant \frac{\mathcal{G}_A(S_B;\theta_A)}{\widetilde{\mathcal{G}}_B(S_B;t)}[\mathcal{L}_B(\theta_A) - \mathcal{L}_B(\theta(t))] + \frac{\beta_A\,[\mathcal{L}_B(\theta_A) - \mathcal{L}_B(\theta(t))]^2}{2\widetilde{\mathcal{G}}_B(S_B;t)^2}\,,$$

where $\widetilde{\mathcal{G}}_B(S_B;t) \geqslant \sqrt{\frac{1-e^{-2\beta_B t}}{2\beta_B t}}\,\mathcal{G}_B(S_B;\theta_A)$.

Remark 5. Similar to Proposition 4, this result demonstrates that the initial gradient norms can remain informative for controlling the stability–plasticity trade-off over a nontrivial training interval. In our experiments in Section 5.2, rankings computed at the initial checkpoint effectively guide continual training over hundreds of training steps. See Section 6 for discussions on the corresponding second-order assumptions for attention blocks.

5.2 Sequential continual learning experiments

Experimental setup. We evaluate whether partitioning attention heads can reduce interference in continual RL. We sequentially fine-tune **QWEN3-1.7B** and **QWEN3-4B** on two- and three-task sequences over Math, IF, and Coding. We compare head-partitioned tuning with continual full-attention and full-tuning baselines. Implementation details are provided in Appendix C.

Motivated by Section 5.1, we consider three head-allocation strategies:

- **Top–Bottom**: the previous task uses top-ranked heads, while the next task uses the remaining.
- **Round-Robin**: tasks alternately select their highest-ranked unassigned heads.
- **Shared**: in addition to disjoint task-specific heads, a remaining subset is shared across tasks.

For two-task experiments, we compare the first two strategies. For three-task experiments, we use the third, assigning 25% of the heads to each task and sharing the remaining 25%. It is worth emphasizing that we do not claim these strategies to be optimal and exhaustive. Instead, they serve to validate our theory and demonstrate possible ways to apply it.

Main results. Figures 2 to 4 show that head partitioning generally reduces forgetting while preserving adaptation to the new task, especially when full-attention tuning causes large drops on earlier tasks. Top–Bottom typically retains the first task better but adapts slightly worse to the second, while Round-Robin shows the opposite trade-off, consistent with their allocation strategies. Figures 5 and 6 further show that the head-partition approach extends to three sequential tasks with minimal degradation on previous tasks, supporting the theoretical claims in Section 5.1. See supplemental results in Appendix B.2.

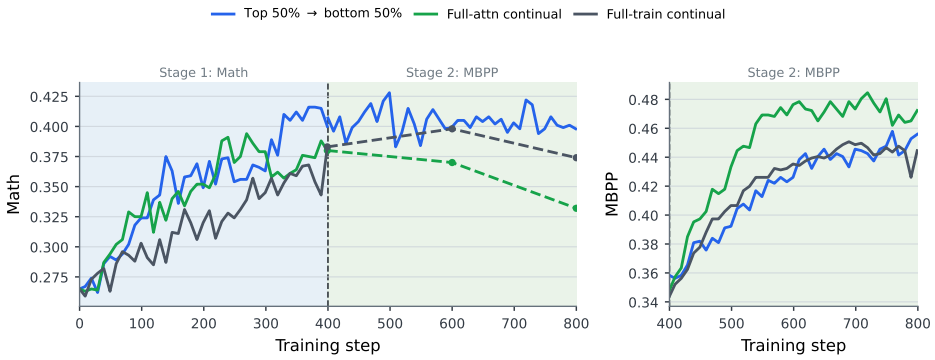


Figure 2: Continual RL accuracy (%) of **QWEN3-1.7B** (Math → Coding). The model is first trained on Math task and then on Coding task. Shaded regions mark stages.

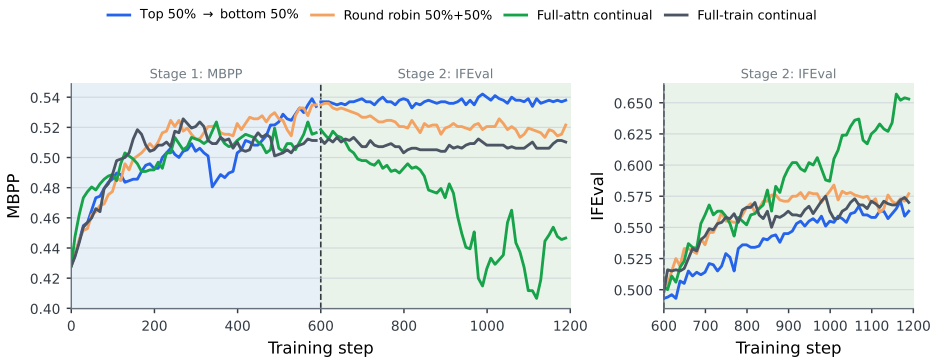


Figure 3: Continual RL accuracy (%) of **QWEN3-4B** (Coding → IF). The model is first trained on Coding task and then on IF task. Shaded regions mark stages.

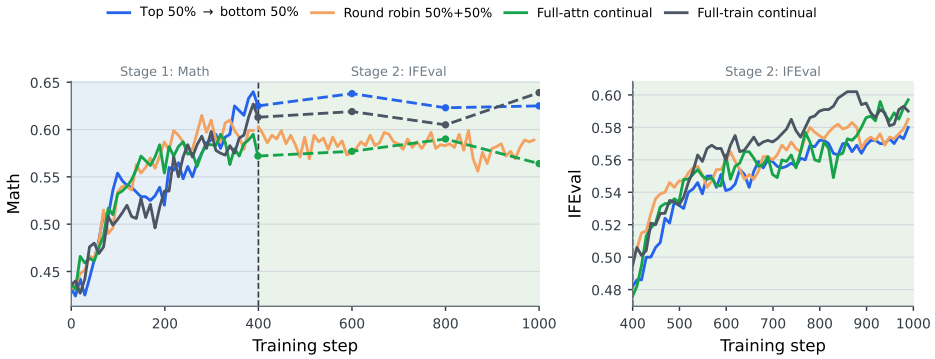


Figure 4: Continual RL accuracy (%) of **QWEN3-4B** (Math → IF). The model is first trained on Math task and then on IF task. Shaded regions mark stages.

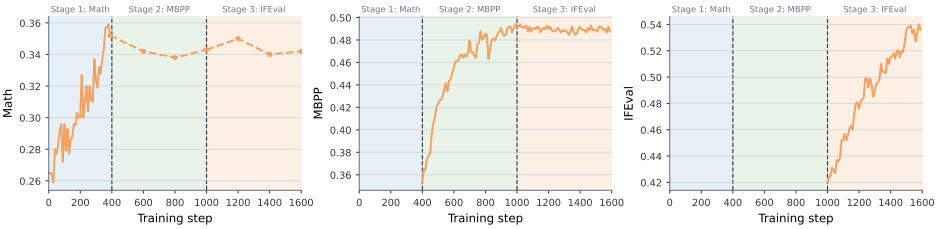


Figure 5: Continual RL accuracy (%) of **QWEN3-1.7B** (Math → Coding → IF). The model is first trained on Math task, then on Coding task, and finally on IF task. Shaded regions mark stages.



Figure 6: Continual RL accuracy (%) of **QWEN3-4B** (Math → IF → Coding). The model is first trained on Math task, then on IF task, and finally on Coding task. Shaded regions mark stages.

Furthermore, to test whether the learned adaptations are functionally localized, we reset the stage-I heads to their original values after stage II. As shown in Table 12 in Appendix B.2, this operation largely degrades stage-I performance while maintaining stage-II performance, suggesting approximate functional isolation across the allocated head subsets. This agrees with the observation of Jin et al. (2025) that attention-only fine-tuning is well approximated by model linearization.

6 Second-Order Structure of Sparse Attention-Head Updates

In Sections 4.1 and 5.1, we show that favorable second-order properties allow us to control how gradient norms evolve during fine-tuning. For $S \subseteq U$, $\|\nabla^2_{\theta_S}\mathcal{L}\|_{\mathrm{op}} \leqslant \|\nabla^2_{\theta_U}\mathcal{L}\|_{\mathrm{op}}$, so sparsity generically controls second-order effects but does not explain why attention heads are preferable to arbitrary sparse parameter subsets. We next develop a more detailed, attention-head-specific second-order analysis. Omitted proofs are deferred to Appendix A.3.

We consider a single multi-head self-attention layer on a fixed rollout batch and study the second-order bound w.r.t. its attention parameters, treating all subsequent computation as part of the loss. Let $F(\theta)$ denote the layer output and let $\mathcal{L}(\theta) = \ell(F(\theta))$. Following Ormaniec et al. (2025), we use the Gauss-Newton decomposition of the Hessian:

$$\mathbf{H} := \nabla^2_\theta \mathcal{L} = \mathbf{H}_f + \mathbf{H}_o \ , \ \mathbf{H}_f := \frac{\partial \ell}{\partial F}\nabla^2_\theta F \ , \ \mathbf{H}_o := \left(\frac{\partial F}{\partial \theta}\right)^\top \nabla^2_F \ell(F) \left(\frac{\partial F}{\partial \theta}\right) \ .$$

Proposition 10. *For multi-head self-attention and two heads $h \neq h'$, $(\mathbf{H}_f)_{hh'} = 0$. Consequently, for any head set S, $\|(\mathbf{H}_f)_{SS}\|_{\mathrm{op}} = \max_{h\in S} \|(\mathbf{H}_f)_{hh}\|_{\mathrm{op}}$.*

Remark 6. Attention heads are naturally isolated as each head depends only on its own parameter block, making H_f block-diagonal across heads. This does not hold for shared parameterizations such as standard LoRA, where the matrix A is shared across attention heads, nor for more unstructured sparse parameter subsets.

We can further show that when attention concentrates on similar token representations, a sharper single-head bound is possible. Such a regime is consistent with the selective and specialized attention patterns observed empirically (Yin & Steinhardt, 2025; Olsson et al., 2022).

Proposition 11 (Special case: concentrated attention)**.** *Consider one attention head h. Suppose that for every attention row $A_{i,:}$ there exists a token set T_i s.t. $\sum_{j\in T_i} A_{ij} \geqslant 1-\varepsilon$ and $\max_{j,j'\in T_i} \|X_{j,:} - X_{j',:}\|_2 \leqslant \rho$. Assume the input and weight norms are bounded and $\|\nabla_F \ell(F)\|_2 \leqslant G$. Then we have*

$$\|(\mathbf{H}_f)_{hh}\|_{\mathrm{op}} = O\left(G\left(\rho^2 + \rho^3 + \varepsilon\right)\right) \ .$$

Proposition 12. *Define the tangent kernel $K_S := \frac{\partial F}{\partial \theta_S}\left(\frac{\partial F}{\partial \theta_S}\right)^\top$ for a head set S. If the outer objective is locally β-smooth in the attention output, i.e., $\|\nabla^2_F \ell(F)\|_{\mathrm{op}} \leqslant \beta$, then*

$$\|(\mathbf{H}_o)_{SS}\|_{\mathrm{op}} \leqslant \beta \|K_S\|_{\mathrm{op}} \ .$$

Remark 7. Jin et al. (2025) suggest that attention-only fine-tuning exhibits kernel-like behavior. In such a locally linearized regime, the Jacobian is unlikely to change rapidly, so K_S remains close to its initial value and its operator norm stays controlled.

Corollary 13. *Under the assumptions of Proposition 12, for any head set S and any vector $v_S \neq 0$, whenever $\nabla_{\theta_S}\mathcal{L} \neq 0$, we have*

$$\frac{\left|(\nabla_{\theta_S}\mathcal{L})^\top \mathbf{H}_{SS} v_S\right|}{\|\nabla_{\theta_S}\mathcal{L}\|_2 \|v_S\|_2} \leqslant \max_{h\in S} \|(\mathbf{H}_f)_{hh}\|_{\mathrm{op}} + \beta \|K_S\|_{\mathrm{op}} \ .$$

This result recovers the second-order terms used in Propositions 4 and 9, explaining why attention-head tuning can enjoy a favorable second-order bound.

7 Conclusion

We study sparse and continual RL fine-tuning through gradient-based ranking and partitioning of attention heads. We show theoretically that gradient norm characterizes the training benefit of a head subset. Empirically, high-ranked heads support efficient sparse RL tuning, while partitioning heads across tasks effectively reduces forgetting in continual RL.

It is also important to acknowledge our limitations. (i) Our experiments are restricted to relatively small language models, limited tasks, and settings without long reasoning traces. (ii) Our analysis focuses on attention parameters and may not directly extend to MLP layers, which would be important for sparse full-model fine-tuning. (iii) The theoretical second-order analysis is restricted to a single multi-head self-attention layer.

AI USE STATEMENT

In this work, we used generative AI tools for paper drafting and polishing, coding, literature review, and quickly testing mathematical intuitions. We have not used generative AI tools for generating synthetic datasets. LLM-generated code was tested for correctness. All final theoretical results are of our own. All AI outputs were reviewed and edited by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Anna Bair, Yixuan Even Xu, Mingjie Sun, and J Zico Kolter. Compressed sensing for capability localization in large language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=X4MFtxw1sB.

Arslan Chaudhry, Naeemullah Khan, Puneet K. Dokania, and Philip H. S. Torr. Continual learning in low-rank orthogonal subspaces. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/70d85f35a1fdc0ab701ff78779306407-Abstract.html.

Howard Chen, Noam Razin, Karthik R Narasimhan, and Danqi Chen. Retaining by doing: The role of on-policy data in mitigating forgetting. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=ODTM64azGa.

Chen Dan, Hong Wang, Hongyang Zhang, Yuchen Zhou, and Pradeep Kumar Ravikumar. Optimal analysis of subset-selection based l_p low rank approximation. In *Neural Information Processing Systems*, 2019. URL https://api.semanticscholar.org/CorpusID:260537614.

Thang Van Doan, Mehdi Abbana Bennani, Bogdan Mazoure, Guillaume Rabusseau, and Pierre Alquier. A theoretical analysis of catastrophic forgetting through the ntk overlap matrix. In *International Conference on Artificial Intelligence and Statistics*, 2020. URL https://api.semanticscholar.org/CorpusID:222208953.

John C. Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, 12:2121–2159, 2011. URL https://api.semanticscholar.org/CorpusID:538820.

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *AISTATS*, pp. 3762–3773, 2020. URL http://proceedings.mlr.press/v108/farajtabar20a.html.

Fireworks.ai. Frontier rl is cheaper than you think. Fireworks.ai Blog, 3 2026. URL https://fireworks.ai/blog/frontier-rl-is-cheaper-than-you-think. Accessed: 2026-09-20.

Demi Guo, Alexander Rush, and Yoon Kim. Parameter-efficient transfer learning with diff pruning. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4884–4896, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.378. URL https://aclanthology.org/2021.acl-long.378/.

Leonardo Iurada, Marco Ciccone, and Tatiana Tommasi. Efficient model editing with task-localized sparse fine-tuning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=TDyE2iuvyc.

Ruochen Jin, Bojian Hou, Jiancong Xiao, Weijie J Su, and Li Shen. Fine-tuning attention modules only: Enhancing weight disentanglement in task arithmetic. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=dj0TktJcVI.

Giulia Lanzillotta, Sidak Pal Singh, Benjamin F. Grewe, and Thomas Hofmann. Towards guarantees for parameter isolation in continual learning. *ArXiv*, abs/2310.01165, 2023. URL https://api.semanticscholar.org/CorpusID:263605982.

Yan-Shuo Liang and Wu-Jun Li. Inflora: Interference-free low-rank adaptation for continual learning. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23638–23647, 2024. URL https://api.semanticscholar.org/CorpusID:268819832.

Jessy Lin, Luke S. Zettlemoyer, Gargi Ghosh, Wen tau Yih, Aram H. Markosyan, Vincent-Pierre Berges, and Barlas Ouguz. Continual learning via sparse memory finetuning. *ArXiv*, abs/2510.15103, 2025. URL https://api.semanticscholar.org/CorpusID:282203348.

Zekai Lin, Chao Xue, Di Liang, Xingsheng Han, Peiyang Liu, Xianjie Wu, Lei Jiang, Yu Lu, Bob Simons, Shuang Liang, and Minlong Peng. Parameter importance is not static: Evolving parameter isolation for supervised fine-tuning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 32846–32864, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.1516. URL https://aclanthology.org/2026.acl-long.1516/.

Lirui Luo, Guoxi Zhang, Hongming Xu, Cong Fang, and Qing Li. SPHERE: Mitigating the loss of spectral plasticity in mixture-of-experts for deep reinforcement learning. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=hXyv6xeHkO.

Sadhika Malladi, Alexander Wettig, Dingli Yu, Danqi Chen, and Sanjeev Arora. A kernel-based view of language model fine-tuning. In *ICML*, pp. 23610–23641, 2023. URL https://proceedings.mlr.press/v202/malladi23a.html.

Erfan Miahi and Eugene Belilovsky. Understanding and exploiting weight update sparsity for communication-efficient distributed rl. *ArXiv*, abs/2602.03839, 2026. URL https://api.semanticscholar.org/CorpusID:285275603.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In *NeurIPS*, pp. 14014–14024, 2019. URL http://papers.nips.cc/paper/9551-are-sixteen-heads-really-better-than-one.

Miles Team. Miles: Enterprise-grade reinforcement learning for large-scale model post-training. https://github.com/radixark/miles, 2026.

Nikhil Shivakumar Nayak, Krishnateja Killamsetty, Ligong Han, Abhishek Bhandwaldar, Prateek Chanda, Kai Xu, Oleg Silkin, Mustafa Eyceoz, Hao Wang, Aldo Pareja, and Akash Srivastava. Sculpting subspaces: Constrained full fine-tuning in LLMs for continual learning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=vQcyqsGJDw.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova Dassarma, Thomas Henighan, Benjamin Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, John Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom B. Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *ArXiv*, abs/2209.11895, 2022. URL https://api.semanticscholar.org/CorpusID:252532078.

Weronika Ormaniec, Felix Dangel, and Sidak Pal Singh. What does it mean to be a transformer? insights from a theoretical hessian analysis. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=3ddi7Uss2A.

Fuli Qiao and Mehrdad Mahdavi. Merge before forget: A single loRA continual learning via continual merging. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=i1Rj7yU6eF.

Haomiao Qiu, Miao Zhang, Ziyue Qiao, Weili Guan, Min Zhang, and Liqiang Nie. SplitloRA: Balancing stability and plasticity in continual learning through gradient space splitting. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Zm1hjXxRQV.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

M. Sagitova, O. Duranthon, and L. Zdeborová. Specialization of softmax attention heads: insights from the high-dimensional single-location model, 2026. URL https://arxiv.org/abs/2603.03993.

Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=3AOj0RCNC2.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. RL's razor: Why online reinforcement learning forgets less. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=7HNRYT4V44.

Ruizhe Shi, Minhak Song, Runlong Zhou, Zihan Zhang, Maryam Fazel, and Simon Shaolei Du. Understanding the performance gap in preference learning: A dichotomy of RLHF and DPO. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=sg94PRd3kD.

Anke Tang, Enneng Yang, Li Shen, Yong Luo, Han Hu, Lefei Zhang, Bo Du, and Dacheng Tao. Merging on the fly without retraining: A sequential approach to scalable continual model merging. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=rdGMyTPhui.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5797–5808, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1580. URL https://aclanthology.org/P19-1580/.

Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Orthogonal subspace learning for language model continual learning. In *EMNLP (Findings)*, pp. 10658–10671, 2023. URL https://doi.org/10.18653/v1/2023.findings-emnlp.715.

Enneng Yang, An Quang Tang, Li Shen, Gui-Bing Guo, Xingwei Wang, Xiaochun Cao, and Jie Zhang. Continual model merging without data: Dual projections for balancing stability and plasticity. *Advances in Neural Information Processing Systems 38*, 2025. URL https://api.semanticscholar.org/CorpusID:290960643.

Yunzhen Yao, Lie He, and Michael Gastpar. Leveraging sparsity for sample-efficient preference learning: A theoretical perspective. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=nxCWKWok8N.

Kayo Yin and Jacob Steinhardt. Which attention heads matter for in-context learning? In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=C7XmEByCFv.

Juzheng Zhang, Jiacheng You, Ashwinee Panda, and Tom Goldstein. LoRI: Reducing cross-task interference in multi-task low-rank adaptation. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=b8cW86QcOD.

Zijian Zhang, Rizhen Hu, Athanasios Glentis, Dawei Li, Chung-Yiu Yau, Hongzhou Lin, and Mingyi Hong. Is one layer enough? training a single transformer layer can match full-parameter rl training. *ArXiv*, abs/2607.01232, 2026. URL https://api.semanticscholar.org/CorpusID:289706488.

Guodong Zheng, Peng Wang, Shengchao Hu, Quan Zheng, and Li Shen. Understanding the dynamics of forgetting and generalization in continual learning via the neural tangent kernel. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=NE2yIxdo1w.

A Omitted Proofs

A.1 Omitted proofs in Section 4.1

Proof of Corollary 2. By Proposition 1,

$$-\frac{d}{dt}\mathcal{L}(\theta(t;S))\Big|_{t=0} = \mathcal{G}(S)^2 = \sum_{h\in S}\mathcal{G}(\{h\})^2 \ .$$

Hence selecting k heads of largest gradient norm maximizes the initial loss decrease. □

Proof of Corollary 3. Integrating Proposition 1 gives

$$\mathcal{L}(\theta_0) - \mathcal{L}(\theta(t;S)) = \int_0^t \mathcal{G}(S;\theta(s;S))^2 ds = t\,\overline{\mathcal{G}}(S;t)^2 \ .$$

Then for any S with $\mathcal{G}(S) > 0$ we have

$$\mathcal{L}(\theta_0) - \mathcal{L}(\theta(t;S_k)) = t\,\overline{\mathcal{G}}(S_k;t)^2 \geqslant \frac{\overline{\mathcal{G}}(S_k;t)^2}{\overline{\mathcal{G}}(S;t)^2}\left[\mathcal{L}(\theta_0) - \mathcal{L}(\theta(t;S))\right] \ .$$

Taking the maximum over $|S| \leqslant k$ proves the claim. □

Proof of Proposition 4. Along the restricted gradient flow,

$$\frac{d}{dt}\mathcal{G}(S;\theta(t;S))^2 = -2(\nabla_{\theta_S}\mathcal{L}(\theta(t;S)))^\top \nabla^2_{\theta_S}\mathcal{L}(\theta(t;S))\nabla_{\theta_S}\mathcal{L}(\theta(t;S)) \ .$$

Hence, whenever $\mathcal{G}(S;\theta(t;S)) > 0$,

$$-2\beta\mathcal{G}(S;\theta(t;S))^2 \leqslant \frac{d}{dt}\mathcal{G}(S;\theta(t;S))^2 \leqslant 2\beta\mathcal{G}(S;\theta(t;S))^2 \ .$$

Integrating gives

$$e^{-2\beta s}\mathcal{G}(S)^2 \leqslant \mathcal{G}(S;\theta(s;S))^2 \leqslant e^{2\beta s}\mathcal{G}(S)^2$$

for every $s \in [0,t]$. Averaging over $s \in [0,t]$ yields

$$\frac{1-e^{-2\beta t}}{2\beta t}\mathcal{G}(S)^2 \leqslant \overline{\mathcal{G}}(S;t)^2 \leqslant \frac{e^{2\beta t}-1}{2\beta t}\mathcal{G}(S)^2 \ .$$ □

Proof of Proposition 5. The objective $\mathcal{L}_{\mathrm{adam}_\lambda}$ is strictly convex and separable across heads. For head h, by the KKT condition, $\Delta_h = 0$ is optimal iff

$$0 \in m_h + \lambda\partial\|\Delta_h\|_2\big|_{\Delta_h=0} = \{m_h + \lambda u : \|u\|_2 \leqslant 1\} \ .$$

This condition is equivalent to $\|m_h\|_2 \leqslant \lambda$. Strict convexity gives uniqueness. □

Proof of Proposition 6. By restricting Δ to S and solving $\mathcal{L}_{\mathrm{adam}}(\Delta)$, we have $\Delta^\star_S = -\eta D_S^{-1} m$ and $\mathcal{L}_{\mathrm{adam}}(\Delta^\star_S) = -\frac{\eta m_S^\top D_{SS}^{-1} m_S}{2}$. Then we have that, for any $|S| \leqslant k$,

$$-\mathcal{L}_{\mathrm{adam}}(\Delta^\star_{S_k^{(m)}}) \geqslant \frac{\eta\|m_{S_k^{(m)}}\|_2^2}{2\max_{i\in\theta_{S_k^{(m)}}}(\sqrt{v_i}+\epsilon_{\mathrm{adam}})} \geqslant \frac{\min_{i\in\theta_S}(\sqrt{v_i}+\epsilon_{\mathrm{adam}})}{\max_{i\in\theta_{S_k^{(m)}}}(\sqrt{v_i}+\epsilon_{\mathrm{adam}})}\left[-\mathcal{L}_{\mathrm{adam}}(\Delta^\star_S)\right] \ ,$$

where the second inequality uses $\|m_{S_k^{(m)}}\|_2^2 \geqslant \|m_S\|_2^2$. □

A.2 Omitted proofs in Section 5.1

Proof of Corollary 8. By Proposition 7,

$$|\mathcal{L}_A(\theta(t)) - \mathcal{L}_A(\theta_A)| \leqslant \int_0^t \mathcal{G}_A(S_B;\theta(s))\mathcal{G}_B(S_B;\theta(s))\,ds = \widetilde{\mathcal{G}}_{A|B}(S_B;t)\int_0^t \mathcal{G}_B(S_B;\theta(s))\,ds\,.$$

On the other hand,

$$\mathcal{L}_B(\theta_A) - \mathcal{L}_B(\theta(t)) = \int_0^t \mathcal{G}_B(S_B;\theta(s))^2\,ds = \widetilde{\mathcal{G}}_B(S_B;t)\int_0^t \mathcal{G}_B(S_B;\theta(s))\,ds\,. \qquad \square$$

Proof of Proposition 9. By Cauchy–Schwarz inequality,

$$\widetilde{\mathcal{G}}_B(S_B;t) = \frac{\int_0^t \mathcal{G}_B(S_B;\theta(s))^2\,ds}{\int_0^t \mathcal{G}_B(S_B;\theta(s))\,ds} \geqslant \sqrt{\frac{1}{t}\int_0^t \mathcal{G}_B(S_B;\theta(s))^2\,ds}\,.$$

Then Proposition 4 gives

$$\mathcal{G}_B(S_B;\theta(s)) \geqslant \sqrt{\frac{1-e^{-2\beta_B t}}{2\beta_B t}}\,\mathcal{G}_B(S_B;\theta_A)\,.$$

Note that

$$\frac{d}{dt}\mathcal{G}_A(S_B;\theta(t)) = -\frac{\nabla_{\theta_{S_B}}\mathcal{L}_A(\theta(t))^\top \nabla^2_{\theta_{S_B}}\mathcal{L}_A(\theta(t))\nabla_{\theta_{S_B}}\mathcal{L}_B(\theta(t))}{\|\nabla_{\theta_{S_B}}\mathcal{L}_A(\theta(t))\|_2} \leqslant \beta_A \mathcal{G}_B(S_B;\theta(t))\,.$$

Hence, for every $s \in [0,t]$,

$$\mathcal{G}_A(S_B;\theta(s)) \leqslant \mathcal{G}_A(S_B;\theta_A) + \beta_A\int_0^s \mathcal{G}_B(S_B;\theta(u))\,du\,.$$

Then we have

$$\begin{aligned}
|\mathcal{L}_A(\theta(t)) - \mathcal{L}_A(\theta_A)| &\leqslant \int_0^t \mathcal{G}_A(S_B;\theta(s))\mathcal{G}_B(S_B;\theta(s))\,ds \\
&\leqslant \int_0^t \left(\mathcal{G}_A(S_B;\theta_A) + \beta_A\int_0^s \mathcal{G}_B(S_B;\theta(u))\,du\right)\mathcal{G}_B(S_B;\theta(s))\,ds \\
&= \mathcal{G}_A(S_B;\theta_A)\int_0^t \mathcal{G}_B(S_B;\theta(s))\,ds + \frac{\beta_A}{2}\left(\int_0^t \mathcal{G}_B(S_B;\theta(s))\,ds\right)^2\,.
\end{aligned}$$

Using the fact that $\int_0^t \mathcal{G}_B(S_B;\theta(s))\,ds = \frac{\mathcal{L}_B(\theta_A)-\mathcal{L}_B(\theta(t))}{\widetilde{\mathcal{G}}_B(S_B;t)}$ gives the desired bound. $\square$

A.3 Omitted proofs in Section 6

For an attention row $A_{i,:}$, define its attention-weighted mean and centered moments by

$$\mu_i := \sum_j A_{ij}X_{j,:}, \quad M_{2,i} := \sum_j A_{ij}(X_{j,:}-\mu_i)(X_{j,:}-\mu_i)^\top, \quad M_{3,i} := \sum_j A_{ij}(X_{j,:}-\mu_i)^{\otimes 3}\,,$$

where $x^{\otimes 3}$ denotes the third-order tensor product.

Lemma 1. *Suppose $\|X_{j,:}\|_2 \leqslant B$ for all j. If there exists T_i s.t.*

$$\sum_{j\in T_i} A_{ij} \geqslant 1-\varepsilon, \qquad \max_{j,j'\in T_i}\|X_{j,:}-X_{j',:}\|_2 \leqslant \rho\,,$$

then

$$\|M_{2,i}\|_{\mathrm{op}} = O_B(\rho^2+\varepsilon), \qquad \|M_{3,i}\| = O_B(\rho^3+\varepsilon)\,.$$

Proof. Fix any $j_0 \in T_i$. Since $\sum_j A_{ij} = 1$,

$$\begin{aligned}\|\mu_i - X_{j_0,:}\|_2 &\leqslant \sum_j A_{ij}\|X_{j,:} - X_{j_0,:}\|_2 \\ &\leqslant \rho + 2B\varepsilon \ .\end{aligned}$$

Hence, for $j \in T_i$,

$$\|X_{j,:} - \mu_i\|_2 \leqslant 2\rho + 2B\varepsilon \ ,$$

while for $j \notin T_i$,

$$\|X_{j,:} - \mu_i\|_2 \leqslant 2B \ .$$

Therefore,

$$\begin{aligned}\|M_{2,i}\|_{\mathrm{op}} &\leqslant \sum_j A_{ij}\|X_{j,:} - \mu_i\|_2^2 \\ &\leqslant (2\rho + 2B\varepsilon)^2 + 4B^2\varepsilon = O_B(\rho^2 + \varepsilon) \ ,\end{aligned}$$

and similarly,

$$\begin{aligned}\|M_{3,i}\| &\leqslant \sum_j A_{ij}\|X_{j,:} - \mu_i\|_2^3 \\ &\leqslant (2\rho + 2B\varepsilon)^3 + 8B^3\varepsilon = O_B(\rho^3 + \varepsilon) \ .\end{aligned}$$

□

Lemma 2. *Consider a single self-attention head*

$$F_h = A_h X W_h^V, \qquad A_h = \mathrm{softmax}\left(\frac{XW_h^Q(W_h^K)^\top X^\top}{\sqrt{d_k}}\right) \ ,$$

with parameter block $\theta_h = (W_h^Q, W_h^K, W_h^V)$. The second-derivative formulas of Ormaniec et al. (2025) express every nonzero block of the corresponding functional Hessian through the second and third centered attention moments. In particular, under bounded input and weight norms, there exists a constant C depending only on these bounds and the dimensions s.t.

$$\|(\mathbf{H}_f)_{hh}\|_{\mathrm{op}} \leqslant C\|\nabla_F \ell(F)\|_2 \left(\max_i \|M_{2,i}\|_{\mathrm{op}} + \max_i \|M_{3,i}\|\right) \ .$$

Proof. For each head h, we further decompose its parameter block as

$$\theta_h = \left(W_h^Q, W_h^K, W_h^V\right) \ .$$

Accordingly, the within-head functional-Hessian block can be written as

$$(\mathbf{H}_f)_{hh} = \begin{pmatrix} (\mathbf{H}_f)_{W_h^Q W_h^Q} & (\mathbf{H}_f)_{W_h^Q W_h^K} & (\mathbf{H}_f)_{W_h^Q W_h^V} \\ (\mathbf{H}_f)_{W_h^K W_h^Q} & (\mathbf{H}_f)_{W_h^K W_h^K} & (\mathbf{H}_f)_{W_h^K W_h^V} \\ (\mathbf{H}_f)_{W_h^V W_h^Q} & (\mathbf{H}_f)_{W_h^V W_h^K} & (\mathbf{H}_f)_{W_h^V W_h^V} \end{pmatrix} .$$

By Theorem 3.2 and Remark 3.1 of Ormaniec et al. (2025), each nonzero (W_h^Q, W_h^K, W_h^V) block above can be written as a linear function of M_2, M_3, or both. Under bounded input and weight norms, this implies

$$\|(\mathbf{H}_f)_{hh}\|_{\mathrm{op}} \leqslant C\|\nabla_F \ell(F)\|_2 \left(\max_i \|M_{2,i}\|_{\mathrm{op}} + \max_i \|M_{3,i}\|\right) \ .$$ □

Proof of Proposition 10. The output of a multi-head self-attention layer is the sum of head outputs, each depending only on its own parameter block. Hence, for $h \neq h'$,

$$\nabla^2_{\theta_h, \theta_{h'}} F = 0 \ .$$

The definition of the functional Hessian therefore gives $(\mathbf{H}_f)_{hh'} = 0$. Thus $(\mathbf{H}_f)_{SS}$ is block diagonal, whose operator norm is the maximum operator norm of its diagonal blocks. □

Proof of Proposition 11. By Lemma 1,

$$\max_i \|M_{2,i}\|_{\mathrm{op}} = O(\rho^2 + \varepsilon), \qquad \max_i \|M_{3,i}\| = O(\rho^3 + \varepsilon) \ .$$

Applying Lemma 2 and using $\|\nabla_F \ell(F)\|_2 \leqslant G$ yields

$$\|(\mathbf{H}_f)_{hh}\|_{\mathrm{op}} = O\left(G\left(\rho^2 + \rho^3 + \varepsilon\right)\right) \ . \qquad \square$$

Proof of Proposition 12. We have

$$\|(\mathbf{H}_o)_{SS}\|_{\mathrm{op}} = \left\| \left(\frac{\partial F}{\partial \theta_S}\right)^\top \nabla_F^2 \ell(F) \left(\frac{\partial F}{\partial \theta_S}\right) \right\|_{\mathrm{op}} \leqslant \left\|\frac{\partial F}{\partial \theta_S}\right\|_{\mathrm{op}}^2 \|\nabla_F^2 \ell(F)\|_{\mathrm{op}} \leqslant \beta \|K_S\|_{\mathrm{op}} \ . \qquad \square$$

Proof of Corollary 13. Note that

$$\left|(\nabla_{\theta_S} \mathcal{L})^\top \mathbf{H}_{SS} v_S\right| \leqslant \|\mathbf{H}_{SS}\|_{\mathrm{op}} \|\nabla_{\theta_S} \mathcal{L}\|_2 \|v\|_2 \leqslant \left(\|(\mathbf{H}_f)_{SS}\|_{\mathrm{op}} + \|(\mathbf{H}_o)_{SS}\|_{\mathrm{op}}\right) \|\nabla_{\theta_S} \mathcal{L}\|_2 \|v\|_2 \ .$$

Combining this with Propositions 10 and 12 gives the desired bound. $\square$

B SUPPLEMENTAL EXPERIMENTAL RESULTS

B.1 SUPPLEMENTAL TRAINING CURVES

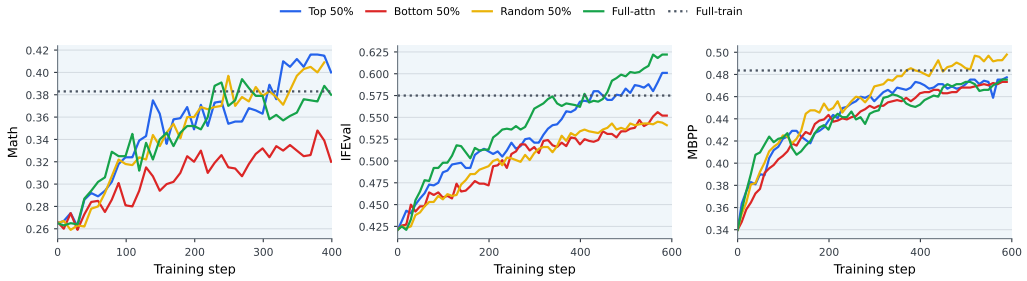


Figure 7: Sparse RL training accuracy (%) curves of **QWEN3-1.7B**. Methods include top 50% and bottom 50% attention-head tuning, full-attention tuning, and full tuning. Full tuning is shown as a final-score horizontal line.



Figure 8: Sparse RL training accuracy (%) curves of **QWEN3-4B**. Methods include top 50% and bottom 50% attention-head tuning, full-attention tuning, and full tuning. Full tuning is shown as a final-score horizontal line.

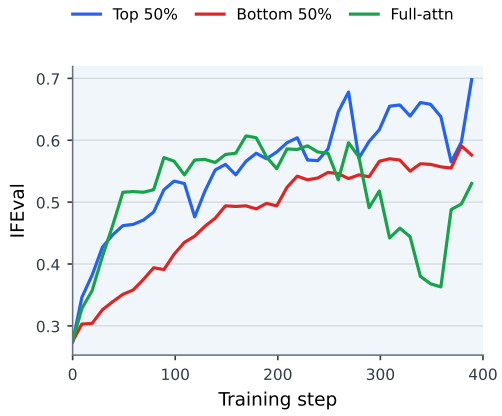


Figure 9: Sparse RL training accuracy (%) curves of **MISTRAL-7B**. Methods include top 50% and bottom 50% attention-head tuning, and full-attention tuning.

B.2 SUPPLEMENTAL NUMERICAL RESULTS

Table 4: Continual RL accuracy (%) of **QWEN3-1.7B** (Math → Coding). The model is first trained on Math task and then on Coding task. Parenthesized arrows are performance changes from the previous stage. The highest performance across all methods is highlighted in pink. The performance before training on the corresponding task is highlighted in gray.

Method	Train stage	Eval split	Math metric	Coding metric
Top 50% → bottom 50%	I	In-dist test	37.5	56.6
		Out-dist test	75.6	37.0
	II	In-dist test	35.8 (↓ 1.7)	56.1
		Out-dist test	75.2 (↓ 0.4)	37.7
Full-attn continual	I	In-dist test	33.1	54.8
		Out-dist test	74.2	35.2
	II	In-dist test	30.0 (↓ 3.1)	60.1
		Out-dist test	73.0 (↓ 1.2)	36.9
Full continual	I	In-dist test	35.3	55.0
		Out-dist test	73.6	35.0
	II	In-dist test	34.5 (↓ 0.8)	58.5
		Out-dist test	73.8 (↑ 0.2)	33.6

Table 5: Continual RL accuracy (%) of **QWEN3-1.7B** (IF → Math). The model is first trained on IF task and then on Math task. Parenthesized arrows are performance changes from the previous stage. The highest performance across all methods is highlighted in pink. The performance before training on the corresponding task is highlighted in gray.

Method	Train stage	Eval split	IF metric	Math metric
Top 50% → bottom 50%	I	In-dist test	58.0	24.6
		Out-dist test	20.3	70.6
	II	In-dist test	57.7 (↓ 0.3)	30.5
		Out-dist test	20.7 (↑ 0.3)	73.8
Round robin 50%+50%	I	In-dist test	52.5	27.2
		Out-dist test	19.0	70.0
	II	In-dist test	52.2 (↓ 0.3)	33.9
		Out-dist test	18.7 (↓ 0.3)	74.0
Full-attn continual	I	In-dist test	59.4	24.0
		Out-dist test	20.0	65.0
	II	In-dist test	59.3 (↓ 0.1)	35.7
		Out-dist test	19.3 (↓ 0.7)	78.2
Full continual	I	In-dist test	54.6	25.7
		Out-dist test	19.7	65.2
	II	In-dist test	54.4 (↓ 0.2)	36.1
		Out-dist test	21.7 (↑ 2.0)	76.8

Table 6: Continual RL accuracy (%) of **QWEN3-4B** (Coding → IF). The model is first trained on Coding task and then on IF task. Parenthesized arrows are performance changes from the previous stage. The highest performance across all methods is highlighted in pink. The performance before training on the corresponding task is highlighted in gray.

Method	Train stage	Eval split	Coding metric	IF metric
Top 50% → bottom 50%	I	In-dist test	69.0	49.9
		Out-dist test	51.4	24.3
	II	In-dist test	68.3 (↓ 0.7)	53.1
		Out-dist test	52.6 (↑ 1.2)	23.0
Round robin 50%+50%	I	In-dist test	68.0	49.8
		Out-dist test	42.9	25.7
	II	In-dist test	64.3 (↓ 3.7)	55.5
		Out-dist test	34.6 (↓ 8.3)	25.0
Full-attn continual	I	In-dist test	65.1	51.8
		Out-dist test	47.9	22.0
	II	In-dist test	54.8 (↓ 10.3)	62.2
		Out-dist test	34.1 (↓ 13.8)	22.3
Full continual	I	In-dist test	66.4	52.1
		Out-dist test	43.6	22.7
	II	In-dist test	65.9 (↓ 0.5)	56.0
		Out-dist test	41.7 (↓ 1.9)	25.0

Table 7: Continual RL accuracy (%) of **QWEN3-4B** (Math → IF). The model is first trained on Math task and then on IF task. Parenthesized arrows are performance changes from the previous stage. The highest performance across all methods is highlighted in pink. The performance before training on the corresponding task is highlighted in gray.

Method	Train stage	Eval split	Math metric	IF metric
Top 50% → bottom 50%	I	In-dist test	59.3	49.6
		Out-dist test	88.4	22.7
	II	In-dist test	60.0 (↑ 0.7)	53.6
		Out-dist test	87.2 (↓ 1.2)	26.0
Round robin 50%+50%	I	In-dist test	53.2	50.3
		Out-dist test	86.0	23.3
	II	In-dist test	53.2 (↓ 0.0)	56.4
		Out-dist test	85.6 (↓ 0.4)	25.3
Full-attn continual	I	In-dist test	55.3	48.9
		Out-dist test	86.0	21.3
	II	In-dist test	51.2 (↓ 4.1)	58.8
		Out-dist test	84.2 (↓ 1.8)	26.0
Full continual	I	In-dist test	58.2	49.8
		Out-dist test	88.2	24.0
	II	In-dist test	57.2 (↓ 1.0)	59.0
		Out-dist test	86.8 (↓ 1.4)	24.3

Table 8: Continual RL accuracy (%) of **QWEN3-1.7B** (Math → Coding → IF). The model is first trained on Math task, then on Coding task, and finally on IF task. Parenthesized arrows are performance changes from the previous stage. The performance before training on the corresponding task is highlighted in gray.

Train stage	Eval split	Math metric	Coding metric	IF metric
I	In-dist test	34.5	55.6	
	Out-dist test	76.2	34.9	
II	In-dist test	32.4 (↓ 2.1)	59.3	41.7
	Out-dist test	75.0 (↓ 1.2)	35.2	14.0
III	In-dist test	33.6 (↑ 1.2)	58.7 (↓ 0.6)	47.3
	Out-dist test	74.8 (↓ 0.2)	34.0 (↓ 1.2)	17.7

Table 9: Continual RL accuracy (%) of **QWEN3-4B** (Math → IF → Coding). The model is first trained on Math task, then on IF task, and finally on Coding task. Parenthesized arrows are performance changes from the previous stage. The performance before training on the corresponding task is highlighted in gray.

Train stage	Eval split	Math metric	IF metric	Coding metric
I	In-dist test	55.7	50.2	
	Out-dist test	86.8	21.3	
II	In-dist test	53.6 (↓ 2.1)	57.7	66.4
	Out-dist test	83.0 (↓ 3.8)	24.0	43.5
III	In-dist test	53.9 (↑ 0.3)	56.7 (↓ 1.0)	67.2
	Out-dist test	85.0 (↑ 2.0)	23.3 (↓ 0.7)	49.2

Table 10: Trainable parameters and gradient memory (FP32; 4 bytes per parameter).

Model	50% Attn	Full Attn	Full
QWEN3-1.7B	0.176B / 0.656 GiB	0.352B / 1.312 GiB	1.721B / 6.410 GiB
QWEN3-4B	0.330B / 1.231 GiB	0.661B / 2.462 GiB	3.739B / 13.930 GiB
MISTRAL-7B-INSTRUCT-V0.3	0.671B / 2.500 GiB	1.342B / 5.000 GiB	7.248B / 27.000 GiB

Table 11: Spearman distances ($d := 1 -$ Spearman correlation) between head rankings, where lower values indicate more similar rankings, identical rankings have distance 0, and two independent uniformly random rankings have expected distance 1. For Math and IF, within-task distances are averaged over the three pairwise comparisons among disjoint 1k-subsets of `DAPO-Math-17k` and `RLVR-IFEval`. For Coding, the within-task distance is computed between two disjoint 487-subsets of `MBPP`. Cross-task distances are computed using the training sets.

	Within-task			Cross-task		
Model	Math	IF	Coding	Math & IF	Math & Coding	IF & Coding
QWEN3-1.7B	0.025	0.026	0.051	0.044	0.084	0.081
QWEN3-4B	0.067	0.040	0.047	0.076	0.121	0.104

Table 12: Head-reset experiments after continual RL. After completing stage II, we reset the heads updated during stage I to their base-model values. We report the resulting test accuracy (%) changes. Stage-II performance that does not decrease after the reset is highlighted in orange. The results suggest that the adaptations learned on the disjoint head sets are functionally localized.

			Stage I task		Stage II task	
Model	Task sequence	Head allocation	In-dist	Out-dist	In-dist	Out-dist
QWEN3-1.7B	Math → Coding	Partition	−10.7	−6.0	+1.3	−4.0
QWEN3-1.7B	IF → Math	Partition	−16.4	−4.3	+1.1	+1.8
		Round robin	−11.6	−1.67	−3.3	−0.4
QWEN3-4B	Coding → IF	Partition	−4.0	−6.4	+0.4	+0.0
		Round robin	−0.8	+6.0	+0.6	+2.3
QWEN3-4B	Math → IF	Partition	−17.4	−7.2	+0.8	−2.3
		Round robin	−9.9	−5.8	+0.8	+0.3

B.3 LAYER DISTRIBUTIONS OF TOP-RANKED HEADS

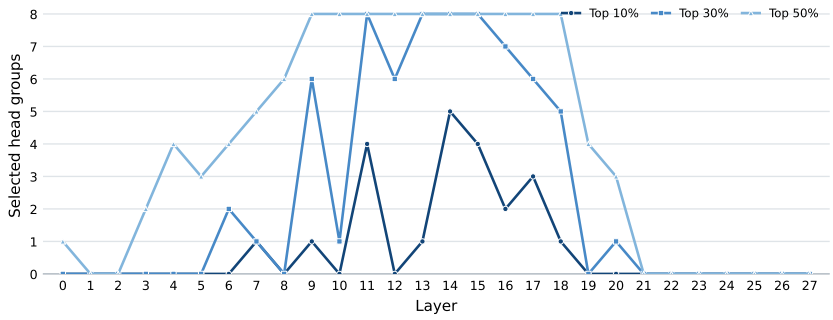


Figure 10: Layer-wise distribution of selected attention-head groups for **QWEN3-1.7B**, computed on the training set of Math.

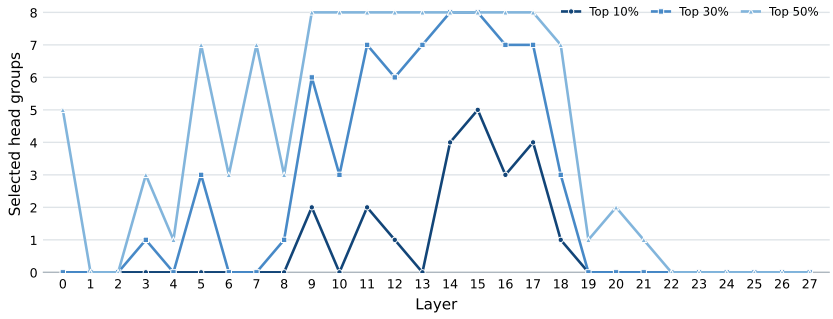


Figure 11: Layer-wise distribution of selected attention-head groups for **QWEN3-1.7B**, computed on the training set of Basic Coding.

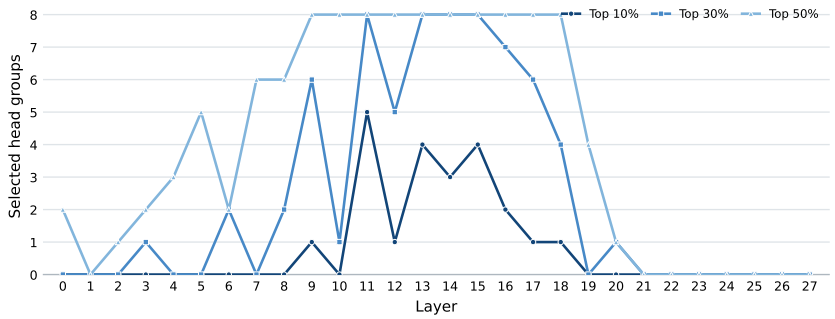


Figure 12: Layer-wise distribution of selected attention-head groups for **QWEN3-1.7B**, computed on the training set of Instruction-following.

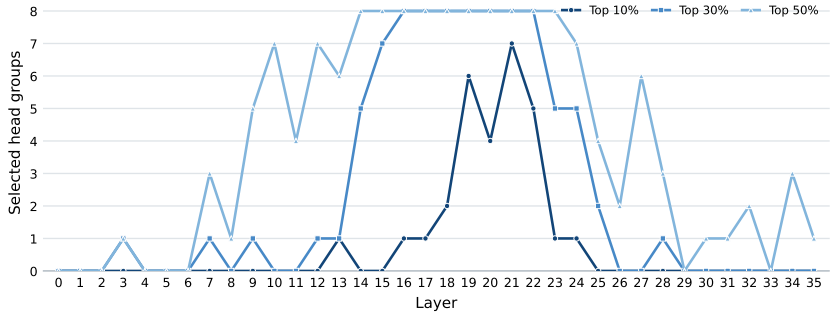


Figure 13: Layer-wise distribution of selected attention-head groups for **QWEN3-4B**, computed on the training set of Math.

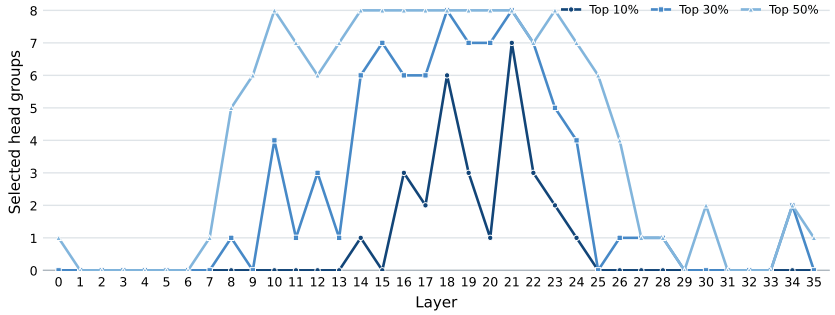


Figure 14: Layer-wise distribution of selected attention-head groups for **QWEN3-4B**, computed on the training set of Basic Coding.

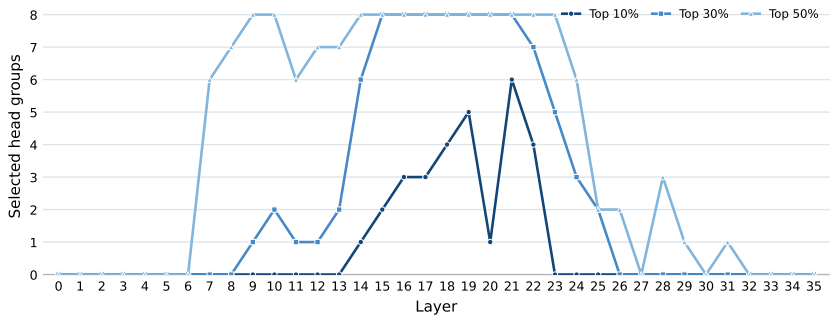


Figure 15: Layer-wise distribution of selected attention-head groups for **QWEN3-4B**, computed on the training set of Instruction-following.

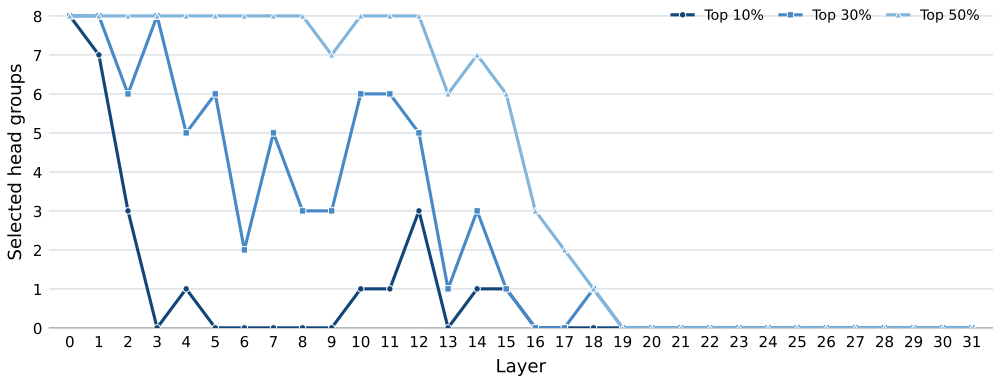
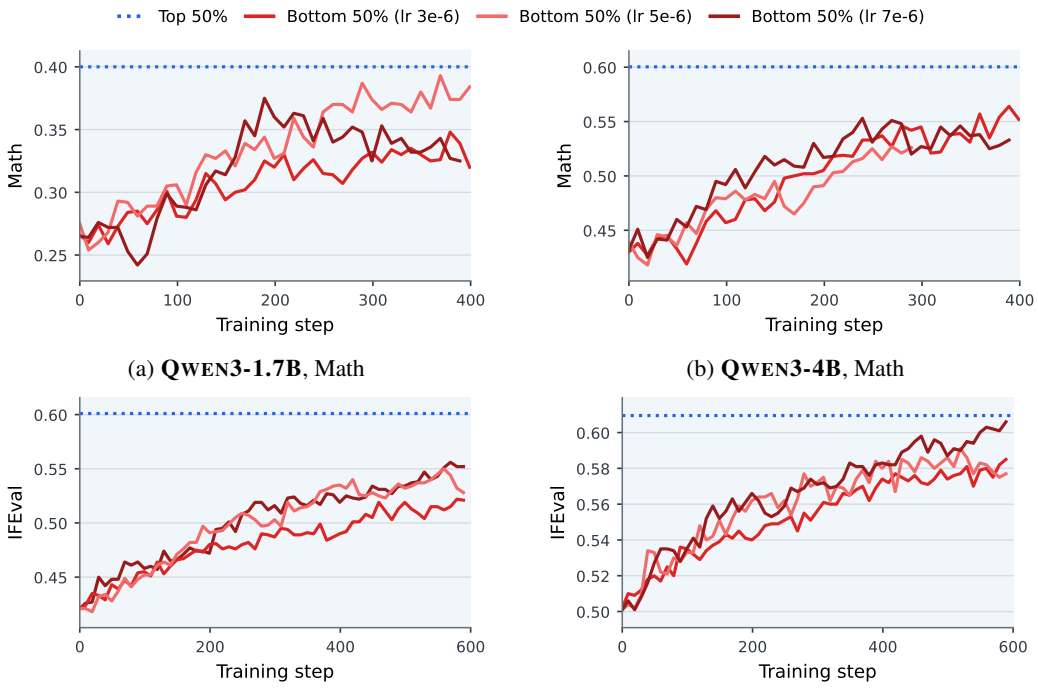


Figure 16: Layer-wise distribution of selected attention-head groups for **MISTRAL-7B-INSTRUCTION-V0.3**, computed on the training set of Instruction-following.

B.4 HYPERPARAMETER TUNING RESULTS



(a) **QWEN3-1.7B**, Math (b) **QWEN3-4B**, Math

(c) **QWEN3-1.7B**, IFEval (d) **QWEN3-4B**, IFEval

Figure 17: Sparse RL training accuracy (%) curves of **QWEN3-4B** for Math and IF tasks. Top 50% attention-head tuning is shown as a final-score horizontal line. The learning rate for top-50% attention-head tuning is fixed at 3×10^{-6}, while bottom-50% attention-head tuning is evaluated with learning rates of 3×10^{-6}, 5×10^{-6}, and 7×10^{-6}.

C IMPLEMENTATION DETAILS

Codebase. Our codebase is mainly based on miles (Miles Team, 2026) (`https://github.com/radixark/miles`). It is worth noting that our current sparse head tuning implementation is not yet fully optimized, though realizing its full efficiency benefits should be straightforward with modern training infrastructure. We are committed to releasing the code. The code was implemented with the assistance of Codex and Claude Code.

Datasets and benchmarks. For math, we use a 1k subset of `DAPO-Math-17k` (`https://huggingface.co/datasets/BytedTsinghua-SIA/DAPO-Math-17k`) for training, a disjoint 1k subset for in-distribution test, and `MATH-500` (`https://huggingface.co/datasets/HuggingFaceH4/MATH-500`) for out-of-distribution test. For instruction-following, we similarly use a 1k subset of `RLVR-IFEval` (`https://huggingface.co/datasets/allenai/RLVR-IFeval`) for training, a disjoint 1k subset for in-distribution test, and `IFBench` (`https://huggingface.co/datasets/allenai/IFBench_test`) for out-of-distribution test. For basic coding, we use `MBPP` (`https://huggingface.co/datasets/google-research-datasets/mbpp`) for training, `MBPP+` (`https://huggingface.co/datasets/evalplus/mbppplus`) for in-distribution test, and `APPS-intro-test` (`https://huggingface.co/datasets/codeparrot/apps`) for out-of-distribution test.

Models. Limited by computation resources, our base models are **QWEN3-1.7B** (Qwen Team, 2025) (`https://huggingface.co/Qwen/Qwen3-1.7B`), **QWEN3-4B** (Qwen Team, 2025) (`https://huggingface.co/Qwen/Qwen3-4B`), and **MISTRAL-7B-INSTRUCT-V0.3** (`https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3`).

Training hyper-parameters. Each rollout contains 16 prompts and 4 responses per prompt, with a global batch size of 64. We use temperature 1.0 and a maximum generation length of 3,072 tokens (2,048 for coding). Adam is used with learning rate 3×10^{-6} (1×10^{-6} for full training, 2×10^{-6} for **MISTRAL-7B**), weight decay 0.1, and $(\beta_1, \beta_2) = (0.9, 0.98)$. The policy-ratio clipping range is 0.2–0.28, with zero KL and entropy coefficients. Training runs for 400–600 rollout steps.

Evaluation hyper-parameters. We use greedy-decoding for evaluation. For each evaluation example, we generate a single response with a maximum length of 4,096 tokens, temperature 1.0, and top-k equal to 1. We report pass@1 accuracy using the task-specific verifier for each task.

Computation resources. Experiments are run on NVIDIA A100 and RTX A6000 GPUs, using four GPUs per job (eight for **MISTRAL-7B**), with a typical runtime of approximately one day.