PASs-MoE: Mitigating Misaligned Co-drift among Router and Experts via Pathway Activation Subspaces for Continual Learning

Anonymous ACL submission

Abstract

Continual instruction tuning (CIT) requires multimodal large language models (MLLMs) to adapt to a stream of tasks without forgetting prior capabilities. A common strategy is to isolate updates by routing inputs to different LoRA experts. However, existing LoRA-based mixture-of-experts(MoE) methods often jointly update the router and experts in an indiscriminate way, causing the router's preferences to co-drift with experts' adaptation pathways and gradually deviate from early-stage input–expert specialization. We term this as ***Misaligned Co-drift***, which blurs expert responsibilities and exacerbates forgetting. To address this, we introduce the ***pathway activation subspace (PASs)***, a LoRA-induced subspace that reflects which low-rank pathway directions an input activates in each expert, providing a capability-aligned coordinate system for routing and preservation. Based on PASs, we propose a fixed-capacity PASs-based MoE–LoRA method with two components: PAS-guided Reweighting, which calibrates routing using each expert's pathway activation signals, and PAS-aware Rank Stabilization, which selectively stabilizes rank directions important to previous tasks. Experiments on CIT benchmark show that our approach consistently outperforms a range of conventional continual learning baselines and MoE–LoRA variants in both accuracy and anti-forgetting without increasing model parameters. Source codes will be released upon acceptance.

1 Introduction

Multimodal large language models (MLLMs) have recently become a strong backbone for vision–language understanding and generation across diverse applications (Achiam et al., 2023; Liu et al., 2024, 2023; Bai et al., 2025). However, task requirements and data distributions change over time in real-world deployments. Continual instruction tuning (CIT) studies how to sequentially adapt

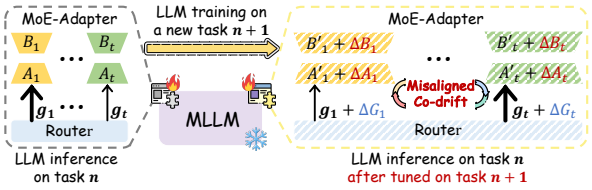


Figure 1: Illustration of the ***Misaligned Co-drift*** in existing MoE-LoRA methods, where the thickness of arrow indicates the sampling probability of router. Here, the drift in router assignments and the internal drift within expert parameters for the same task jointly drive this issue, thereby exacerbating catastrophic forgetting.

MLLMs on a stream of tasks with supervised instruction–response data while retaining previously acquired capabilities (Chung et al., 2022).

Serving as the most widely-used Parameter-efficient fine-tuning method for CIT (Sung et al., 2022; Zhang et al., 2021; Hu et al., 2022), the standard practice of sharing a single Low-rank adaptation (LoRA) module across tasks forces MLLMs to concentrate updates within the same low-rank subspace, increasing interference and exacerbating forgetting. To address this issue, recent studies combine the unique strengths of mixture-of-experts (MoE) architectures with LoRA. For instance, DDAS (Yu et al., 2024), D-MoLE (Ge et al., 2025) and RMoE (Huai et al., 2025) attempt to expand MLLMs with additional experts and routing modules, which typically incur ever-growing parameters and storage, thereby increasing training and deployment complexity (Fedus et al., 2022). In contrast, MoELoRA (Chen et al., 2024) routes each input to a fixed pool of LoRA experts, reducing forgetting during continual training without increasing model capacity. Nevertheless, its training procedure typically entails indiscriminate joint optimization of both the router and the expert parameters. As tasks accumulate within CIT, both the router and expert parameters evolve to accommodate new tasks, yet their optimization directions are not fully aligned. This ***Misaligned Co-drift*** may undermine

the input-expert specialization established in earlier stages: *Router updates can reroute the same input away from its previously associated expert, while expert updates alter that expert's responses on samples from previous tasks.*

Drilling down into the underlying mechanism of ***Misaligned Co-drift***, we notice that existing MoE-LoRA variants fail to anchor routing and preservation in a capability-aligned coordinate system defined by each expert's low-rank adaptation pathway. Under this perspective, we revisit the formulation of MoE-LoRA (Jacobs et al., 1991; Shazeer et al., 2017), especially the structural property of LoRA as $\Delta y(x) = B(Ax)$. Intuitively, the down-projection A specifies the set of input directions an expert can respond to through its low-rank pathway, while the up-projection B combines these coordinates into the additive correction within such low-rank coordinate system. Based on this observation, we creatively propose the notion of ***P**athway **A**ctivation **S**ubspace (**PASs**)* $\mathcal{S} = \mathrm{span}(A^\top)$. Unlike prior notions of "activation subspace" which are typically derived from intermediate activations, the proposed *PASs* $\mathcal{S}$ is induced by LoRA's down-projection A, where the low-rank activation Ax provides a capability-tied coordinate system for aligning routing with parameter updates and selectively stabilizing task-critical rank directions, thus mitigating catastrophic forgetting in CIT.

Building on this notion, we propose a **PASs-based MoE-LoRA method** to mitigate catastrophic forgetting in CIT by aligning expert routing and preservation with each expert's low-rank adaptation pathway. It includes two key designs, *PASs-guided Reweighting (PASs-RW)* and *PASs-aware Rank Stabilization (PASs-RS)*, both grounded in the ***PASs***. Specifically, the proposed *PASs-RW* modulates each expert's contribution according to the input's activation within the corresponding pathway activation subspace. Here, each expert's routing weight is driven by its own low-rank pathway rather than by a separate router space, thereby alleviating *misaligned co-drift*. Nevertheless, even with the improved routing alignment, the experts selected by the router still must adapt to new tasks, and their low-rank pathways may drift under the sequential updates, potentially leading to forgetting. To address this expert-side drift, we further design *PASs-RS* to selectively constrain directions that matter for previous tasks by utilizing the same *PASs* signal. This rank-level stabilization reduces forgetting while preserving plasticity.

Our contributions are threefold: (I) We introduce a *pathway activation subspace* perspective for CL, modeling the low-dimensional update directions induced by LoRA as task-dependent pathway activation subspaces to analyze knowledge update and retention in MoE-LoRA. (*cf.* Section 3) (II) We propose implicit self-routing and rank-level stability regularization for MoE-LoRA under a fixed parameter setting. (*cf.* Section 4) (III) Experiments demonstrate consistent improvements over a range of traditional CL algorithms and advanced MoE-related methods in both forgetting mitigation and downstream task performance (*cf.* Section 5).

2 Related Works

Continual Instruction Tuning. Continual Instruction Tuning (CIT) incrementally aligns multimodal large language models (MLLMs) over a task stream, training on disjoint instruction–response pairs while aiming to retain prior capabilities (He et al., 2023; Chen et al., 2024; Guo et al., 2025). With limited or no access to past data, CIT faces the stability–plasticity trade-off and catastrophic forgetting. Existing mitigation strategies include importance constraints (Kirkpatrick et al., 2017; Li and Hoiem, 2017; Aljundi et al., 2018), relational knowledge distillation (Zhang et al., 2024), isolated adaptation (Razdaibiedina et al., 2023; Wang et al., 2022; Cui et al., 2025; Wang et al., 2023), and replay-based strategies (Rolnick et al., 2019a,b; Yan et al., 2021; Wang et al., 2024; He et al., 2024) approximation of past distributions. CIT in MLLMs is further challenged by cross-modal alignment drift, which destabilizes vision-language mappings and exacerbates task interference.

Activation Subspace. Activation subspaces provide low-dimensional signals for interpreting and constraining model behavior (Kim et al., 2018). CAV-style approaches represent concepts as activation directions (Wenkmann and Garreau, 2025; Huang et al., 2024), and subspace-based continual learning reduces interference by separating or constraining updates within task-relevant subspaces (Farajtabar et al., 2020; Saha et al., 2021; Chaudhry et al., 2020; Magistri et al., 2024; Roy et al., 2023). These subspaces are typically statistics-driven and not tied to architectural units. We instead define a LoRA-parameter-induced pathway activation subspace that binds subspace signals to low-rank channel functionality, enabling routing alignment and expert responsibility preservation.

Mixture of Experts. Sparse Mixture-of-Experts (MoE) has been explored for continual vision-language tuning, including MoE adapters and MoE-LoRA variants (Chen et al., 2024; Zhang et al., 2025) that dynamically compose LoRA experts, using rank-level micro-experts to retain pretraining knowledge (Zhao et al., 2025). However, most rely on explicitly defined expert pools and a shared routing space, which can accumulate misalignment between routing decisions and expert functionality. Our activation-subspace view couples routing with LoRA expert pathways to reduce routing drift and responsibility drift.

3 Preliminary

3.1 Problem Formulation

We study continual instruction tuning (CIT) for Multimodal Large Language Models (MLLMs). Let $\{\mathcal{D}_t\}_{t=1}^{T}$ denote a stream of tasks, where each stage-t dataset $\mathcal{D}_t = \{(v_i^{(t)}, x_i^{(t)}, y_i^{(t)})\}_{i=1}^{N_t}$ consists of multimodal instruction-response pairs. Here, $v_i^{(t)}$ denotes a visual input (e.g., an image), $x_i^{(t)}$ a textual instruction, and $y_i^{(t)}$ the target response. Starting from a pretrained multimodal backbone, training proceeds sequentially: at stage t, the model is updated using only the current dataset $\mathcal{D}_t$.

At each stage, the learning objective minimizes the empirical risk on the current task,

$$\min_{\theta_t} \mathbb{E}_{(v,x,y)\sim\mathcal{D}_t}\big[\mathcal{L}(f(v,x;\theta_t),y)\big], \qquad (1)$$

where $f(\cdot;\theta_t)$ denotes the model at stage t. Following standard instruction tuning for MLLMs, we adopt the autoregressive negative log-likelihood loss. Under parameter-efficient fine-tuning, θ_t denotes the trainable adapter parameters, while the pretrained backbone remains frozen.

3.2 Decomposition of MoE-LoRA

Parameter-efficient fine-tuning (PEFT) adapts a pretrained model by introducing a small number of trainable parameters while keeping the backbone weights frozen. A widely used PEFT approach is Low-Rank Adaptation (LoRA), which augments a linear layer with a low-rank update. Given a weight matrix $W \in \mathbb{R}^{d_{\text{out}}\times d_{\text{in}}}$, LoRA defines

$$W' = W + \Delta W, \quad \Delta W = BA, \qquad (2)$$

where $A \in \mathbb{R}^{r\times d_{\text{in}}}$ and $B \in \mathbb{R}^{d_{\text{out}}\times r}$ are trainable matrices, and $r \ll \min(d_{\text{in}}, d_{\text{out}})$.

MoE-LoRA extends LoRA by instantiating multiple LoRA experts and combining their updates with a router. Consider E experts $\{(A_e, B_e)\}_{e=1}^{E}$ attached to a linear layer, and let $h \in \mathbb{R}^{d_{\text{in}}}$ denotes the hidden features input to this layer. The router $g(\cdot)$ maps h to expert logits, from which mixture weights $\{\pi_e(h)\}_{e=1}^{E}$ are derived (e.g., dense normalization or sparse top-k selection). The resulting input-dependent update to the weight matrix is

$$\Delta W(h) = \sum_{e=1}^{E} \pi_e(h)\, B_e A_e h. \qquad (3)$$

In continual instruction tuning, PEFT optimizes the expert parameters $\{(A_e, B_e)\}_{e=1}^{E}$ together with the router parameters in g.

3.3 Definition of Pathway Activation Subspace

We revisit the low-rank structure underlying MoE-LoRA. From the perspective of a single expert e, the LoRA branch adds an incremental output to the linear layer in the form

$$\Delta y_e(h) = B_e(A_e h). \qquad (4)$$

This factorization exposes an intermediate response vector $z_e(h) = A_e h$, which is a r-dimensional linear response induced by A_e from the input h. The j-th entry $z_{e,j}(h)$ can be interpreted as the response coefficient along the j-th rank direction, capturing how strongly expert e reacts to the input through that direction. The matrix B_e then maps this r-dimensional response back to the output space, yielding the expert's incremental contribution to the layer output. Motivated by this structure, we define the PASs of expert e as

$$\mathcal{S}_e = \text{span}(A_e^{\top}). \qquad (5)$$

This subspace is induced by the row space of A_e and characterizes the set of input-side directions that the expert uses to generate its low-rank response $z_e(h)$. In this sense, $\mathcal{S}_e$ provides a parameter-grounded reference space that directly links the expert's input-side response pattern to its A_e. Building on this view, we develop PASs-RW and PASs-RS in Section 4.

4 Methodology

4.1 Overview

We study continual instruction tuning (CIT) under a fixed-structure MoE-LoRA setting, where a single

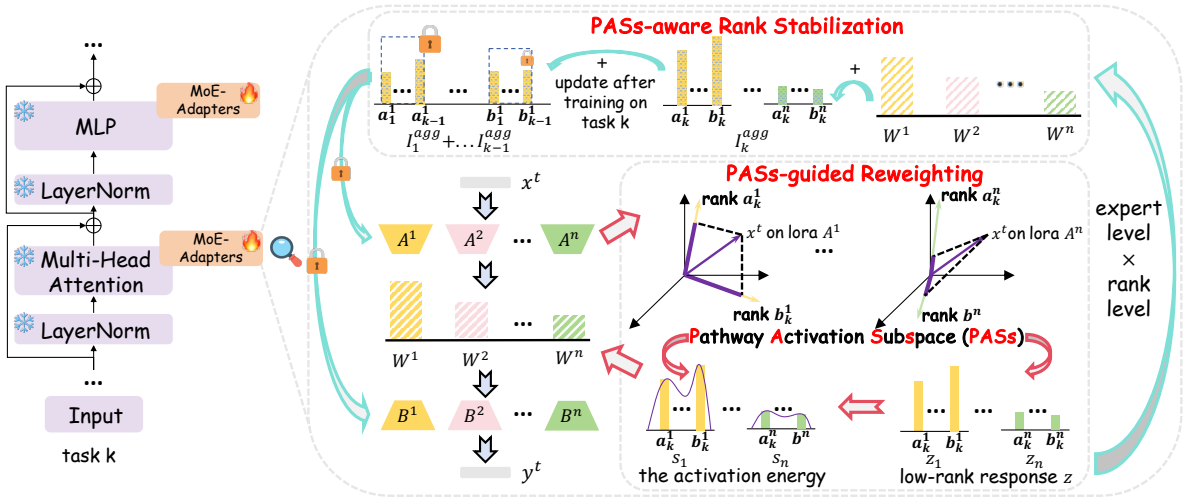


Figure 2: **Overview of the proposed PASs-based MoE-LoRA method.** We consider continual instruction tuning with a *fixed* set of LoRA experts (E). Each expert defines a pathway activation subspace (PASs) induced by its LoRA down-projection A_e, which we use to couple expert routing and rank-level preservation. The two components, **PASs-guided Reweighting (PASs-RW)** and **PASs-aware Rank Stabilization (PASs-RS)**, are highlighted in red.

model learns tasks sequentially without expanding the expert pool. Figure 2 illustrates the overall pipeline. Given an input x_t, the backbone produces a hidden representation h. Each LoRA expert e defines a capability-tied pathway activation subspace (PASs) induced by its down-projection A_e, which yields a low-rank activation z_e. We use this PASs signal as a shared anchor to jointly design routing and preservation, targeting the *Misaligned Co-drift* discussed in the introduction.

Based on this view, we propose two components. First, **PASs-RW** computes mixture weights from the expert-specific low-rank activation, so that expert assignment is driven by the expert pathway itself rather than by an independent router space. Second, **PASs-RS** accumulates PASs-based activation statistics during sequential training and selectively stabilizes historically important rank directions. This reduces expert-side drift and, because routing depends on the same PASs signal, helps maintain more consistent routing behavior for earlier-task inputs. The following sections detail PASs-RW 4.2 and PASs-RS, respectively.

4.2 PASs-guided Reweighting

PASs-RW mitigates *Misaligned Co-drift* by tying routing to each expert's functional response. Prior MoE-LoRA variants often learn routing in a shared space that is decoupled from the experts' low-rank pathways, so routing policies and expert parameters can drift inconsistently as tasks accumulate. We instead derive routing weights from the low-rank activation $z_e = A_e h$, which measures how strongly expert e responds to the input along its LoRA pathway.

We define the activation energy of expert e as

$$s_e(h) = \frac{1}{r}\|A_e h\|_2^2, \tag{6}$$

and compute mixture coefficients by softmax normalization:

$$\pi_e(h) = \frac{\exp(s_e(h))}{\sum_{e'=1}^{E}\exp(s_{e'}(h))}. \tag{7}$$

Grounding $\pi_e(h)$ in z_e anchors routing in a capability-tied signal, coupling routing behavior with expert evolution under sequential training.

4.3 PASs-aware Rank Stabilization

While PASs-RW couples routing to the expert pathway, the selected experts still need to adapt to new tasks, and their low-rank parameters may drift under sequential updates, which can induce forgetting. The PAS formulation makes this drift analyzable at the rank level. Specifically, we define the low-rank response as $z_e = A_e h$, where $z_e \in \mathbb{R}^r$, which admits a coordinate-wise interpretation: each coordinate corresponds to the activation on one rank direction of A_e. Let $a_{e,k} \in \mathbb{R}^d$ denote the k-th row of A_e, then the k-th component of z_e is

$$z_{e,k} = a_{e,k}^\top h. \tag{8}$$

This view enables rank-level importance estimation within each expert. We define the importance of rank direction (e,k) on task t as

$$\begin{aligned} I_{e,k}(t) &\triangleq \mathbb{E}_{h\sim\mathcal{D}_t}\left[\pi_e(h)\,(z_{e,k})^2\right] \\ &= \mathbb{E}_{h\sim\mathcal{D}_t}\left[\pi_e(h)\,(a_{e,k}^\top h)^2\right]. \end{aligned} \tag{9}$$

Method	Math QA		Arts VQA		Math VQA		Econ. QA		Med. VQA		OCR VQA		Sci. VQA		AP	BWT
	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget		
SeqFT	37.89	−19.75	5.52	−32.35	39.91	−11.97	64.42	−5.74	15.70	−20.40	7.19	−16.67	**84.07**	–	36.39	−15.27
SeqLoRA	0.00	−54.43	6.28	−24.81	27.75	−22.18	38.44	−35.15	24.70	−12.54	16.96	−5.53	83.52	–	28.24	−22.09
Replay	40.64	−12.56	25.44	−7.97	34.37	−14.09	60.15	−11.83	**31.79**	−2.34	**23.68**	−3.21	78.25	–	42.05	−8.29
EWC	16.78	−36.18	6.83	−26.42	25.68	−26.66	37.40	−36.19	23.63	−11.25	17.49	−4.66	83.52	–	30.19	−20.20
LwF	18.98	−34.71	6.45	−27.01	26.80	−25.31	36.51	−34.66	23.61	−9.94	17.38	−8.10	81.68	–	30.20	−19.96
MAS	19.48	−34.95	7.87	−25.65	28.48	−23.63	38.48	−35.11	21.58	−13.37	17.78	−6.80	78.74	–	30.34	−19.36
O-LoRA	24.58	−29.85	9.87	−22.61	30.79	−20.98	38.48	−34.01	22.82	−12.85	18.64	−5.92	82.44	–	32.52	−19.46
HiDe-llava	26.85	−25.12	11.84	−22.32	31.58	−16.77	49.29	−9.08	8.22	−20.75	14.67	−3.07	29.97	–	24.63	−16.16
DDAS	44.83	−7.88	20.87	−5.36	37.40	−1.25	58.77	−13.51	23.74	−0.34	2.35	−0.07	74.46	–	37.49	−4.06
MoE-lora (Top-k)	35.68	−15.06	6.78	−24.18	33.58	−17.69	39.80	−27.18	22.67	−10.89	14.57	−3.41	73.56	–	32.38	−15.34
MoE-lora (Softmax)	42.98	−11.45	35.89	+2.65	40.84	−11.61	56.10	−15.47	29.24	−6.04	18.89	−4.57	79.57	–	43.36	−6.64
Ours	**49.52**	−5.90	**43.22**	+10.43	**44.70**	−6.50	**66.13**	−6.05	29.99	−5.38	21.95	−1.63	83.73	–	**48.46**	−2.15

Table 1: Comparison with traditional methods and MoE-LoRA-based methods on MLLM-CTBENCH. Best and second-best results for Acc and AP are marked in **bold** and underline.

and maintain the importance over past tasks by

$$I_{e,k}^{\text{agg}}(t-1) \triangleq \sum_{t'=1}^{t-1} I_{e,k}(t'). \tag{10}$$

When learning task t, we impose a weighted stabilization constraint on historically important rank directions:

$$\mathcal{L}_{\text{stab}} = \sum_{e=1}^{E} \sum_{k=1}^{r} w_{e,k} \left\| a_{e,k}^{(t)} - a_{e,k}^{(t-1)} \right\|_2^2. \tag{11}$$

where $w_{e,k}$ is a normalized and clipped version of the aggregated importance $I_{e,k}^{\text{agg}}(t-1)$ to ensure scale invariance and numerical stability. The overall objective is $\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda \mathcal{L}_{\text{stab}}$. By concentrating stabilization on rank directions that are more important to past tasks, PASs-RS reduces expert-side drift that harms prior capabilities. Moreover, since PASs-RW derives routing weights from the same pathway response, stabilizing task-critical rank directions also helps maintain more consistent routing for earlier-task inputs during later training.

5 Experiments

5.1 Experimental Setting

Datasets We evaluate our method on MLLM-CTBench, a continual instruction tuning benchmark for multimodal large language models. Following the official protocol, we perform sequential training on the default task sequence without revisiting past data. MLLM-CTBench involves heterogeneous tasks with substantial distribution shifts, posing a challenging testbed for continual adaptation. Detailed dataset descriptions and the exact evaluation protocol are provided in Appendix A.1.

Baselines We compare against representative continual learning methods for MLLMs, including LwF, EWC, L2P, O-LoRA, and HiDe-LLaVA, and also include MoE-LoRA variants (MoELoRA and DDAS). We further report zero-shot and multi-task fine-tuning results as the lower and upper bounds. Full training recipes and hyperparameters are provided in Appendix A.1.

Implementation Details We adopt LLaVA-v1.5-7b as the base model and use CLIP-L/14-336 (Radford et al., 2021) to extract visual and textual features. Following LLaVA's LoRA fine-tuning protocol, we insert LoRA modules into all linear layers of the language model and set the rank to $r = 128$. Unless otherwise specified, we use $E = 6$ experts and set the stability regularization coefficient to $\lambda = 5 \times 10^{-4}$. Under the continual learning setting, we perform sequential training strictly following the task order of each benchmark, with 3 epochs per task on MLLM-CTBENCH, using a warmup ratio of 0.03. All methods are trained with a batch size of 12 on NVIDIA H20 GPUs, and all experiments are conducted with a fixed random seed of 42 for reproducibility.

Metrics We report the final performance on each task t after training on the full stream, $\text{Acc}_t^{\text{final}}$, and its task-wise forgetting $\text{Forget}_t = \text{Acc}_t^{\text{after } t} - \text{Acc}_t^{\text{final}}$, where $\text{Acc}_t^{\text{after } t}$ is the accuracy measured immediately after learning task t. We summarize overall continual learning performance with $\text{AP} = \frac{1}{T}\sum_{t=1}^{T} \text{Acc}_t^{\text{final}}$ and $\text{BWT} = \frac{1}{T}\sum_{t=1}^{T} \text{Forget}_t$.

5.2 Main Results

The main results on MLLM-CTBENCH are reported in Table 1, while results on an alternative task order are provided in Appendix A.2. Our method achieves the best overall performance, im-

Method	MoE	PASs-RW	PASs-RS	Math QA		Arts VQA		Math VQA		Econ. QA		Med. VQA		OCR VQA		Sci. VQA		AP	BWT
				Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget		
MoE-LoRA (top-k)	Top-k	x	x	35.68	−15.06	6.78	−24.18	33.58	−17.69	39.80	−27.18	22.67	−10.89	14.57	−3.41	73.56	–	32.38	−15.34
MoE-LoRA (softmax)	All	x	x	42.98	−11.45	35.89	+2.65	40.84	−11.61	56.10	−15.47	29.24	−6.04	18.89	−4.57	79.57	–	43.36	−6.64
+ PASs-RW	All	✓	x	49.99	−5.18	39.11	+6.26	42.70	−8.27	60.90	−9.87	27.79	−8.74	20.27	−3.86	83.41	–	46.31	−4.24
Ours (Full)	All	✓	✓	**49.52**	−5.90	**43.22**	+10.43	**44.70**	−6.50	**66.13**	−6.05	**29.99**	−5.38	**21.95**	−1.63	**83.73**	–	**48.46**	−2.15

Table 2: Structural ablation of our method on continual instruction learning. We compare two MoE-LoRA baselines, using either softmax weighting over all experts (All) or Top-k expert selection, and then progressively add PASs-RW and PASs-RS, where the combination corresponds to our method. Best and second-best results for Acc and AP are marked in **bold** and underline.

proving the final Average Performance (AP) and reducing forgetting measured by Backward Transfer (BWT) compared with conventional continual learning baselines. It also surpasses MoE-LoRA variants, including the two representative design lines summarized in the Introduction. Notably, on MLLM-CTBENCH, our final AP exceeds that of the second-best method by **5.1%**. We observe consistent gains under the alternative task order, where our AP improves by **9.46%** over the second-best method (Appendix A.2).

5.3 Ablation Study

Table 2 reports the structural ablation results under a fixed MoE-LoRA parameter budget. Since the Top-k routing baseline performs substantially worse in the continual instruction tuning setting, we treat it only as a reference point. Accordingly, we introduce our designs on top of the softmax-based MoE-LoRA baseline and progressively ablate the two core components of our framework.

5.3.1 Structural Ablation

Table 2 presents structural ablations under a fixed MoE-LoRA parameter budget. Since the Top-k routing baseline performs notably worse in continual instruction tuning, we include it only as a reference point. All ablations are built upon a softmax-based MoE-LoRA baseline.

We ablate the two key components of our framework to quantify their individual contributions. Enabling *PASs-RW* alone consistently improves the final performance over the MoE-LoRA baseline, suggesting that activation-subspace signals provide an effective basis for expert reweighting under sequential training. This observation aligns with our motivation in the Introduction that routing should be grounded in expert-specific low-rank pathways, rather than learned in a separate shared routing space. We defer a closer examination of routing dynamics to the dedicated analysis section 5.4.

Adding *PASs-aware Rank Stabilization (PASs-RS)* on top of reweighting yields additional gains, indicating that selectively constraining task-relevant rank directions can further improve retention while preserving adaptation to new tasks. We further compare our stabilization with a random regularization scheme; the corresponding results are reported in Appendix A.3. Overall, the comparison supports the importance of using activation-subspace signals to decide which rank directions to stabilize.

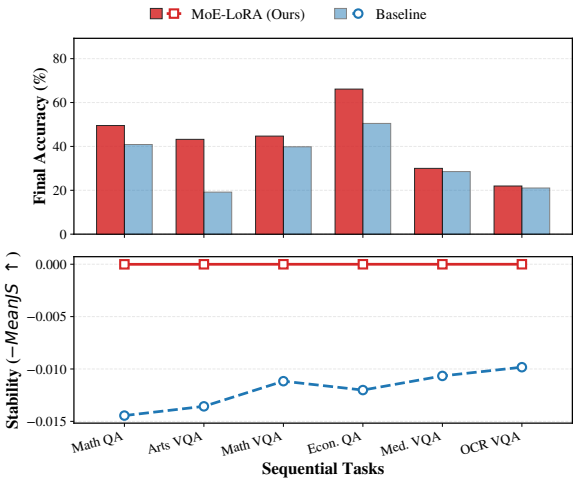


Figure 3: **Comparison of final performance and router stability.** The top and bottom panels display the final accuracy and the corresponding router stability across the task stream, respectively. Stability is quantified as the negative Mean Jensen–Shannon (JS) divergence.

5.3.2 Hyperparameter Ablation

We further study sensitivity to two key hyperparameters: the number of experts E (Table 3) and the stabilization strength λ (Fig. 4). For the expert count, the performance is non-monotonic: our method achieves the best final AP at a moderate E (peaking at E=6), while larger E brings marginal or slightly reduced gains (e.g., E=8). This trend suggests that there exists a favorable capacity regime under a fixed training budget, whereas further increasing E

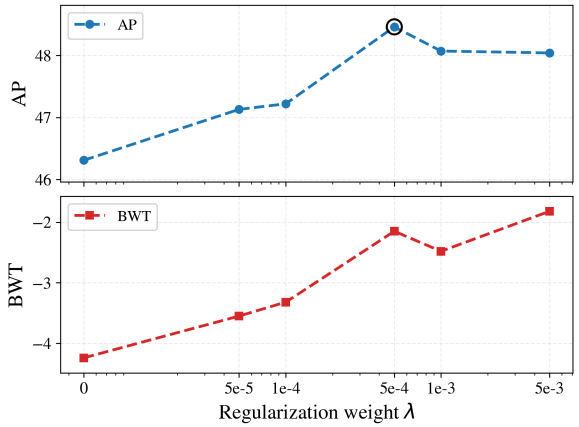


Figure 4: Ablation results on the regularization strength λ for PASs-MoE in MLLM-CTBENCH. The upper panel reports AP, while the lower panel shows BWT.

Method	Experts	Train Time	Infer Time	AP	BWT
LoRA	1	120.14	129.87	28.24	−22.09
MoE-LoRA	2	130.66	143.81	39.44	−10.18
Ours	2	140.39	132.78	46.61	−2.44
MoE-LoRA	4	130.48	131.15	41.11	−7.91
Ours	4	140.29	148.59	44.85	−5.27
MoE-LoRA	6	130.65	127.56	43.36	−6.64
Ours	6	140.89	132.84	**48.46**	−2.15
MoE-LoRA	8	130.60	130.91	43.58	−6.79
Ours	8	140.48	145.71	47.67	−2.13

Table 3: Effect of the number of experts on sequential multimodal continual learning. We compare our method against MoE-LoRA with softmax weighting across different expert counts.

may yield diminishing returns.(Table 3)

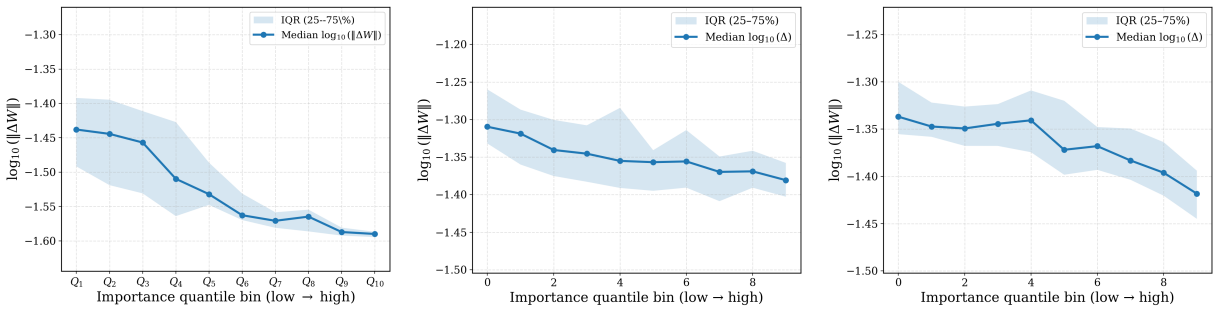
For the stabilization strength λ, we observe a clear stability-plasticity trade-off(Fig. 4). As λ increases, forgetting is gradually alleviated, reflected by the monotonic increase of BWT. Meanwhile, the final performance (AP) exhibits a non-monotonic trend: it improves under mild regularization but starts to drop when λ becomes larger. This is expected because stronger regularization increasingly constrains parameter updates, reducing the model's ability to adapt to new tasks; as a result, forgetting is reduced at the cost of diminished task learning, leading to lower AP at high λ. Overall, these results suggest that a moderate stabilization strength provides the best balance between retaining previous knowledge and preserving sufficient plasticity for continual adaptation.

5.4 In-depth Analysis

Routing drift correlates with old-task degradation. To examine whether the specialization between old-task inputs and experts is preserved under sequential updates, we track the gating distribution $G(x)$ for a fixed set of old-task inputs at two checkpoints: immediately after learning the corresponding task and after completing the full task stream. We quantify the change using distributional distances (e.g., Jensen-Shannon divergence), averaged over layers and samples. As shown in Fig. 3, the MoE-LoRA baseline exhibits substantially larger routing drift, which coincides with lower final accuracy on old tasks, whereas our method yields consistently smaller drift while retaining higher final performance. These results support the motivation in Section 1 that *misaligned co-drift* can undermine previously established input-expert specialization.

Activation-derived importance is highly non-uniform across ranks. Rank importance is highly non-uniform. PASs-RS uses activation statistics to localize which rank channels within each expert are most critical for retaining prior-task behavior. Intuitively, if a channel (e, j) consistently yields a stronger low-rank response on earlier tasks, it is more likely to encode functionality that should be preserved and thus warrants stronger stabilization. To quantify this effect, we continuously record and accumulate an importance score $I_{\text{agg}}[e, j]$ based on the low-rank response $z_e = A_e h$. As shown in Fig. 7, I_{agg} exhibits a sparse, concentrated pattern over the expert-rank grid rather than a uniform spread across rank directions, indicating that prior-task behavior is dominated by a small subset of channels. This pattern aligns with our PASs perspective that LoRA adaptation operates through structured, capability-tied pathways, making $I_{\text{agg}}[e, j]$ a principled rank-level weight for selectively stabilizing the most critical directions.

Rank stabilization reshapes updates toward low-importance directions. Building on the non-uniform importance, we examine whether rank-level stabilization preferentially protects high-importance directions. For each layer, we sort rank directions by I_{agg}, split them into quantile bins (e.g., deciles), and compute the median of $\log \|\Delta W\|$ with its interquartile range (IQR) within each bin. As shown in Fig. 5 for representative layers (Layer 1/15/29), the median update magnitude generally decreases with increasing importance, indicating that stabilization suppresses drift on high-importance directions while leaving more flexibil-



(a) Layer 1 (low-level) (b) Layer 15 (mid-level) (c) Layer 29 (top-level)

Figure 5: **Update magnitude versus old-task importance across representative layers.** Each plot bins coefficients by the previous task's aggregated importance I_{agg} (low $\rightarrow$ high) and reports the median (with interquartile range) of the log update norm $\log \|\Delta W\|$ within each bin.

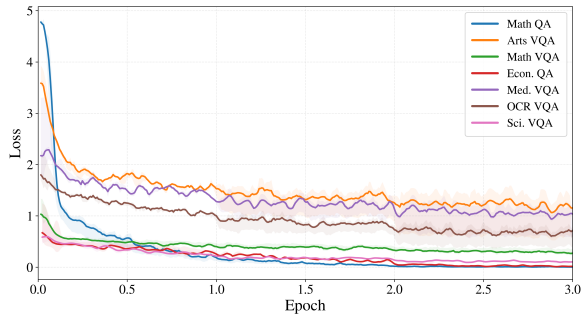


Figure 6: Training loss curves across tasks.

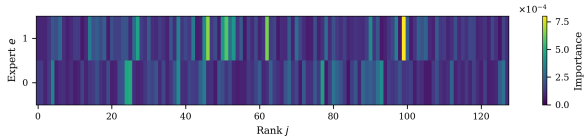


Figure 7: **Aggregated rank importance over the task stream.** We show the cumulative importance map I_{agg} over the sequential training procedure, and visualize the two most important experts for clarity. Importance concentrates on a small subset of all rank channels, suggesting that a few expert-rank directions dominate contributions across different tasks.

ity to low-importance ones. The trend strength varies across layers, consistent with layer-wise differences in representational roles and update dispersion: when importance is flatter or updates are more diffuse, the coupling between I_{agg} and ΔW becomes weaker.

5.5 Training Dynamics Across Tasks

To evaluate the robustness of our proposed PASs-MoE, we visualize its training loss curves on all downstream tasks from MLLM-CTBENCH. From figure 6, we can observe clear task-dependent convergence rates, where some tasks start with higher loss and decrease more slowly than others. This disparity is consistent with the strong heterogeneity of MLLM-CTBENCH in terms of modality dependence and reasoning difficulty (Guo et al., 2025).

Meanwhile, we can also notice that the optimization of our PASs-MoE on MLLM-CTBENCH remains relatively stable throughout the training procedure. That is, we donot observe any loss explosion phenomenon, and the loss of each task continues to decrease steadily during the later stages of training rather than rebounding. These discoveries suggest that our sequential training does not suffer from severe optimization instability, and we do not observe evident overfitting under the adopted training protocol.

6 Conclusion

We study CIT of MLLMs under a fixed-capacity Mixture-of-Experts adapter setting, where indiscriminate joint updates of routers and experts can trigger *Misaligned Co-drift* and erode early input-expert specialization. We introduce *PASs* induced by LoRA down-projections to provide a capability-tied coordinate system for aligning routing and preservation. Based on PASs, we propose a PASs-based MoE-LoRA method with PASs-guided Reweighting and PASs-aware Rank Stabilization, which couple routing to low-rank pathway responses and selectively stabilize historically important rank directions, respectively. On MLLM-CTBENCH, our method improves both final performance and forgetting, outperforming the second-best approach by **5.1%** AP without increasing model capacity. We hope PASs offers a principled tool for capability-aligned routing and fine-grained preservation in CIT.

Limitations

This work has several limitations. First, we focus on fixed-capacity MoE-LoRA in continual instruction tuning; the conclusions may not directly carry over to settings with capacity expansion, replay access, or substantially different continual learning paradigms. Second, PASs leverages the structural property of LoRA (the down-projection A and the induced low-rank response $A_e h$), so extending the framework to other PEFT forms may require redefining a capability-aligned coordinate system and corresponding routing signals. Third, PASs-RW uses low-rank energy as a proxy for input–expert compatibility; while capability-related, its robustness and calibration under more severe distribution shifts or more complex instructions deserve further study. For PASs-RS, importance is estimated from activation statistics and can be noisy when task data are scarce or highly conflicting; it also requires maintaining historical statistics and tuning an additional regularization hyperparameter, which introduces extra implementation and tuning overhead. Finally, our evaluation primarily relies on AP/BWT, which captures overall performance and forgetting, but more direct measurements of routing mismatch, stronger interpretability analyses of evolving expert responsibilities, and evaluations under realistic online distribution dynamics remain important future directions.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. 2018. Memory aware synapses: Learning what (not) to forget. *Preprint*, arXiv:1711.09601.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *Preprint*, arXiv:2502.13923.

Arslan Chaudhry, Naeemullah Khan, Puneet Dokania, and Philip Torr. 2020. Continual learning in low-rank orthogonal subspaces. *Advances in Neural Information Processing Systems*, 33:9900–9911.

Cheng Chen, Junchen Zhu, Xu Luo, Hengtao Shen, Lianli Gao, and Jingkuan Song. 2024. Coin: A benchmark of continual instruction tuning for multimodel large language model. *Preprint*, arXiv:2403.08350.

Hyung Won Chung, Le Hou, S. Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Dasha Valter, Sharan Narang, Gaurav Mishra, and 13 others. 2022. Scaling instruction-finetuned language models. *ArXiv*, abs/2210.11416.

Yawen Cui, Jian Zhao, Zitong Yu, Rizhao Cai, Xun Wang, Lei Jin, Alex C Kot, Li Liu, and Xuelong Li. 2025. Cmoa: Contrastive mixture of adapters for generalized few-shot continual learning. *IEEE Transactions on Multimedia*.

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. 2020. Orthogonal gradient descent for continual learning. In *International conference on artificial intelligence and statistics*, pages 3762–3773. PMLR.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.

Chendi Ge, Xin Wang, Zeyang Zhang, Hong Chen, Jiapei Fan, Longtao Huang, Hui Xue, and Wenwu Zhu. 2025. Dynamic mixture of curriculum lora experts for continual multimodal instruction tuning. *arXiv preprint arXiv:2506.11672*.

Haiyun Guo, ZhiYan Hou, Yu Chen, Jinghan He, Yandu Sun, Yuzhe Zhou, Shujing Guo, Kuan Zhu, and Jinqiao Wang. 2025. Mllm-cbench: A comprehensive benchmark for continual instruction tuning of multimodal llms with chain-of-thought reasoning analysis. *arXiv preprint arXiv:2508.08275*.

Jinghan He, Haiyun Guo, Ming Tang, and Jinqiao Wang. 2023. Continual instruction tuning for large multimodal models. *arXiv preprint arXiv:2311.16206*.

Jinghan He, Haiyun Guo, Kuan Zhu, Zihan Zhao, Ming Tang, and Jinqiao Wang. 2024. Seekr: Selective attention-guided knowledge retention for continual learning of large language models. *arXiv preprint arXiv:2411.06171*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Tianyu Huai, Jie Zhou, Xingjiao Wu, Qin Chen, Qingchun Bai, Ze Zhou, and Liang He. 2025. Clmoe: Enhancing multimodal large language model with dual momentum mixture-of-experts for continual visual question answering. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19608–19617.

Qihan Huang, Jie Song, Mengqi Xue, Haofei Zhang, Bingde Hu, Huiqiong Wang, Hao Jiang, Xingen Wang, and Mingli Song. 2024. Lg-cav: Train any concept activation vector with language guidance. *Advances in Neural Information Processing Systems*, 37:39522–39551.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and 1 others. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.

Zhizhong Li and Derek Hoiem. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. *Preprint*, arXiv:2310.03744.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Simone Magistri, Tomaso Trinci, Albin Soutif-Cormerais, Joost van de Weijer, and Andrew D Bagdanov. 2024. Elastic feature consolidation for cold start exemplar-free incremental learning. *arXiv preprint arXiv:2402.03917*.

Anastasia Razdaibiedina, Yuning Mao, Rui Hou, Madian Khabsa, Mike Lewis, and Amjad Almahairi. 2023. Progressive prompts: Continual learning for language models. In *The Eleventh International Conference on Learning Representations*.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. 2019a. Experience replay for continual learning. *Advances in neural information processing systems*, 32.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy P. Lillicrap, and Greg Wayne. 2019b. Experience replay for continual learning. *Preprint*, arXiv:1811.11682.

Kaushik Roy, Christian Simon, Peyman Moghadam, and Mehrtash Harandi. 2023. Subspace distillation for continual learning. *Neural Networks*, 167:65–79.

Gobinda Saha, Isha Garg, and Kaushik Roy. 2021. Gradient projection memory for continual learning. *arXiv preprint arXiv:2103.09762*.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. 2022. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5227–5237.

Quanziang Wang, Renzhen Wang, Yuexiang Li, Dong Wei, Hong Wang, Kai Ma, Yefeng Zheng, and Deyu Meng. 2024. Relational experience replay: Continual learning by adaptively tuning task-wise relationship. *IEEE Transactions on Multimedia*, 26:9683–9698.

Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuan-Jing Huang. 2023. Orthogonal subspace learning for language model continual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10658–10671.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. 2022. Learning to prompt for continual learning. *Preprint*, arXiv:2112.08654.

Julia Wenkmann and Damien Garreau. 2025. On the variability of concept activation vectors. *arXiv preprint arXiv:2509.24058*.

Shipeng Yan, Jiangwei Xie, and Xuming He. 2021. Der: Dynamically expandable representation for class incremental learning. *Preprint*, arXiv:2103.16788.

Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Ping Hu, Dong Wang, Huchuan Lu, and You He. 2024. Boosting continual learning of vision-language models via mixture-of-experts adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23219–23230.

Duzhen Zhang, Yong Ren, Zhong-Zhi Li, Yahan Yu, Jiahua Dong, Chenxing Li, Zhilong Ji, and Jinfeng Bai. 2025. Enhancing multimodal continual instruction tuning with branchlora. *arXiv preprint arXiv:2506.02041*.

Renrui Zhang, Rongyao Fang, Wei Zhang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. 2021. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*.

Zhiheng Zhang, Daojian Zeng, and Xue Bai. 2024. Improving continual few-shot relation extraction through relational knowledge distillation and prototype augmentation. In *Proceedings of the 2024 Joint*

International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), pages 8756–8767.

Ziyu Zhao, Yixiao Zhou, Zhi Zhang, Didi Zhu, Tao Shen, Zexi Li, Jinluan Yang, Xuwu Wang, Jing Su, Kun Kuang, and 1 others. 2025. Each rank could be an expert: Single-ranked mixture of experts lora for multi-task learning. *arXiv preprint arXiv:2501.15103*.

A Appendix

This appendix provides complementary details and additional results to support the main text. Appendix A.1 describes the experimental setup, including dataset details, task order construction, training hyperparameters, and evaluation protocols. Appendix A.2 reports results on MLLM-CTBench under an alternative task order, evaluating the robustness of our method to task order. Appendix A.3 presents an additional validation of PASs-RS by contrasting our structured regularization with a random regularization baseline, isolating the effect of stabilizing task-important rank directions.

A.1 Experimental Setup

We conduct all experiments on MLLM-CTBENCH, a recently proposed benchmark for continual instruction tuning of multimodal large language models. It is designed to evaluate continual adaptation under reasoning-intensive, non-saturated tasks with explicit domain shifts, making it particularly suitable for diagnosing catastrophic forgetting in modern MLLMs.

MLLM-CTBENCH consists of seven tasks spanning six diverse domains, including Math, Economics, Science, Medicine, OCR, and Arts. As summarized in Table 4, the benchmark covers both text-only QA and vision-language VQA settings, and integrates data from 16 public datasets. The tasks jointly stress symbolic reasoning, visual grounding, OCR robustness, and domain-specific knowledge. To avoid task dominance in continual learning, the training data size of each task is carefully controlled to a comparable scale.

For each task, MLLM-CTBENCH defines a canonical instruction template and a task-specific final-answer evaluation metric, as shown in Table 5. These templates are consistently applied across all methods, ensuring protocol-consistent comparison. Depending on the task characteristics, final answers are evaluated using Exact Match or ROUGE-L, while reasoning traces are used only for analysis and are not directly optimized unless specified.

Overall, the heterogeneity in modality, reasoning format, and domain semantics introduces substantial distribution shifts across tasks, posing significant challenges for continual instruction tuning. In this work, we strictly follow the official task order and training protocol of MLLM-CTBENCH, without revisiting data from previous tasks. Additional details on task ordering and evaluation settings are provided in Appendix A.1.

A.2 Results under an Alternative Task Order

Task Orders. MLLM-CTBENCH provides two official continual instruction tuning protocols with different task orders. Order-A follows: Math QA $\rightarrow$ Arts VQA $\rightarrow$ Math VQA $\rightarrow$ Economics QA $\rightarrow$ Medicine VQA $\rightarrow$ OCR VQA $\rightarrow$ Science VQA. Order-B is the reverse of Order-A.

Supplementary Results on Order-B. All main experiments in the paper follow Order-A. To further examine robustness to task ordering, we additionally evaluate all methods under Order-B. As shown in Table 6, our method achieves the best overall performance and improves the final AP by **9.46%** over the second-best method, indicating that the gains are not tied to a specific task sequence.

wo shan

A.3 Comparison with Random Rank Regularization

To further validate the effectiveness of *PASs-RS*, we compare it with a random rank regularization scheme under the same MoE-LoRA parameter budget. Specifically, instead of weighting rank directions based on their historical importance in the pathway activation subspace, the random baseline assigns stabilization weights to rank directions uniformly at random.

The results are reported in Table 7. Random rank regularization consistently leads to inferior performance, indicating that constraining arbitrary rank directions does not effectively preserve task-relevant knowledge and may even hinder adaptation to new tasks. In contrast, PASs-RS yields more reliable improvements by selectively stabilizing rank directions that are important for previous tasks, as identified by activation statistics in the pathway activation subspace. This comparison highlights the importance of using capability-aligned signals, rather than indiscriminate or random constraints, when designing rank-level stabilization mechanisms for continual instruction tuning.

Task	Data Source	Train (Text / Image)	Test (Text / Image)
Math QA	TRACE	10K / 0	0.5K / 0
Economics QA	TRACE	5K / 0	0.5K / 0
Science VQA	AI2D, ScienceQA	9K / 4K	1K / 0.5K
Math VQA	IconQA, GeoQA, CHARTX, MMMU	8.3K / 8.3K	0.9K / 0.9K
Medicine VQA	VQA-RAD, VQA-Med-2021, PMC-VQA, PathVQA	9K / 6.9K	1K / 1K
OCR VQA	ChartOCR, CROHME, ESTVQA	12K / 12.1K	1.4K / 1.4K
Arts VQA	AQUA	9K / 7K	1K / 0.9K

Table 4: Statistics of the MLLM-CTBench datasets.

Task	Instruction Prompt (Canonical)	Final-Answer Metric
Math QA	Solve the following math problem and give your reasoning, then give the answer.	Exact Match
Economics QA	Give your reasoning about the monetary policy stance, then answer with the option's letter directly.	Exact Match
Science VQA	Give the reasoning process, then answer with the option's letter directly.	Exact Match
Math VQA	Analyze the problem and give the solution; then answer with the option's letter.	Exact Match / ROUGE-L
Medicine VQA	Analyze and give the reasoning process, then answer using a single word or phrase.	ROUGE-L
OCR VQA	Give the reasoning process for text recognition, then answer using a single word or phrase.	ROUGE-L
Arts VQA	Analyze the artwork and give a reasoning process, then answer briefly.	ROUGE-L

Table 5: Canonical instruction prompts and metrics across tasks.

Method	Math QA		Arts VQA		Math VQA		Econ. QA		Med. VQA		OCR VQA		Sci. VQA		AP	BWT
	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget		
SeqFT	53.94	–	9.08	−24.99	36.57	−14.87	35.65	−37.48	20.01	−16.48	11.47	−14.41	56.84	−28.63	33.93	−17.56
SeqLoRA	**55.17**	–	11.62	−25.76	29.89	−23.93	31.97	−40.51	17.54	−19.18	9.19	−15.74	43.70	−42.63	28.44	−23.96
Replay	52.71	–	23.41	−10.50	35.92	−11.40	61.29	−11.49	27.81	−5.87	5.68	−16.74	53.63	−28.93	37.21	−12.13
EWC	53.94	–	2.27	−31.25	30.56	−22.12	37.90	−35.29	22.93	−12.09	2.27	−20.17	50.99	−31.20	28.69	−21.74
LwF	45.10	–	5.21	−28.42	36.36	−15.07	32.16	−39.01	21.77	−11.25	**19.84**	−1.95	52.40	−30.35	30.41	−18.00
MAS	53.20	–	2.63	−30.92	30.22	−21.32	37.10	−36.49	22.70	−12.50	2.84	−19.60	51.08	−31.48	28.54	−21.76
O-LoRA	54.84	–	29.16	−5.00	35.48	−16.63	34.53	−37.45	23.63	−10.16	13.49	−8.66	56.84	−25.35	35.42	−14.75
HiDe-llava	42.12	–	28.33	−2.74	43.33	−1.82	47.78	−14.82	24.87	−7.70	15.17	−8.62	46.65	−36.57	28.80	−16.99
DDAS	40.89	–	13.48	−0.48	37.06	+10.38	54.64	−13.71	21.65	−2.82	1.07	−0.78	50.05	−13.95	31.26	−3.05
MoE-lora (Top-k)	51.56	–	2.47	−30.61	29.66	−19.08	34.27	−32.92	16.08	−16.39	3.47	−18.32	42.86	−35.62	25.77	−21.85
MoE-lora (Softmax)	53.45	–	2.49	−31.05	31.24	−20.19	38.71	−35.08	22.94	−11.81	2.27	−20.38	50.42	−32.14	28.79	−21.52
Ours	53.69	–	**60.25**	+26.99	**46.52**	−4.34	**66.94**	−1.61	**28.11**	−6.43	6.25	−16.04	**64.94**	−17.72	**46.67**	−2.74

Table 6: Comparison with traditional methods and MoE-LoRA-based methods on MLLM-CTBench. Best and second-best results for Acc and AP are marked in **bold** and underline.

Reg. scheme	Math QA		Arts VQA		Math VQA		Econ. QA		Med. VQA		OCR VQA		Sci. VQA		AP	BWT
	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget	Acc	Forget		
Random	49.01	−6.16	40.79	+7.94	41.50	−9.47	57.10	−13.67	27.50	−9.03	20.19	−3.94	79.76	–	45.12	−4.90
Ours	49.52	−5.90	43.22	+10.43	44.70	−6.50	66.13	−6.05	29.99	−5.38	21.95	−1.63	83.73	–	48.46	−2.15

Table 7: Ablation on the regularization scheme in sequential multimodal continual learning. We compare random regularization with our projection-aware regularization while keeping all other components fixed.