Dual-Pronged LoRA: Achieving Near-Zero Forgetting and High-Performance Adaptation for MLLMs

Jingqi Ye[1,2] **Haonan He**[1,2] **Minglei Li**[1,3] **Fujun Han**[1,4] **Tao Chen**[3,6] **Peng Ye**[1,5]*
[1]Shanghai Artificial Intelligence Laboratory [2]University of Science and Technology of China
[3]Fudan University [4]The Chinese University of Hong Kong, Shenzhen
[5]The Chinese University of Hong Kong [6]Shanghai Innovation Institute
amzingye@mail.ustc.edu.cn

Abstract

Adapting Multimodal Large Language Models (MLLMs) to specialized and diverse downstream applications necessitates fine-tuning, typically through parameter-efficient methods such as *Low-Rank Adaptation* (LoRA). However, fine-tuning MLLMs severely degrades pre-trained upstream capabilities, a phenomenon known as *catastrophic forgetting*. Existing methods primarily rely on parameter pruning to strike a balance between the downstream adaptation and upstream knowledge retention. Empirical analysis indicates that these methods are subject to *parameter space entanglement*, wherein the imposition of downstream adaptation and upstream retention within a unified parameter space results in suboptimal outcomes for both objectives. To address this, we introduce Dual-Pronged LoRA, a novel fine-tuning framework for MLLMs featuring two components. (1) *Task-Decoupled Contextual Routing* (TDCR) makes the first attempt to explicitly decouple tasks by modeling a downstream contextual support region. It adaptively activates the LoRA branch exclusively for downstream-relevant inputs and bypasses it otherwise, thereby preventing downstream adaptation from interfering with pre-trained knowledge. (2) *Downstream-Performance-Oriented Rank Sculpting* (DPO-RS) maximizes downstream adaptation performance by adaptively sculpting ranks via gradient energy. It effectively strengthens critical layers while compressing redundant ones and is theoretically grounded in the minimization of information loss. Extensive experimental results indicate that Dual-Pronged LoRA achieves near-zero forgetting while attaining new state-of-the-art overall performance.

1 Introduction

Multimodal large language models (MLLMs) [31, 2, 3] have advanced rapidly and emerged as a central research topic in artificial intelligence. Despite their powerful and general capabilities [37, 25, 62, 11, 13, 15], real-world scenarios encompass specialized and diverse applications, which necessitate downstream fine-tuning to enhance performance on targeted tasks [32, 63, 12, 14]. However, downstream fine-tuning inevitably degrades the model's pre-established upstream capabilities due to the distribution shift between upstream pre-training and downstream task data, a well-documented phenomenon known as catastrophic forgetting [78, 22, 34, 24, 1]. In MLLMs, this issue is significantly exacerbated by inherent structural complexities, cross-modal data discrepancies, and cross-task interference [26, 54, 34, 73]. Consequently, simultaneously mitigating catastrophic forgetting while maximizing the downstream fine-tuning performance when adapting MLLMs remains a long-standing, crucial, yet challenging problem.

*Corresponding author: 20110720039@fudan.edu.cn

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

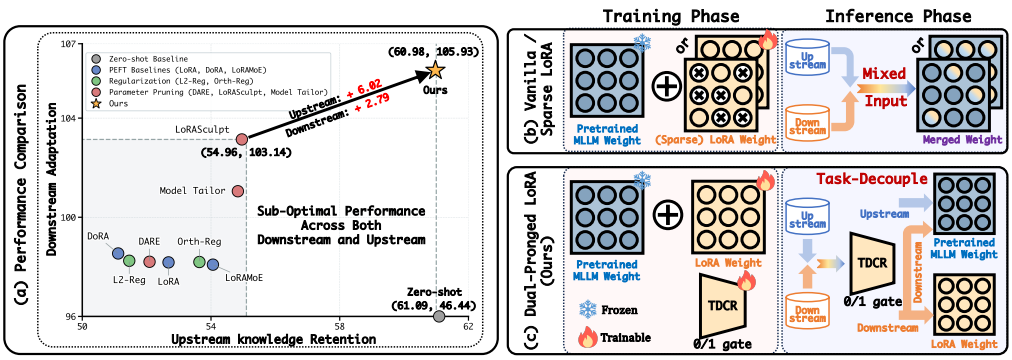


Figure 1: Entangled vs. Decoupled Adaptation. (a) Performance comparison between Dual-Pronged LoRA and baselines. We present the average performance on IconQA and COCO-Caption under a parameter budget matched to LoRA rank $r = 32$ to ensure fair comparison. (b) Existing LoRA variants entangle downstream adaptation and upstream knowledge retention within a shared parameter space. (c) In contrast, Dual-Pronged LoRA introduces explicit task decoupling via TDCR, effectively isolating parameter spaces.

Existing studies on this problem can be broadly divided into two categories: (1) *Mitigation forgetting within full fine-tuning*. Model Tailor [78] introduces a post-pruning strategy that identifies and compensates for a small set of critical parameters to alleviate catastrophic forgetting. SPIDER [22] models the importance of parameter pairs to both the upstream knowledge and downstream performance and performs selective updates through masking. Model-Dowser [24] further proposes a data-free method for parameter importance estimation. However, full fine-tuning often incurs substantial computational cost, especially as model scales increase. (2) *Mitigation forgetting within Low-Rank Adaptation (LoRA)*. Model Tailor [78] also demonstrates that removing harmful redundant parameters in LoRA helps alleviate forgetting. LoRASculpt [34] further proposes dynamic parameter pruning during training. We follow this emerging direction due to its balanced performance and efficiency. However, as shown in Fig. 1(a), these LoRA methods still struggle to eliminate catastrophic forgetting, and their downstream performance often remains sub-optimal.

We argue that the bottleneck of existing methods lies in *parameter space entanglement*, where forcing downstream adaptation and upstream knowledge retention within a unified parameter space inevitably creates a structural conflict. As illustrated in Fig. 1(b), optimizing LoRA for downstream performance ineluctably overwrites upstream knowledge. Although applying pruning can preserve general knowledge to some extent, it also prevents LoRA from fully fitting the downstream distribution. Trapped in this trade-off, models perform sub-optimally in both aspects. Based on the above analysis, we identify *explicit task decoupling* as the fundamental solution (Fig. 1(c)). By dynamically activating the LoRA module exclusively for downstream inputs and bypassing it otherwise, we achieve strict *parameter space decoupling*, which completely circumvents catastrophic forgetting and liberates LoRA to focus on maximizing downstream performance. Realizing this target is non-trivial and has two primary challenges: (1) How to route tasks when only downstream data is accessible? and (2) How to configure the liberated LoRA parameters to maximize adaptation of MLLMs.

In this paper, we propose Dual-Pronged LoRA, a novel fine-tuning framework for MLLMs, which preserves upstream knowledge via *Task-Decoupled Contextual Routing* (TDCR) while maximizing downstream adaptation through *Downstream-Performance-Oriented Rank Sculpting* (DPO-RS). Inspired by *One-Class Classification* (OCC) [52, 48], TDCR learns a tightly bounded *contextual support region* using solely downstream data. During inference, inputs within this region are identified as relevant to downstream tasks, while those deviated are recognized as general upstream tasks. By adaptively activating the LoRA branch only for the former and bypassing it for the latter, TDCR effectively shields pre-trained upstream knowledge from catastrophic interference and frees LoRA to focus exclusively on maximizing downstream performance. Leveraging this structural advantage, we further introduce DPO-RS, a gradient-driven rank sculpting strategy. Motivated by the structural complexity of MLLMs and the theoretical derivation of minimum information loss, DPO-RS dynamically sculpts layer-wise ranks based on gradient energy. It strengthens critical layers while pruning redundant ones [16, 65, 70], thus maximizing adaptation efficacy within comparable parameters.

Extensive experiments on various VQA and captioning tasks demonstrate that Dual-Pronged LoRA achieves near-zero catastrophic forgetting while attaining state-of-the-art overall performance. At parameter budgets matched to LoRA ranks 16 and 32, our approach improves the strongest baseline *Avg* by +3.16/+0.76 on IconQA and +5.64/+5.28 on COCO-Caption, respectively. At the rank-16-equivalent budget, it also maintains upstream scores of 60.86 and 61.09, respectively (vs. the zero-shot baseline of 61.09), whereas the best baselines reach only 55.24 and 55.44. Besides, we demonstrate that explicit task decoupling completely liberates LoRA from the retention-adaptation trade-off. Furthermore, we observe that TDCR may benefit from a *semantic buffering* effect across related tasks and exhibits exceptional generalization even under highly specialized domain shifts. Our main contributions are summarized as follows:

- **Identifying Parameter Entanglement as the Core Bottleneck.** We reveal that existing methods for mitigating catastrophic forgetting in MLLMs suffer from a persistent trade-off caused by parameter entanglement. We propose a novel perspective of ***explicit task decoupling*** as a fundamental solution to resolve this structural conflict.
- **Novel MLLM Fine-Tuning Framework Dual-Pronged LoRA.** It features two modules: **TDCR** decouples tasks explicitly in MLLMs by routing inputs to isolated parameter spaces, preventing upstream knowledge from downstream interference. **DPO-RS** sculpts LoRA ranks guided by gradient energy, it strengthens critical layers and compresses redundant ones under a constrained parameter budget, maximizing downstream adaptation.
- **Near-Zero Forgetting and High-Performance Adaptation.** Extensive experiments demonstrate that Dual-Pronged LoRA outperforms the highest *Avg* baseline in downstream adaptation (+0.38 IconQA, +4.87 COCO-Caption) while achieving near-zero forgetting, maintaining scores of 60.86 and 61.09, respectively (vs. 61.09 zero-shot) compared to $\leq$ 59.43 for existing methods. Our in-depth analyses also validate its generalization and robustness.

2 Related Work

• **Catastrophic Forgetting** is a fundamental challenge in deep learning, referring to the previously acquired knowledge being lost when adapting to new tasks [22, 20, 21, 58, 78]. Traditional mitigation strategies, such as pruning-based [42, 41, 60, 61], regularization-based [28, 39, 53], and architecture-based methods [47, 59, 71, 76], are predominantly designed for smaller networks and rely heavily on FFT. As MLLMs scale, PEFT methods, especially LoRA [18], have become the dominant choice for downstream adaptation. Nevertheless, recent studies reveal that LoRA still suffers from severe catastrophic forgetting [78, 34, 1]. To address this, Model Tailor [78] alleviates forgetting by eliminating LoRA redundancy via post-pruning. While LoRASculpt [34] further introduces training-time LoRA sparsification and conflict-aware regularization, achieving SOTA performance. However, these works remain confined to the traditional paradigm of *parameter space entanglement*, forcing an inevitable trade-off between downstream adaptation and upstream knowledge retention.

• **One-Class Classification (OCC)** is a classical approach for anomaly detection and out-of-distribution (OOD) identification, designed to model the support of a target distribution using only positive examples. Traditional methods such as One-Class SVM [52] and Support Vector Data Description (SVDD) [56] construct hyperplanes or hyperspheres around the target class, while deep learning extensions such as Deep SVDD [48] employ neural networks to learn compact latent representations for high-dimensional data. These approaches have been widely applied in computer vision [51, 48, 50, 30], language modeling [49, 27], and multimodal embeddings [17, 30]. Inspired by these successes, our method is the first attempt to apply OCC for *explicit task-decoupling* in MLLMs. This enables dynamic activation of LoRA for downstream-related tasks while bypassing LoRA for irrelevant inputs, effectively mitigating or even eliminating catastrophic forgetting.

3 Method

3.1 Overview of LoRA

LoRA [18] is built on the hypothesis that fine-tuning updates lie in low-rank subspaces. Specifically, given a pretrained weight matrix $W_0 \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, LoRA approximates the weight update ΔW using two low-rank matrices $A \in \mathbb{R}^{r \times d_{\text{in}}}$ and $B \in \mathbb{R}^{d_{\text{out}} \times r}$, where $r \ll \min(d_{\text{out}}, d_{\text{in}})$. Accordingly, the

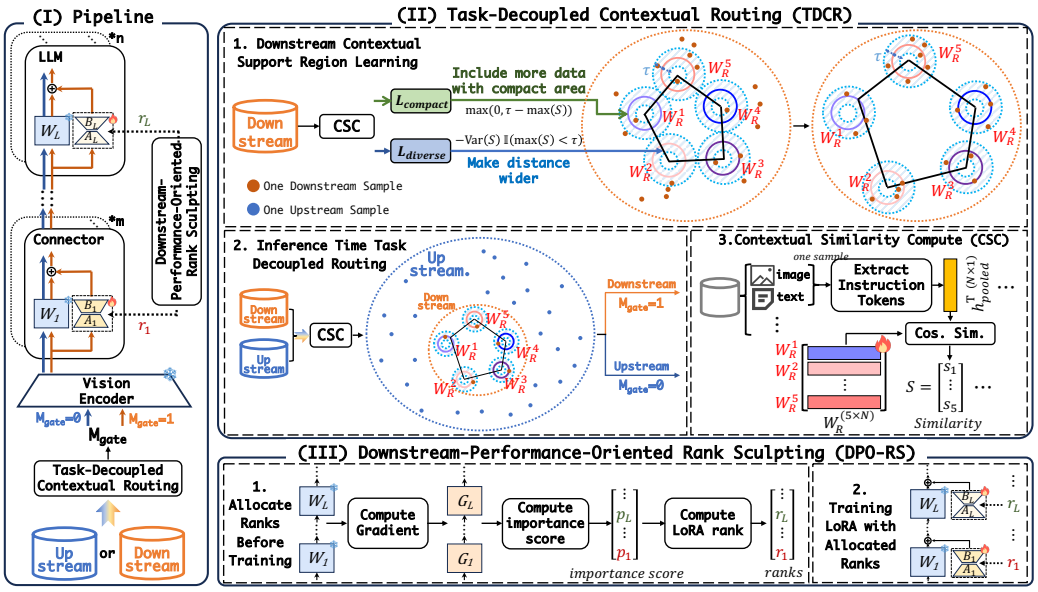


Figure 2: Overview of the Dual-Pronged LoRA framework. (I) The overall pipeline combines explicit task routing with layer-wise rank-sculpted LoRA. (II) TDCR learns a context matrix W_R to learn the downstream *contextual support region*, enabling inference-time routing that activates LoRA for downstream inputs and bypasses it otherwise. (III) DPO-RS allocates layer-wise LoRA ranks (r_l) from gradient importance scores under a fixed parameter budget.

fine-tuned LoRA weight with scaling factor α is expressed as:

$$W = W_0 + \Delta W = W_0 + \frac{\alpha}{r} BA, \tag{1}$$

Eq. (1) reveals a key property of LoRA: it is inherently *plug-and-play*, as weight updates are injected as an additive LoRA module without directly modifying the pretrained weights. This property inspires a novel perspective on mitigating catastrophic forgetting. Instead of always activating LoRA during inference, we selectively enable it based on input relevance. For target downstream tasks, LoRA is activated to inject specialized knowledge; otherwise, it is bypassed, relying solely on the MLLM backbone's generic knowledge. This is formulated as:

$$y = \left(W_0 + \frac{\alpha}{r} M_{\text{gate}}(x) BA\right) x, \tag{2}$$

where $M_{\text{gate}}(x) \in \{0, 1\}$ is a gating variable. When x is identified as downstream-related, $M_{\text{gate}}(x) = 1$; otherwise, $M_{\text{gate}}(x) = 0$.

Building on this insight, we propose Dual-Pronged LoRA (Fig. 2), which instantiates *explicit task-decoupling* through two synergistic components: **TDCR** learns a downstream contextual support region to route inputs into isolated parameter spaces; **DPO-RS** then sculpts layer-wise ranks via gradient energy, maximizing downstream adaptation performance.

3.2 Task-Decoupled Contextual Routing

We introduce *Task-Decoupled Contextual Routing* (TDCR), which reframes routing as an OCC problem. Specifically, TDCR operates through a two-step mechanism (Fig. 2-II): it first explicitly learns a *contextual support region* to characterize the downstream distribution during training, and subsequently leverages this region for *task-decoupling* during inference.

3.2.1 Downstream Contextual Support Region Learning

Given an input sequence, we first extract an instruction-aware global representation via masked embedding pooling:

$$h_{\text{pooled}} = \frac{\sum_{i=1}^{L} h_i \, m_i^{\text{ins}}}{\sum_{i=1}^{L} m_i^{\text{ins}} + \epsilon} \tag{3}$$

where $h_i \in \mathbb{R}^d$ denotes the input embedding of the i-th token, $m_i^{\text{ins}} \in \{0, 1\}$ is a binary mask constructed from the prompt template, and ϵ is a small constant for numerical stability. Specifically, the mask selects only instruction tokens between the image placeholder and the assistant tag. It is constructed identically during training and inference.

To characterize the downstream contextual support region, we measure the alignment of h_{pooled} against a learnable context matrix $W_R \in \mathbb{R}^{d \times K}$ [49], whose K columns capture diverse task semantics. Let $\bar{h}_{\text{pooled}}$ and $\bar{W}_R$ denote their ℓ_2-normalized counterparts. The contextual similarity vector $S \in \mathbb{R}^K$ is then computed as:

$$S = \bar{h}_{\text{pooled}}^{\top} \bar{W}_R \in \mathbb{R}^K \tag{4}$$

By mapping these representations via cosine similarity, the resulting scores are strictly bounded to $[-1, 1]$. This ensures the measurement focuses exclusively on directional semantic alignment rather than feature magnitude, thereby inherently stabilizing gradient updates in high-dimensional spaces.

Crucially, we optimize the context matrix W_R to ensure its support region tightly bounds $\bar{h}_{\text{pooled}}$. To achieve this, we train the context matrix W_R with a joint objective. First, we introduce a *contextual compactness loss*:

$$\mathcal{L}_{\text{compact}} = \max(0, \tau - \max(S)) \tag{5}$$

where τ is a predefined compactness threshold, this loss penalizes the model if the highest similarity to the pooled representation falls below τ, thereby forming compact support regions.

Furthermore, to avoid representation collapse into a single dominant direction [4, 67, 68] and encourage the context matrix to capture complementary semantic aspects of downstream tasks, we introduce a *context diversity regularization loss*:

$$\mathcal{L}_{\text{diverse}} = -\text{Var}(S)\, \mathbb{I}(\max(S) < \tau) \tag{6}$$

Here $\text{Var}(S)$ computes the variance across the K dimensions of S.

Ultimately, the context matrix W_R is trained alongside the standard causal language modeling loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda_1 \mathcal{L}_{\text{compact}} + \lambda_2 \mathcal{L}_{\text{diverse}}, \tag{7}$$

where $\mathcal{L}_{\text{CE}}$ denotes the language modeling loss, and λ_1 and λ_2 govern the strength of the compactness and diversity regularizers, respectively.

3.2.2 Inference-Time Task-Decoupled Routing

During inference, the routing decision is based on a soft boundary defined by the alignment between the incoming pooled representation and the learned context matrix W_R. Specifically, we compute the contextual similarity vector S and derive a global gating signal to dictate the routing behavior:

$$M_{\text{gate}} = \mathbb{I}(\max(S) > \tau - \delta) \tag{8}$$

where δ serves as a soft tolerance margin. Because the context vectors are optimized to form a compact support region rather than a rigid geometric bound, incorporating δ provides the necessary representational flexibility. This prevents false rejections of valid downstream inputs near the distribution edge, ensuring they are robustly routed to the LoRA branch.

This gating signal dynamically controls the activation of the LoRA branch during the forward pass:

$$y = W_0 x + \frac{\alpha}{r} M_{\text{gate}}(h_{\text{pooled}}) BAx \tag{9}$$

When $M_{\text{gate}} = 1$, the input representation falls within the *contextual support region* defined by the context matrix W_R, triggering the activation of the LoRA branch to inject downstream knowledge. Conversely, for inputs whose representations deviate from this downstream support region (i.e., general upstream tasks), M_{gate} evaluates to 0, and the LoRA module is entirely bypassed. This task-decoupling mechanism ensures that the specialized LoRA parameter is invoked exclusively for downstream-relevant tasks, effectively protecting the general upstream knowledge within MLLMs.

3.3 Downstream-Performance-Oriented Rank Sculpting

The heterogeneous adaptation demands across MLLM sub-modules make a uniform LoRA rank inherently suboptimal [5], motivating a principled layer-wise allocation criterion. To this end, Appendix A shows that constraining the effective update of the l-th layer to rank r_l incurs an irreducible optimization information loss, which is approximately characterized by the truncated tail gradient energy:

$$\mathcal{E}_{\text{loss}}^{(l)} \approx \eta \sum_{i=2r_l+1}^{d^{(l)}} \left(\sigma_i^{(l)}\right)^2, \tag{10}$$

where $\sigma_i^{(l)}$ denotes the singular values of the layer gradient G_l, η is the learning rate, and $d^{(l)} = \min\left(d_{\text{in}}^{(l)}, d_{\text{out}}^{(l)}\right)$. Under a global parameter budget $\mathcal{B}$, minimizing the total optimization information loss leads to the constrained objective:

$$\begin{aligned} &\min_{\{r_l\}} \sum_{l=1}^{L} \mathcal{E}_{\text{loss}}^{(l)} = \eta \sum_{l=1}^{L} \sum_{i=2r_l+1}^{d^{(l)}} \left(\sigma_i^{(l)}\right)^2 \\ &\text{s.t.} \quad \sum_{l=1}^{L} r_l \left(d_{\text{in}}^{(l)} + d_{\text{out}}^{(l)}\right) \le \mathcal{B}, \quad r_l \in \mathbb{Z}_+, \quad r_{\min} \le r_l \le r_{\max}. \end{aligned} \tag{11}$$

To obtain a lightweight yet principled approximation, DPO-RS estimates layer-wise adaptation demand from the *initial* full gradient before fine-tuning. Specifically, for each pretrained weight matrix W_l, we compute its gradient G_l on a sampled downstream mini-batch, and use the normalized gradient Frobenius norm as the layer-wise importance score:

$$p_l = \frac{\|G_l\|_F}{\sum_k \|G_k\|_F}. \tag{12}$$

This design is justified in Appendix A: the initial gradient provides a clean estimate of each layer's intrinsic adaptation demand before the optimization trajectory is biased by the low-rank LoRA subspace, while the resulting truncation loss can be approximated by a lightweight surrogate based on total gradient energy.

Since the parameter of a rank-r_l LoRA is proportional to $r_l\left(d_{\text{in}}^{(l)} + d_{\text{out}}^{(l)}\right)$, the rank is assigned as:

$$r_l = \max\left(r_{\min}, \min\left(r_{\max}, \left\lfloor \frac{\mathcal{B} \cdot p_l}{\sqrt{d_{\text{in}}^{(l)} + d_{\text{out}}^{(l)}}} \right\rfloor\right)\right), \tag{13}$$

where $\mathcal{B}$ is the global parameter budget, and $r_{\min}$ and $r_{\max}$ are lower and upper bounds used to avoid overly aggressive compression or unstable over-allocation. By allocating higher ranks to more sensitive layers and compressing less sensitive ones, DPO-RS preserves capacity where low-rank truncation would be most harmful while reducing parameter redundancy elsewhere. The overall training pipeline of Dual-Pronged LoRA is summarized in Algorithm 1.

4 Experiments

Following established experimental protocols from prior work [34, 78, 22], we compare Dual-Pronged LoRA against several baselines to comprehensively evaluate its effectiveness in mitigating catastrophic forgetting and its downstream adaptation performance.

All experiments are built upon the LLaVA-1.5 [36] architecture. For downstream adaptation, we fine-tune and evaluate the model on two representative visual tasks: Visual Question Answering (VQA) using IconQA [40] and image captioning using COCO-Caption [35]. To assess the mitigation of catastrophic forgetting, the fine-tuned models are subsequently evaluated on four datasets originally included in the LLaVA-1.5 training corpus [77, 34]: OKVQA [43], OCRVQA [44], GQA [23], and TextVQA [55]. These datasets serve as proxies for the model's pre-trained upstream knowledge. The training process spans 3 epochs with a batch size of 16. We fine-tune the LLM backbone with a learning rate of 2e-4, while fine-tuning the projector at a learning rate of 2e-5. Due to space constraints, please refer to Appendix D for a more detailed experimental setup.

Table 1: Compare our Dual-Pronged LoRA with SOTA baselines on IconQA and COCO-Caption. *Source* represents the model's capacity to retain upstream knowledge, while *Target* indicates the performance after fine-tuning on the corresponding downstream task.

Methods	IconQA							COCO-Caption						
	OKVQA	OCRVQA	GQA	TextVQA	*Source*	*Target*	*Avg*	OKVQA	OCRVQA	GQA	TextVQA	*Source*	*Target*	*Avg*
Zero-shot	57.99	66.20	61.93	58.23	61.09	23.18	42.13	57.99	66.20	61.93	58.23	61.09	40.40	50.74
Budget matched to LoRA Rank=16														
LoRA [18]	51.10	54.90	55.56	49.48	52.76	84.14	68.45	47.34	60.55	56.91	45.61	52.60	112.22	82.41
DoRA [38]	49.94	54.40	55.39	47.26	51.75	84.48	68.11	45.38	59.75	55.40	41.38	50.48	112.60	81.54
AdaLoRA [74]	53.02	54.05	58.20	51.57	54.21	84.56	69.39	25.57	54.00	38.92	27.13	36.41	121.53	78.97
GoRA [66]	42.54	58.00	50.63	22.11	43.32	83.09	63.21	46.25	63.05	57.02	42.15	52.12	117.56	84.84
L2-Reg [28]	49.96	51.40	55.15	47.47	51.00	84.66	67.83	46.05	61.50	56.60	43.54	51.92	111.83	81.88
Orth-Reg [19]	53.12	56.10	57.43	51.00	54.41	84.52	69.47	48.11	59.85	56.83	46.69	52.87	111.87	82.37
Model Tailor [78]	52.68	56.55	56.26	51.41	54.23	83.26	68.74	51.83	59.75	58.73	51.43	55.44	118.84	87.14
DARE [72]	51.10	54.30	55.55	49.20	52.54	84.20	68.37	46.64	59.45	56.01	44.46	51.64	112.21	81.93
LoRAMoE [7]	53.58	55.15	57.66	51.72	54.53	84.90	69.71	48.81	59.55	57.46	48.54	53.59	111.77	82.68
LoRASculpt [34]	54.07	57.00	57.66	52.24	55.24	85.02	70.13	50.86	59.10	57.64	51.10	54.68	121.26	87.97
Dual-Pronged LoRA (Ours)	57.92	66.15	61.75	57.60	**60.86**	**85.72**	**73.29** ↑3.16	57.97	66.25	61.93	58.21	**61.09**	**126.13**	**93.61** ↑5.64
Budget matched to LoRA Rank=32														
LoRA [18]	45.06	48.40	49.22	36.47	44.79	82.52	63.65	44.92	59.90	54.88	39.75	49.86	110.27	80.07
DoRA [38]	46.35	48.50	40.80	33.01	42.17	84.82	63.49	44.10	60.05	54.69	40.17	49.75	109.25	79.50
AdaLoRA [74]	53.19	51.60	57.68	51.55	53.51	**85.89**	69.70	28.71	55.20	43.32	31.26	39.62	121.32	80.47
GoRA [66]	56.53	63.45	61.03	56.69	59.43	85.47	72.45	25.59	24.50	26.36	18.95	23.85	92.26	58.06
L2-Reg [28]	30.16	26.65	16.62	17.16	22.65	82.36	52.50	43.60	60.00	54.09	36.53	48.56	109.39	78.97
Orth-Reg [19]	41.51	40.45	53.58	30.92	41.62	83.14	62.38	43.77	60.70	53.74	38.61	49.21	110.44	79.82
Model Tailor [78]	50.07	53.85	53.04	46.92	50.97	81.76	66.37	50.99	60.60	58.54	49.65	54.95	117.64	86.29
DARE [72]	44.39	46.85	48.75	34.84	43.71	82.28	62.99	42.98	58.65	53.92	38.61	48.54	108.57	78.56
LoRAMoE [7]	51.51	55.05	56.65	49.83	53.26	84.77	69.02	48.64	59.75	57.05	47.59	53.26	112.86	83.06
LoRASculpt [34]	53.52	59.50	57.63	53.76	56.10	85.26	70.68	49.99	58.65	57.63	50.73	54.25	120.35	87.30
Dual-Pronged LoRA (Ours)	57.99	66.25	61.93	56.12	**60.57**	85.85	**73.21** ↑0.76	57.98	66.25	61.93	58.21	**61.09**	**124.07**	**92.58** ↑5.28

4.1 Comparison to State-of-the-Art Baselines

The results are summarized in Table 1. We draw the following conclusions:

Superior mitigation of catastrophic forgetting. Under rank=16, vanilla LoRA drops from 61.09 to 52.76 on IconQA, and even the strongest baseline LoRASculpt only recovers to 55.24. In contrast, Dual-Pronged LoRA reaches 60.86 on IconQA and exactly matches the zero-shot 61.09 on COCO-Caption. Such near-lossless retention directly validates TDCR's *task decoupling*: by bypassing the LoRA branch for upstream inputs, the frozen backbone is shielded from interference.

Breaking the retention–adaptation trade-off. Liberated from the burden of upstream retention, DPO-RS fully devotes its capacity to downstream adaptation, achieving Target scores of 85.72/85.85 on IconQA and 126.13/124.07 on COCO-Caption at the LoRA-equivalent rank-16/32 budgets, respectively. Although AdaLoRA marginally exceeds our IconQA Target at rank 32 (85.89 vs. 85.85), Dual-Pronged LoRA attains the best *Avg* across both tasks and budgets, surpassing the strongest baselines by +3.16/+0.76 on IconQA and +5.64/+5.28 on COCO-Caption (rank 16/32), respectively.

Robustness to rank scaling. Increasing the rank from 16 to 32 exacerbates forgetting in baselines (e.g., LoRA: $52.76 \rightarrow 44.79$; L2-Reg collapses to 22.65), as larger adapters induce stronger interference. In contrast, Dual-Pronged LoRA remains stable across both ranks, further evidencing that *explicit parameter decoupling* is robust to knowledge retention. Notably, our best results are already reached at rank=16, highlighting DPO-RS's efficiency in downstream learning.

4.2 Ablation Study

4.2.1 Complementary Synergy of TDCR and DPO-RS

We ablate the two core components of Dual-Pronged LoRA in Table 2. DPO-RS substantially boosts downstream adaptation, improving the Target score from 112.17 to 123.74 on COCO-Caption and from 83.90 to 85.97 on IconQA over vanilla LoRA. This confirms that sculpting ranks via gradient energy strengthens critical layers, while compressing redundant ones further provides implicit regularization that partially alleviates forgetting. However, compared to the zero-shot base-

Table 2: Ablation study on the core components of Dual-Pronged LoRA: TDCR and DPO-RS.

TDCR	DPO-RS	IconQA			COCO-Caption		
		Source	*Target*	*Avg*	*Source*	*Target*	*Avg*
Zero-shot		61.09	23.18	42.13	61.09	40.40	50.74
LoRA		54.32	83.90	69.11	52.29	112.17	82.23
✓		60.48	83.90	72.19	61.10	112.17	86.63
	✓	54.58	**85.97**	70.24	53.45	123.74	88.59
✓	✓	**60.57**	85.85	**73.21**	**61.09**	**124.07**	**92.58**

line, DPO-RS alone cannot overcome *parameter space entanglement*, leaving upstream knowledge compromised. In contrast, introducing TDCR alone improves the Source score from 54.32 to 60.48 on IconQA and from 52.29 to 61.10 on COCO-Caption, without sacrificing any downstream Target performance. Combining both components achieves near-zero forgetting while further enhancing downstream adaptation, validating the complementary roles of the two components: TDCR enforces *explicit task decoupling* to shield upstream knowledge, while DPO-RS is dedicated to maximizing downstream adaptation performance.

4.2.2 Robustness of Task-Decoupled Contextual Routing

We study the sensitivity of the hyperparameters λ_1 and λ_2 in Eq. (7) over a wide range on IconQA with LoRA rank $= 32$. Results are reported in Fig. 3, from which we draw three observations.

Robustness to Hyperparameters. Dual-Pronged LoRA is remarkably stable across a broad search space of λ_1 and λ_2: both upstream retention and downstream performance remain consistent. Even in the worst case, the upstream score stays at 59.61, far exceeding vanilla LoRA (44.79). This confirms that the gains of Dual-Pronged LoRA stem from its intrinsic mechanism rather than exhaustive hyperparameter tuning. In terms of dynamics, λ_1 should be sufficiently large to form a precise *contextual support region*, whereas λ_2 is most effective at moderate values—overly large weights yield negative impacts.

Empirical Validation of Explicit Task Decoupling. The downstream performance remains stable across a wide range of λ_1 and λ_2, indicating that the capacity of the activated LoRA branch is independent of the routing objective. This empirically corroborates our core claim: *parameter space decoupling* liberates LoRA from upstream knowledge retention. As shown in Fig. 3(b), the slight drop when $\lambda_1 < 10^{-3}$ stems from a few misrouted samples under an underfit routing boundary.

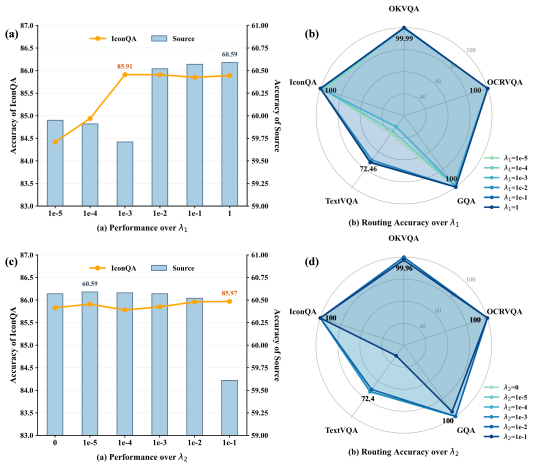


Figure 3: Ablation of coefficient λ_1 and λ_2

Robustness to Routing Errors via Semantic Buffering. On TextVQA, despite a relatively lower routing accuracy, upstream retention degrades only marginally. This may be partly attributed to the semantic proximity between TextVQA and IconQA (see Sec. 5.2): when a sample is misrouted to a semantically related path, the induced interference may be limited rather than catastrophic. This suggests a possible *semantic buffering* effect across related tasks, which may contribute to the observed robustness.

4.2.3 Robustness to the Rank Sculpting Range.

We evaluate the impact of the sculpting range $(r_{\min}, r_{\max})$ in Eq. (13) on IconQA, under a parameter budget comparable to LoRA at rank $r = 32$. As reported in Table 3, Dual-Pronged LoRA is highly robust to this range: the *Avg* score is tightly concentrated within 73.13–73.30, with only marginal fluctuations in upstream retention and downstream performance. More importantly, the results substantiate the efficacy of DPO-RS in maximizing downstream adaptation. While vanilla LoRA at $r = 32$ reaches a Target of only 82.52, all variants of Dual-Pronged LoRA consistently push it above 85.70.

Table 3: Ablation of different rank sculpting ranges under comparable parameters to LoRA with rank $r = 32$.

Methods	Fine-Tune on IconQA Dataset						
	OKVQA	OCRVQA	GQA	TextVQA	*Source*	*Target*	*Avg*
Zero-shot	57.99	66.20	61.93	58.23	61.09	23.18	42.13
LoRA	45.06	48.40	49.22	36.47	44.79	82.52	63.65
Ours (8-128)	57.99	66.25	61.93	56.12	60.57	85.85	73.21
Ours (16-128)	57.88	66.25	61.79	56.21	60.53	85.72	73.13
Ours (8-256)	57.99	66.25	61.93	56.03	**60.55**	**86.04**	**73.30**
Ours (16-256)	57.89	66.25	61.81	55.66	60.40	85.92	73.16

This confirms that gradient-energy-guided rank sculpting yields consistently stronger downstream adaptation than uniform rank allocation, in line with the structural heterogeneity of MLLMs.

5 Discussion

5.1 Generalization Across Diverse Distributions

To further validate the generalization of Dual-Pronged LoRA, we conduct additional experiments on ArxivQA [33] and Flickr30k [45], with full settings in Appendix D. As shown in Table 4, Dual-Pronged LoRA consistently achieves strong downstream adaptation together with near-zero forgetting, confirming its robustness under heterogeneous domain shifts.

The advantage is most pronounced on ArxivQA, a highly specialized academic domain. Existing methods collapse catastrophically: the *Source* scores of LoRA and LoRASculpt drop sharply to 38.05 and 36.10, respectively. In contrast, Dual-Pronged LoRA sustains a *Source* score of 60.78 (near-zero forgetting) while simultaneously attaining superior downstream performance. This result confirms that TDCR effectively isolates even highly complex, domain-specialized tasks.

5.2 Case Study

To illustrate the possible *semantic buffering* effect discussed in Sec. 4.2.2, we conduct a case study on TextVQA, as illustrated in Fig. 4. Vanilla LoRA exhibits pronounced catastrophic forgetting, with degraded instruction-following and hallucinated responses, reaffirming the necessity of mitigating forgetting in LoRA-based adaptation of MLLMs.

Table 4: Generalization of Dual-Pronged LoRA under heterogeneous domain shifts on ArxivQA and Flickr30k.

Methods	Fine-Tune on ArxivQA Dataset						
	OKVQA	OCRVQA	GQA	TextVQA	*Source*	*Target*	*Avg*
Zero-shot	57.99	66.20	61.93	58.23	61.09	53.80	57.45
LoRA	41.61	27.20	45.41	37.97	38.05	89.77	63.91
LoRASculpt	39.29	24.45	42.73	37.92	36.10	91.43	63.76
Ours	58.02	66.25	61.68	57.16	**60.78**	**92.17**	**76.47**
Methods	Fine-Tune on Flicker30k Dataset						
	OKVQA	OCRVQA	GQA	TextVQA	*Source*	*Target*	*Avg*
Zero-shot	57.99	66.20	61.93	58.23	61.09	58.44	59.77
LoRA	49.29	59.40	57.00	51.24	54.23	80.78	67.51
LoRASculpt	51.91	60.30	58.19	53.38	55.95	85.42	70.68
Ours	57.99	66.25	61.93	58.21	**61.09**	**90.13**	**75.61**

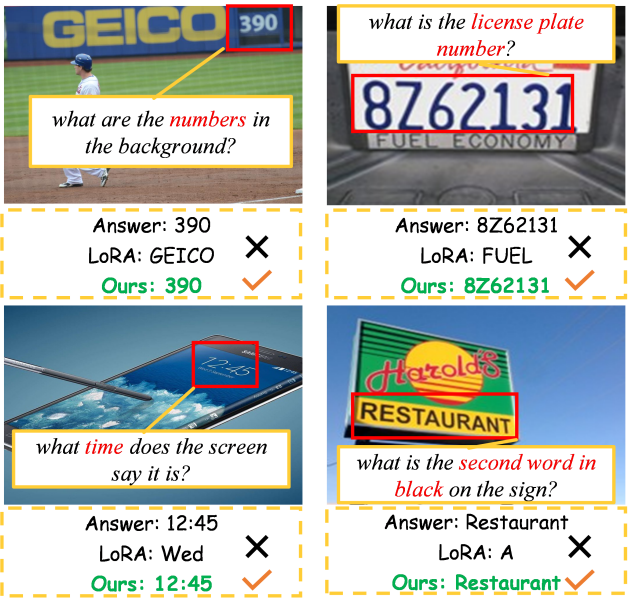


Figure 4: Case study on TextVQA. LoRA causes catastrophic forgetting and hallucinations (×), while Dual-Pronged LoRA gives accurate, instruction-following responses (✓).

We further observe that the misrouted TextVQA queries are structurally and semantically aligned with the downstream IconQA distribution: most follow a "what"-centric interrogative pattern and target localized visual attribute extraction or attribute-specific reasoning. This suggests that cross-task semantic overlap may contribute to these routing errors. Despite this, Dual-Pronged LoRA exhibits limited performance degradation overall, which may be partly explained by two complementary factors. First, the implicit regularization induced by DPO-RS may help stabilize the activated LoRA branch (Sec. 4.2.1). Second, the semantic proximity between TextVQA and IconQA may allow the erroneously activated LoRA to supply relevant reasoning priors rather than conflicting knowledge. Consequently, routing errors may result in limited interference in this setting, providing a possible explanation for the observed robustness of TDCR under *explicit task decoupling*.

6 Conclusion

We introduce Dual-Pronged LoRA to address catastrophic forgetting in LoRA-based MLLM fine-tuning while simultaneously maximizing downstream adaptation performance. By synergizing TDCR for precise task decoupling and DPO-RS for maximizing downstream performance, our approach breaks the conventional retention-adaptation trade-off, achieving near-zero forgetting together with state-of-the-art overall performance. Our study suggests that catastrophic forgetting in MLLM fine-tuning is fundamentally a structural entanglement problem, and that TDCR offers a new perspective for mitigating it through explicit task decoupling rather than shared-parameter competition. More importantly, TDCR's lightweight, plug-and-play nature allows it to generalize beyond LoRA to diverse fine-tuning architectures. Looking forward, we believe that explicit task decoupling provides a promising foundation for broader settings such as multi-task adaptation, continual learning, and lifelong learning, especially when task boundaries evolve over time.

Acknowledgements

This work was supported by the Shanghai Artificial Intelligence Laboratory and a locally commissioned task from the Shanghai Municipal Government. We also express our gratitude to our collaborators (Jingqi Ye, Haonan He, Minglei Li, and Fujun Han) for their valuable contributions during their internships at the Shanghai Artificial Intelligence Laboratory, both in-person and remotely.

References

[1] Muhammad Ahmad, Jingjing Zheng, and Yankai Cao. "On Catastrophic Forgetting in Low-Rank Decomposition-Based Parameter-Efficient Fine-Tuning". In: *arXiv preprint arXiv:2603.09684* (2026).

[2] Shuai Bai et al. *Qwen2.5-VL Technical Report*. 2025. arXiv: 2502.13923 [cs.CV]. URL: https://arxiv.org/abs/2502.13923.

[3] Shuai Bai et al. "Qwen3-vl technical report". In: *arXiv preprint arXiv:2511.21631* (2025).

[4] Tianlong Chen et al. "The principle of diversity: Training stronger vision transformers calls for reducing all levels of redundancy". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2022, pp. 12010–12020.

[5] Minkyoung Cho et al. "MARS: Harmonizing Multimodal Convergence via Adaptive Rank Search". In: *arXiv preprint arXiv:2603.00720* (2026).

[6] Tri Dao. "Flashattention-2: Faster attention with better parallelism and work partitioning". In: *arXiv preprint arXiv:2307.08691* (2023).

[7] Shihan Dou et al. "LoRAMoE: Alleviating world knowledge forgetting in large language models via MoE-style plugin". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2024, pp. 1932–1945.

[8] Zhangwei Gao et al. "Mini-internvl: a flexible-transfer pocket multi-modal model with 5% parameters and 90% performance". In: *Visual Intelligence* 2.1 (2024), p. 32.

[9] Chendi Ge et al. "Dynamic mixture of curriculum lora experts for continual multimodal instruction tuning". In: *arXiv preprint arXiv:2506.11672* (2025).

[10] Danna Gurari et al. "Vizwiz grand challenge: Answering visual questions from blind people". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE. 2018, pp. 3608–3617.

[11] Fujun Han and Peng Ye. "Mllm-isu: The first-ever comprehensive benchmark for multimodal large language models based intrusion scene understanding". In: *Advances in Neural Information Processing Systems* 38 (2026).

[12] Fujun Han et al. "Ada-id: Active domain adaptation for intrusion detection". In: *Proceedings of the 32nd ACM International Conference on Multimedia*. 2024, pp. 7404–7413.

[13] Fujun Han et al. "Can Multimodal Large Language Models Truly Understand Small Objects?" In: *arXiv preprint arXiv:2604.22884* (2026).

[14] Fujun Han et al. "Ovid: Open-vocabulary intrusion detection". In: *International Conference on Learning Representations*. Vol. 2026. 2026, pp. 64320–64341.

[15] Haonan He et al. "A Unified Study of LoRA Variants: Taxonomy, Review, Codebase, and Empirical Evaluation". In: *arXiv preprint arXiv:2601.22708* (2026).

[16] Haonan He et al. "Gora: Gradient-driven adaptive low rank adaptation". In: *arXiv preprint arXiv:2502.12171* (2025).

[17] Chenlong Hu et al. "One-class text classification with multi-modal deep support vector data description". In: *Journal of Natural Language Processing* 28.4 (2021), pp. 1053–1088.

[18] Edward J Hu et al. “Lora: Low-rank adaptation of large language models.” In: *Iclr* 1.2 (2022), p. 3.

[19] Jiajun Hu et al. “Learn to preserve and diversify: Parameter-efficient group with orthogonal regularization for domain generalization”. In: *European Conference on Computer Vision*. Springer. 2024, pp. 198–216.

[20] Wenke Huang, Mang Ye, and Bo Du. “Learn from others and be yourself in heterogeneous federated learning”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 10143–10153.

[21] Wenke Huang et al. “Generalizable heterogeneous federated cross-correlation and instance similarity learning”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46.2 (2023), pp. 712–728.

[22] Wenke Huang et al. “Learn from downstream and be yourself in multimodal large language model fine-tuning”. In: *arXiv preprint arXiv:2411.10928* (2024).

[23] Drew A Hudson and Christopher D Manning. “Gqa: A new dataset for real-world visual reasoning and compositional question answering”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 6700–6709.

[24] Hyeontaek Hwang, Nguyen Dinh Son, and Daeyoung Kim. “Model-Dowser: Data-Free Importance Probing to Mitigate Catastrophic Forgetting in Multimodal Large Language Models”. In: *arXiv preprint arXiv:2602.04509* (2026).

[25] Wenyuan Jiang et al. “Killing two birds with one stone: Cross-modal reinforced prompting for graph and language tasks”. In: *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2024, pp. 1301–1312.

[26] Yao Jiang et al. “Effectiveness assessment of recent large vision-language models”. In: *Visual Intelligence* 2.1 (2024), p. 17.

[27] Erel Joffe et al. “Expert guided natural language processing using one-class classification”. In: *Journal of the American Medical Informatics Association* 22.5 (2015), pp. 962–966.

[28] James Kirkpatrick et al. “Overcoming catastrophic forgetting in neural networks”. In: *Proceedings of the national academy of sciences* 114.13 (2017), pp. 3521–3526.

[29] Vignesh Kothapalli et al. “From Spikes to Heavy Tails: Unveiling the Spectral Evolution of Neural Networks”. In: *Transactions on Machine Learning Research* (2025).

[30] Firas Laakom et al. “Convolutional autoencoder-based multimodal one-class classification”. In: *2025 IEEE Symposium on Computational Intelligence in Image, Signal Processing and Synthetic Media Companion (CISM Companion)*. IEEE. 2025, pp. 1–5.

[31] Bo Li et al. “Llava-onevision: Easy visual task transfer”. In: *arXiv preprint arXiv:2408.03326* (2024).

[32] Chen Li et al. “Vision-language instruction tuning: A review and analysis”. In: *arXiv preprint arXiv:2311.08172* (2023).

[33] Lei Li et al. “Multimodal arxiv: A dataset for improving scientific comprehension of large vision-language models”. In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2024, pp. 14369–14387.

[34] Jian Liang et al. “Lorasculpt: Sculpting lora for harmonizing general and specialized knowledge in multimodal large language models”. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2025, pp. 26170–26180.

[35] Tsung-Yi Lin et al. “Microsoft coco: Common objects in context”. In: *European conference on computer vision*. Springer. 2014, pp. 740–755.

[36] Haotian Liu et al. “Improved baselines with visual instruction tuning”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2024, pp. 26296–26306.

[37] Haotian Liu et al. “Visual instruction tuning”. In: *Advances in neural information processing systems* 36 (2023), pp. 34892–34916.

[38] Shih-Yang Liu et al. “Dora: Weight-decomposed low-rank adaptation”. In: *Forty-first International Conference on Machine Learning*. 2024.

[39] Xialei Liu et al. “Rotate your networks: Better weight consolidation and less catastrophic forgetting”. In: *arXiv preprint arXiv:1802.02950* (2018).

[40] Pan Lu et al. “Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning”. In: *arXiv preprint arXiv:2110.13214* (2021).

[41] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. “Piggyback: Adapting a single network to multiple tasks by learning to mask weights”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 67–82.

[42] Arun Mallya and Svetlana Lazebnik. “Packnet: Adding multiple tasks to a single network by iterative pruning”. In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2018, pp. 7765–7773.

[43] Kenneth Marino et al. “Ok-vqa: A visual question answering benchmark requiring external knowledge”. In: *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*. 2019, pp. 3195–3204.

[44] Anand Mishra et al. "Ocr-vqa: Visual question answering by reading text in images". In: *2019 international conference on document analysis and recognition (ICDAR)*. IEEE. 2019, pp. 947–952.

[45] Bryan A Plummer et al. "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 2641–2649.

[46] Samyam Rajbhandari et al. "Zero: Memory optimizations toward training trillion parameter models". In: *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE. 2020, pp. 1–16.

[47] Anastasia Razdaibiedina et al. "Progressive prompts: Continual learning for language models". In: *arXiv preprint arXiv:2301.12314* (2023).

[48] Lukas Ruff et al. "Deep one-class classification". In: *International conference on machine learning*. PMLR. 2018, pp. 4393–4402.

[49] Lukas Ruff et al. "Self-attentive, multi-context one-class classification for unsupervised anomaly detection on text". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019, pp. 4061–4071.

[50] Mohammad Sabokrou et al. "Adversarially learned one-class classifier for novelty detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 3379–3388.

[51] Patrick Schlachter, Yiwen Liao, and Bin Yang. "Deep one-class classification using intra-class splitting". In: *2019 IEEE data science workshop (DSW)*. IEEE. 2019, pp. 100–104.

[52] Bernhard Schölkopf et al. "Estimating the Support of a High-Dimensional Distribution". In: *Neural Computation* 13.7 (2001), pp. 1443–1471. DOI: 10.1162/089976601750264965.

[53] Jonathan Schwarz et al. "Progress & compress: A scalable framework for continual learning". In: *International conference on machine learning*. PMLR. 2018, pp. 4528–4537.

[54] Ying Shen et al. "Multimodal instruction tuning with conditional mixture of lora". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2024, pp. 637–648.

[55] Amanpreet Singh et al. "Towards vqa models that can read". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 8317–8326.

[56] David MJ Tax and Robert PW Duin. "Support vector data description". In: *Machine learning* 54.1 (2004), pp. 45–66.

[57] Qwen Team. *Qwen2.5-VL*. Jan. 2025. URL: https://qwenlm.github.io/blog/qwen2.5-vl/.

[58] Liyuan Wang et al. "A comprehensive survey of continual learning: Theory, method and application". In: *IEEE transactions on pattern analysis and machine intelligence* 46.8 (2024), pp. 5362–5383.

[59] Zhicheng Wang et al. "Rehearsal-free continual language learning via efficient parameter isolation". In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2023, pp. 10933–10946.

[60] Zifeng Wang et al. "Learn-prune-share for lifelong learning". In: *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE. 2020, pp. 641–650.

[61] Zifeng Wang et al. "Sparcl: Sparse continual learning on the edge". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 20366–20380.

[62] Ziqi Wang et al. "Smolora: Exploring and defying dual catastrophic forgetting in continual visual instruction tuning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2025, pp. 177–186.

[63] Junda Wu et al. "Mitigating visual knowledge forgetting in mllm instruction-tuning via modality-decoupled gradient descent". In: *arXiv preprint arXiv:2502.11740* 8 (2025).

[64] Xun Wu, Shaohan Huang, and Furu Wei. "Mixture of lora experts". In: *arXiv preprint arXiv:2404.13628* (2024).

[65] Jingqi Ye et al. "Gradient intrinsic dimensionality alignment: Narrowing the gap between low-rank adaptation and full fine-tuning". In: *International Conference on Learning Representations*. Vol. 2026. 2026, pp. 55765–55790.

[66] Peng Ye, Yuchen Ren, et al. "Gora: Gradient-driven adaptive low rank adaptation". In: *Advances in Neural Information Processing Systems* 38 (2026), pp. 114074–114102.

[67] Peng Ye et al. "β-DARTS: Beta-Decay Regularization for Differentiable Architecture Search". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2022, pp. 10864–10873.

[68] Peng Ye et al. "β-DARTS++: Bi-level Regularization for Proxy-robust Differentiable Architecture Search". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2025).

[69] Peng Ye et al. "Dynamic model merging with mixture of weights". In: *IEEE Transactions on Circuits and Systems for Video Technology* 35.8 (2025), pp. 7925–7939.

[70] Peng Ye et al. “Stimulative training++: Go beyond the performance limits of residual networks”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 47.8 (2025), pp. 6463–6479.

[71] Jaehong Yoon et al. “Lifelong learning with dynamically expandable networks”. In: *arXiv preprint arXiv:1708.01547* (2017).

[72] Le Yu et al. “Language models are super mario: Absorbing abilities from homologous models as a free lunch”. In: *Forty-first International Conference on Machine Learning*. 2024.

[73] Yuexiang Zhai et al. “Investigating the catastrophic forgetting in multimodal large language model fine-tuning”. In: *Conference on Parsimony and Learning*. PMLR. 2024, pp. 202–227.

[74] Qingru Zhang et al. “Adalora: Adaptive budget allocation for parameter-efficient fine-tuning”. In: *arXiv preprint arXiv:2303.10512* (2023).

[75] Yuanhe Zhang, Fanghui Liu, and Yudong Chen. “LoRA-One: One-Step Full Gradient Could Suffice for Fine-Tuning Large Language Models, Provably and Efficiently”. In: *International Conference on Learning Representations*. 2025.

[76] Shanshan Zhong et al. “Moextend: Tuning new experts for modality and task extension”. In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*. 2024, pp. 494–505.

[77] Xiongtao Zhou et al. “An empirical study on parameter-efficient fine-tuning for multimodal large language models”. In: *Findings of the Association for Computational Linguistics: ACL 2024*. 2024, pp. 10057–10084.

[78] Didi Zhu et al. “Model tailor: Mitigating catastrophic forgetting in multi-modal large language models”. In: *arXiv preprint arXiv:2402.12048* (2024).

Appendix of Dual-Pronged LoRA: Achieving Near-Zero Forgetting and High-Performance Adaptation for MLLMs

In this appendix, we provide detailed theoretical proofs and experimental details.

- **Appendix A**: Theoretical Analysis and Mathematical Justification for DPO-RS.
- **Appendix B**: Optimization Dynamics of LoRA.
- **Appendix C**: Full Training Procedure of Dual-Pronged LoRA.
- **Appendix D**: Comprehensive Experimental Setups.
- **Appendix E**: Additional Analyses of TDCR.
- **Appendix F**: Additional Evaluation on Similar Tasks and Routing-Based Methods.
- **Appendix G**: Limitations.

A Theoretical Analysis and Mathematical Justification for DPO-RS

In this section, we provide a theoretical motivation for using the gradient Frobenius norm, $I_l = \|G_l\|_F$, as the layer-wise importance proxy in DPO-RS. Our analysis follows a simple logic: (1) we first characterize the ideal single-step loss reduction of a full-rank update; (2) we then quantify how much optimization capacity is inevitably lost under a low-rank constraint; (3) based on this characterization, we formulate the global rank-allocation problem; and (4) we finally explain why the Frobenius norm serves as a lightweight and effective surrogate for practical rank pre-allocation.

A.1 Reference Single-Step Loss Reduction

Let $\mathcal{L}$ denote the downstream fine-tuning loss. For the l-th layer, let $G_l = \nabla_{W_l}\mathcal{L}$ denote the gradient of the loss with respect to the equivalent weight matrix W_l. Under a first-order Taylor expansion, the loss change caused by an arbitrary update ΔW_l can be approximated as

$$\Delta\mathcal{L}_l \approx \langle G_l, \Delta W_l\rangle = \mathrm{Tr}(G_l^\top \Delta W_l). \tag{14}$$

As a reference case, consider one standard full-gradient descent step with learning rate η, namely

$$\Delta W_l^\star = -\eta G_l. \tag{15}$$

Substituting this into Eq. (14) gives

$$\Delta\mathcal{L}_l^\star \approx -\eta\,\mathrm{Tr}(G_l^\top G_l) = -\eta\|G_l\|_F^2. \tag{16}$$

This quantity serves as a reference single-step loss reduction for the unconstrained full-rank update. It shows that the total gradient energy $\|G_l\|_F^2$ determines the maximum reduction achieved by a standard gradient step of size η.

A.2 Information Loss Induced by Low-Rank Constraints

We next consider the effect of the LoRA rank constraint. As derived in Appendix B, the effective first-order update induced by LoRA at training step t can be written as

$$\Delta W_t^{\mathrm{LoRA}} \approx -\eta\left(B_tB_t^\top G_t + G_tA_t^\top A_t\right). \tag{17}$$

Since each term has rank at most r_l, the overall effective update satisfies

$$\mathrm{rank}\big(\Delta W_t^{\mathrm{LoRA}}\big) \le 2r_l. \tag{18}$$

Therefore, LoRA can be viewed as producing a rank-constrained approximation to the ideal full-rank update. Let the singular values of G_l be

$$\sigma_1^{(l)} \geq \sigma_2^{(l)} \geq \cdots \geq \sigma_{d^{(l)}}^{(l)} \geq 0, \tag{19}$$

where $d^{(l)} = \min(d_{\text{in}}^{(l)}, d_{\text{out}}^{(l)})$. According to the Eckart–Young–Mirsky theorem, among all matrices of rank at most $2r_l$, the best approximation to G_l in the Frobenius sense preserves the top $2r_l$ singular components. Hence, the largest first-order loss reduction achievable by any rank-$2r_l$ update is

$$\overline{\Delta\mathcal{L}}_l^{(2r_l)} \approx -\eta \sum_{i=1}^{2r_l} \left(\sigma_i^{(l)}\right)^2. \tag{20}$$

Compared with the reference full-rank reduction in Eq. (16), the unrecovered part is given by

$$\mathcal{E}_{\text{loss}}^{(l)} = \left|\Delta\mathcal{L}_l^\star - \overline{\Delta\mathcal{L}}_l^{(2r_l)}\right| \approx \eta \sum_{i=2r_l+1}^{d^{(l)}} \left(\sigma_i^{(l)}\right)^2. \tag{21}$$

This term measures the optimization information that cannot be captured under the rank constraint. In other words, the low-rank restriction inevitably discards the tail gradient energy beyond the top $2r_l$ singular directions.

A.3 Global Rank Allocation under a Parameter Budget

Given a total parameter budget $\mathcal{B}$, the goal is to distribute layer-wise ranks $\{r_l\}$ such that the total unrecovered optimization information is minimized:

$$\begin{aligned} &\min_{\{r_l\}} \sum_{l=1}^{L} \mathcal{E}_{\text{loss}}^{(l)} = \eta \sum_{l=1}^{L} \sum_{i=2r_l+1}^{d^{(l)}} \left(\sigma_i^{(l)}\right)^2 \\ &\text{s.t.} \quad \sum_{l=1}^{L} r_l \left(d_{\text{in}}^{(l)} + d_{\text{out}}^{(l)}\right) \leq \mathcal{B}, \quad r_{\min} \leq r_l \leq r_{\max}, \quad r_l \in \mathbb{Z}_+. \end{aligned} \tag{22}$$

Eq. (22) makes the objective of DPO-RS explicit: **under a limited budget, more rank should be allocated to layers whose low-rank truncation would otherwise incur larger optimization loss**.

A.4 From Exact Optimization to a Lightweight Surrogate

Although minimizing Eq. (22) mathematically minimizes the optimization information loss, solving it exactly requires continuously performing Singular Value Decomposition (SVD) on the gradient matrices of all layers throughout the training process. This incurs an unbearable time complexity of $\mathcal{O}(d^3)$ per step, defeating the efficiency purpose of PEFT. To bypass this bottleneck, we introduce a static, SVD-free surrogate that relies on the initial gradient Frobenius norm, which is theoretically justified from two perspectives.

(1) The Rationale for Pre-training ($t = 0$) Estimation

Instead of dynamically tracking gradients, DPO-RS computes the gradient exclusively on a single downstream mini-batch at $t = 0$ (i.e., before fine-tuning begins, when the equivalent weight is exactly the pretrained backbone W_0). This design is not merely a computational compromise, but a theoretical necessity for unbiased capacity allocation. At $t = 0$, the gradient $G_l|_{W_0}$ captures the *intrinsic representational gap* between the pre-trained upstream knowledge and the specific downstream task requirements. Once fine-tuning commences ($t > 0$), the gradient G_t becomes conditioned on the restricted low-rank subspace already fitted by the LoRA modules. Consequently, $G_{t>0}$ primarily reflects the residual error rather than the holistic capacity demand, making it biased by LoRA's initialization and optimization trajectory. Therefore, the pre-training gradient $G_l|_{W_0}$ provides the purest, uncontaminated signal for how much adaptation capacity a specific layer intrinsically demands to fit the downstream distribution.

(2) From Spectral Tail to Frobenius Norm Surrogate

Even with $G_l|_{W_0}$ fixed, Eq. (22) dictates that the optimal rank depends strictly on the energy of the *tail* singular values $\sum_{i>2r_l}(\sigma_i^{(l)})^2$, rather than the total energy. However, prior spectral analyses of trained deep neural networks have repeatedly reported heavy-tailed spectral structures in layer matrices, suggesting that spectral energy is often concentrated in a relatively small number of dominant directions [29]. Motivated by these observations, we adopt a mild approximation that the unrecovered tail energy after low-rank truncation can be approximately ordered by the total gradient energy $\|G_l\|_F^2$. Therefore, instead of explicitly computing the full singular spectrum for each layer, we use the initial gradient Frobenius norm as a lightweight proxy for layer-wise rank demand. This choice is also consistent with recent theoretical results showing that LoRA is closely related to the singular subspaces of the one-step full fine-tuning gradient [75]. Consequently, a layer with a larger $\|G_l\|_F$ at $t = 0$ is expected to incur a larger truncation loss under the same low-rank constraint, and thus should receive a larger rank budget.

(3) Final Rank Sculpting Formulation

By linearly binding the rank allocation to the Frobenius norm of the initial gradient, we prioritize the global parameter budget for critical layers capable of recovering the most $\mathcal{E}_{\text{loss}}^{(l)}$. Normalizing this proxy across all L candidate layers yields the relative importance score:

$$p_l = \frac{\|G_l\|_F}{\sum_{k=1}^{L}\|G_k\|_F} \tag{23}$$

Accounting for the fact that the parameter cost of a rank-r_l LoRA module is $r_l(d_{\text{in}}^{(l)} + d_{\text{out}}^{(l)})$, the allocated rank under the global parameter budget $\mathcal{B}$ is scaled accordingly:

$$r_l = \max\left(r_{\min}, \min\left(r_{\max}, \left\lfloor \frac{\mathcal{B}\cdot p_l}{\sqrt{d_{\text{in}}^{(l)} + d_{\text{out}}^{(l)}}} \right\rfloor\right)\right) \tag{24}$$

where $r_{\min}$ and $r_{\max}$ enforce structural stability bounds. This closed-form solution achieves a principled approximation of the optimal rank distribution while maintaining strict computational efficiency.

B Optimization Dynamics of LoRA

In this section, we derive the effective first-order update induced by LoRA under SGD. This result explains why LoRA can be viewed as a rank-constrained approximation to the ideal full-gradient update.

Let the pretrained weight matrix be $W_0 \in \mathbb{R}^{d_{\text{out}}\times d_{\text{in}}}$. LoRA parameterizes the weight update as

$$\Delta W = \frac{\alpha}{r}BA, \tag{25}$$

where $A \in \mathbb{R}^{r\times d_{\text{in}}}$ and $B \in \mathbb{R}^{d_{\text{out}}\times r}$. For simplicity, we absorb the constant factor α/r into the learning rate and write the effective weight as

$$W = W_0 + BA. \tag{26}$$

Accordingly, for an input X, the forward pass is

$$Y = (W_0 + BA)X. \tag{27}$$

Let

$$G = \frac{\partial \mathcal{L}}{\partial W} \tag{28}$$

denote the gradient of the loss with respect to the equivalent weight matrix. By the chain rule, the gradients with respect to the LoRA factors are

$$\nabla_A \mathcal{L} = B^\top G, \qquad \nabla_B \mathcal{L} = GA^\top. \tag{29}$$

Under SGD with learning rate η, the updates are

$$A_{t+1} = A_t - \eta B_t^\top G_t, \qquad B_{t+1} = B_t - \eta G_t A_t^\top. \tag{30}$$

The corresponding change in the equivalent weight is

$$\Delta W_t^{\mathrm{LoRA}} = B_{t+1}A_{t+1} - B_t A_t. \tag{31}$$

Substituting the SGD updates yields

$$\begin{aligned} \Delta W_t^{\mathrm{LoRA}} &= (B_t - \eta G_t A_t^\top)(A_t - \eta B_t^\top G_t) - B_t A_t \\ &= -\eta \left(B_t B_t^\top G_t + G_t A_t^\top A_t \right) + \eta^2 G_t A_t^\top B_t^\top G_t. \end{aligned} \tag{32}$$

Since the learning rate is small in practical fine-tuning, the second-order term is negligible. Therefore, the effective first-order update can be approximated as

$$\Delta W_t^{\mathrm{LoRA}} \approx -\eta \left(B_t B_t^\top G_t + G_t A_t^\top A_t \right). \tag{33}$$

This shows that LoRA does not freely follow the full gradient G_t. Instead, **it restricts the update to a low-rank subspace determined by the current LoRA factors**. Since each term in Eq. (33) has rank at most r, the total effective update has rank at most $2r$:

$$\mathrm{rank}\big(\Delta W_t^{\mathrm{LoRA}}\big) \leq 2r. \tag{34}$$

This rank ceiling provides the foundation for the information-loss analysis in Appendix A.

C Full Training Procedure of Dual-Pronged LoRA

For completeness, we provide the full training procedure of Dual-Pronged LoRA, including the pre-training rank sculpting stage of DPO-RS and the joint optimization stage of TDCR, in Algorithm 1.

D Comprehensive Experimental Setups

D.1 Training Configurations

Datasets and Model. All experiments are built upon the LLaVA-1.5 [36] architecture. For downstream adaptation, we fine-tune and evaluate the model on two representative visual tasks: Visual Question Answering (VQA) using IconQA [40] and image captioning using COCO-Caption [35]. To assess the mitigation of catastrophic forgetting, the fine-tuned models are subsequently evaluated on four datasets originally included in the LLaVA-1.5 training corpus [77, 34]: OKVQA [43], OCRVQA [44], GQA [23], and TextVQA [55]. Furthermore, in Sec. 5.1, we conduct additional downstream experiments on ArxivQA [33] and Flickr30k [45] to validate the generalization of our proposed method across diverse tasks and data distributions.

Implementation Details. For all baseline methods, we follow the established experimental protocols and recommended configurations from prior work [34], with method-specific settings kept consistent with their original formulations whenever applicable. All methods are implemented within the same LLaVA-1.5[2] framework and evaluated under the same downstream adaptation setting. For Dual-Pronged LoRA, unless otherwise specified, we set $\lambda_1 = 1 \times 10^{-1}$, $\lambda_2 = 1 \times 10^{-3}$, $K = 5$, the contextual support threshold $\tau = 0.7$, and the inference-time tolerance margin $\delta = 0.12$. All experiments are implemented in PyTorch and conducted on two NVIDIA H200 GPUs, with DeepSpeed ZeRO-2 [46] and FlashAttention-2 [6] used for efficient training.

D.2 Evaluation Metrics

We employ a dual-metric evaluation scheme to simultaneously capture task-specific acquisition and general knowledge retention. The adaptation capability on the downstream task is denoted as *Target*, calculated using standard dataset-specific metrics (Accuracy for VQA and CIDEr for image

[2] https://github.com/haotian-liu/LLaVA

Algorithm 1 Training Procedure of Dual-Pronged LoRA

Require: Pretrained weights $\{W_0^{(l)}\}_{l=1}^{L}$, global budget $\mathcal{B}$, rank bounds $r_{\min}, r_{\max}$, target threshold τ
Ensure: Sculpted LoRA matrices $\{A_l, B_l\}_{l=1}^{L}$, context matrix W_R

Stage 1: DPO-RS Rank Sculpting
Sample a downstream mini-batch and compute $\mathcal{L}_{\text{CE}}$ ▷ before training
$I_l \leftarrow \left\| \nabla_{W_0^{(l)}} \mathcal{L}_{\text{CE}} \right\|_F, \ \forall l \in [1, L]$
$p_l \leftarrow I_l / \sum_{k=1}^{L} I_k$
for $l = 1$ to L **do**
    $r_l \leftarrow \text{Clip}\left(\left\lfloor \frac{\mathcal{B}\, p_l}{\sqrt{d_{\text{in}}^{(l)} + d_{\text{out}}^{(l)}}} \right\rfloor, r_{\min}, r_{\max} \right)$
end for

Stage 2: TDCR Joint Training
$W_R \sim \mathcal{O}(\text{Orthogonal}) \in \mathbb{R}^{d \times K}$, $A_l \sim \mathcal{U}(\text{Kaiming})$, $B_l \leftarrow \mathbf{0}$
for training batch $\{X^{(b)}\}_{b=1}^{B}$, where $X^{(b)} = (x_1^{(b)}, \ldots, x_L^{(b)})$ **do**
Instruction Mask
    for $b = 1$ to B **do**
        $p^{(b)} \leftarrow \max\{\, i \mid x_i^{(b)} = v_{\text{img}} \,\} + 1$ ▷ v_{img}: Image Token
        $q^{(b)} \leftarrow \min\{\, (x_i^{(b)}, \ldots, x_{i+k-1}^{(b)}) = V_{\text{ast}} \,\}$ ▷ V_{ast}: "ASSISTANT:"
        $m_i^{\text{ins},(b)} \leftarrow \mathbb{I}(p^{(b)} \le i < q^{(b)}), \ \forall i \in [1, L]$
    end for
Instruction-Aware Pooling & Similarity
    $h_{\text{pooled}} \leftarrow \frac{\sum_{i=1}^{L} h_i \cdot m_i^{\text{ins}}}{\sum_{i=1}^{L} m_i^{\text{ins}} + \epsilon}$ ▷ h_i: input embedding of x_i
    $\bar{h}_{\text{pooled}} \leftarrow \frac{h_{\text{pooled}}}{\|h_{\text{pooled}}\|_2}, \quad \bar{W}_R \leftarrow \frac{W_R}{\|W_R\|_2}$
    $s \leftarrow \bar{h}_{\text{pooled}}^{\top} \bar{W}_R$
TDCR Objective Computation
    $\mathcal{L}_{\text{compact}} \leftarrow \text{ReLU}(\tau - \max(s))$
    $\mathcal{L}_{\text{diverse}} \leftarrow -\text{Var}(s) \cdot \mathbb{I}(\mathcal{L}_{\text{compact}} > 0)$
    $\mathcal{L}_{\text{TDCR}} \leftarrow \lambda_1 \mathcal{L}_{\text{compact}} + \lambda_2 \mathcal{L}_{\text{diverse}}$
Task-Decoupled Optimization
    $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{TDCR}}$
    Update $\{A_l, B_l\}_{l=1}^{L}$ and W_R via $\nabla \mathcal{L}_{\text{total}}$
end for
return $\{A_l, B_l\}_{l=1}^{L}$, W_R

captioning). To assess the preservation of pre-trained capabilities, we define the *Source* metric as the unweighted arithmetic mean of the performance scores across the four upstream datasets. Ultimately, to quantify the global proficiency of the model and its success in overcoming the retention-adaptation trade-off, we report the *Avg* score, mathematically formulated as:

$$Source = \frac{1}{|\mathcal{D}|} \sum_{\mathcal{D}_i \in \mathcal{D}} \text{Score}(\mathcal{D}_i), \quad Target = \text{Score}(\mathcal{T}) \tag{35}$$

$$Avg = \frac{Source + Target}{2} \tag{36}$$

where $\text{Score}(\cdot)$ represents the corresponding evaluation metric, $\mathcal{D}$ denotes the set of upstream evaluation datasets, and $\mathcal{T}$ represents the downstream target task.

D.3 Brief overview of baselines

- LoRA [ICLR'22] [18]: Approximates pretrained weight updates using the product of two low-rank matrices.
- DoRA [ICML'24] [38]: Decomposes model weights into two components—magnitude and direction. By exclusively updating the directional component, it enhances both the feature representation capacity and the training stability of traditional LoRA.

- AdaLoRA [ICLR'23] [74]: Dynamically adjusts layer-wise LoRA ranks during fine-tuning according to importance scores under a fixed parameter budget.
- GoRA[NeurIPS'25] [66]: Uses gradient-informed importance scores obtained before fine-tuning to adaptively allocate layer-wise LoRA ranks under a fixed parameter budget.
- L2-Regularization [PNAS'17] [28]: Employs a standard L2 regularization strategy (with a coefficient of 1e-3) to constrain LoRA parameter updates. By limiting the deviation of the fine-tuned model from its initial pretrained state, it mitigates catastrophic forgetting in downstream tasks.
- Orth-Reg [ECCV'24] [19]: Imposes an orthogonal regularization constraint on the LoRA weights (with the hyperparameter set to 1e-3), encouraging the features learned during fine-tuning to remain orthogonal to the pretrained features.
- Model Tailor [ICML'24] [78]: Integrates parameter salience and sensitivity analyses to selectively replace a marginal fraction (approximately 10%) of the fine-tuned parameters, while retaining the vast majority of the original pretrained weights.
- DARE [ICML'24] [72]: Executes random dropping and proportional rescaling operations on the fine-tuned LoRA parameters, effectively alleviating knowledge conflicts between the target task and the model's pre-trained capabilities.
- LoRAMoE [ACL'24] [7]: Introduces a MoE-style plugin that employs multiple LoRA as experts, integrated via a router network with a localized balancing constraint. By dedicating specific experts to leverage world knowledge, it alleviates catastrophic forgetting of pre-trained knowledge during large-scale supervised fine-tuning while enhancing multi-task performance.
- LoRASculpt [CVPR'25] [34]: Introduces sparse updates into LoRA training to eliminate harmful redundant parameters, coupled with a Conflict Mitigation Regularizer that guides task-specific updates toward less critical pretrained parameters. By harmonizing general and specialized knowledge, it simultaneously enhances downstream task performance and mitigates catastrophic forgetting.

D.4 Details of Datasets

To comprehensively evaluate both the downstream adaptation performance and the retention of upstream pre-trained knowledge, we categorize the utilized datasets into two distinct groups. For all downstream adaptation tasks, we randomly sample a subset of 10K data from their respective training splits.

Downstream Adaptation Datasets:

- **IconQA** [40] investigates multimodal reasoning within the context of abstract diagrams and iconography. In this work, our evaluation is specifically conducted on its text-based multiple-choice subset.
- **COCO-Caption** [35] provides rich annotations for complex everyday scenes. It stands as a foundational benchmark for assessing multimodal alignment and natural language generation in image captioning tasks.
- **Flickr30k** [45] comprises 30,000 everyday images paired with multiple crowd-sourced textual descriptions. It is extensively utilized to benchmark cross-modal understanding and evaluate the semantic fidelity and descriptive quality of vision-language models.
- **ArXivQA** [33] challenges models to comprehend abstract scientific figures through multiple-choice question-answering pairs on real academic diagrams. It serves as a rigorous benchmark for evaluating multimodal mathematical reasoning and domain-specific scientific understanding, addressing the critical gap in large vision-language models' ability to interpret complex visualizations across diverse scientific domains.

Upstream Knowledge Datasets:

- **TextVQA** [55] assesses VQA capabilities by demanding the extraction and comprehension of scene text within images. It specifically tests the integration of optical character recognition (OCR) with multimodal reasoning, addressing the shortcomings of traditional VQA frameworks in text-rich environments.
- **OKVQA** [43] challenges models to go beyond explicit visual cues by incorporating external, commonsense, and factual knowledge bases. It serves as a rigorous benchmark for evaluating the fusion of visual perception with open-domain, knowledge-driven reasoning.
- **OCRVQA** [44] is dedicated to tasks where answering queries heavily relies on deciphering text situated within visual scenes. It gauges a model's proficiency in jointly executing OCR and contextualizing the recognized text against the surrounding visual elements.
- **GQA** [23] targets compositional visual reasoning through complex, multi-step queries about real-world images. By focusing on the structural relationships, objects, and attributes, it facilitates a granular evaluation of spatial and relational understanding.

E Additional Analyses of TDCR

E.1 Sensitivity to Routing Hyperparameters

We investigate the sensitivity of TDCR to the compactness threshold τ and the inference-time tolerance margin δ, which jointly determine the routing boundary (Sec. 3.2). We vary each hyperparameter independently on IconQA while keeping other settings fixed. The results are summarized in Table 5.

Table 5: Sensitivity analysis of the TDCR routing hyperparameters τ and δ on IconQA. Upstream Acc. denotes the proportion of upstream samples correctly routed to the pretrained path, while IconQA TPR denotes the proportion of downstream samples correctly routed to the LoRA branch.

Parameter	Value	*Target*	*Source*	*Avg*	Upstream Acc. (%)	IconQA TPR (%)
τ	0.3	85.96	60.42	73.19	92.52	100.00
	0.5	85.96	61.09	73.53	99.92	100.00
	0.7	85.96	60.57	73.27	92.99	100.00
	0.9	63.44	61.09	62.27	100.00	60.15
δ	0.06	85.96	61.05	73.50	99.38	100.00
	0.12	85.96	60.57	73.27	92.99	100.00
	0.18	85.96	59.44	72.70	79.31	100.00
	0.24	85.96	54.69	70.32	3.48	100.00

As shown in Table 5, TDCR maintains stable downstream performance and near-zero upstream forgetting within moderate ranges of $\tau \in [0.3, 0.7]$ and $\delta \in [0.06, 0.12]$. However, extreme values can lead to overly restrictive or permissive routing boundaries, reducing downstream TPR to 60.15% ($\tau = 0.9$) or upstream routing accuracy to 3.48% ($\delta = 0.24$). These results demonstrate that TDCR is robust across a reasonable range of hyperparameter settings without requiring precise tuning.

E.2 Cross-Architecture Generalization of TDCR

To examine whether the contextual support region learned by TDCR generalizes beyond LLaVA-1.5 [36], we conduct additional TDCR-only routing evaluations on Mini-InternVL-Chat-4B-V1-5 [8] and Qwen2.5-VL-3B-Instruct [57], using their respective instruction templates. We evaluate routing separability through the TPR on downstream IconQA samples and the FPR on upstream samples.

Table 6: Cross-architecture routing performance of TDCR under different instruction templates. TPR measures correct downstream activation, while FPR measures incorrect upstream activation.

Architecture	IconQA TPR (%)	Upstream FPR (%)
LLaVA-1.5 [36]	100.00	7.01
Mini-InternVL-Chat-4B-V1-5 [8]	90.48	0.00
Qwen2.5-VL-3B-Instruct [57]	99.38	2.51

Table 7: Similar-task evaluation on VizWiz [10].

Method	OKVQA	OCRVQA	GQA	TextVQA	Source	Target
Zero-shot	57.99	66.25	61.93	58.21	61.10	50.98
LoRA	47.80	42.10	57.06	51.24	49.55	52.12
Dual-Pronged LoRA	57.26	59.65	61.84	58.21	59.24	**52.72**

Table 8: Comparison with D-MoLE on IconQA and COCO-Caption Dataset.

Task	Method	α	Source	Target	Avg	TPR (%)	FPR (%)
IconQA	Dual-Pronged LoRA	–	**60.57**	**85.85**	**73.21**	–	–
	D-MoLE [9]	0.1 / 0.5	45.75	45.11	45.43	0.00	0.00
	D-MoLE [9]	1	0.60	36.80	18.70	100.00	99.05
	D-MoLE [9]	2 / 10	0.05	36.80	18.42	100.00	100.00
COCO-Caption	Dual-Pronged LoRA	–	**61.09**	**124.07**	**92.58**	–	–
	D-MoLE [9]	0.1 / 0.5	57.08	66.08	61.58	0.00	0.00
	D-MoLE [9]	1	50.04	116.45	83.24	98.66	61.91
	D-MoLE [9]	2 / 10	43.88	116.85	80.37	100.00	100.00

As shown in Table 6, TDCR maintains effective routing separability across different MLLM architectures and instruction templates.

F Additional Evaluation on Similar Tasks and Routing-Based Methods

F.1 Similar-Task Evaluation on VizWiz

We evaluate adaptation to VizWiz [10], which shares the image–question–answer format of the upstream VQA benchmarks. As shown in Table 7, Dual-Pronged LoRA improves Source and Target over LoRA by 9.69 and 0.60 points, respectively. The limited Source degradation relative to zero-shot may be attributed to semantic buffering, whereby semantically related knowledge in the adapted LoRA branch mitigates forgetting even when routing errors occur.

F.2 Comparison with D-MoLE

We compare against D-MoLE [9] under matched trainable-parameter budgets, using 1% of data for gradient sensitivity and selecting the top 50% of blocks. Its reconstruction-error routing threshold is scaled from the 99th percentile (Q_{99}) by α. Table 8 shows that D-MoLE is sensitive to this threshold: low values reject downstream inputs, while high values activate the adapted path for most upstream inputs. Dual-Pronged LoRA achieves higher Source and Target scores in both tasks under this protocol.

F.3 Conceptual Comparison with MoLE

MoLE [64] addresses the composition of multiple independently trained LoRA experts through learnable gating. In contrast, our setting assumes a single downstream adaptation task without access to the original upstream training data. TDCR estimates the downstream contextual support region using downstream examples and conditionally selects either the adapted LoRA path or the frozen pretrained path at inference. Thus, MoLE primarily concerns the composition of existing adapters like [69], whereas TDCR targets input-dependent preservation of pretrained capabilities during single-task adaptation. We report this as a conceptual comparison rather than a direct numerical benchmark, since the original MoLE setup presupposes multiple trained adapters.

G Limitations

This paper addresses catastrophic forgetting in LoRA-based MLLM fine-tuning from the perspective of *explicit task decoupling*. To this end, we introduce TDCR to separate upstream knowledge retention from downstream adaptation, and DPO-RS to allocate layer-wise LoRA ranks according

to downstream adaptation demand. Although our empirical results consistently demonstrate strong forgetting mitigation and downstream performance gains, several limitations remain.

First, the effectiveness of TDCR currently remains primarily empirical. While our experiments show that the learned contextual support region can successfully distinguish downstream-relevant inputs from general upstream ones, the routing behavior is still governed by a learned approximation based only on downstream data. Its robustness under more ambiguous task boundaries, substantial semantic overlap, or more diverse real-world input distributions still requires broader empirical validation and stronger theoretical analysis.

Second, the rank sculpting strategy in DPO-RS relies on a lightweight surrogate rather than an exact solution to the constrained optimization objective. In particular, using the initial full-gradient Frobenius norm as a proxy for layer-wise adaptation demand is motivated by our analysis, but still depends on simplifying assumptions about the relationship between total gradient energy and the tail energy discarded under low-rank truncation. Although this approximation works well in our experiments, its validity may vary across architectures, tasks, and optimization regimes, and deserves further theoretical and empirical study.

Moreover, although TDCR is lightweight and conceptually orthogonal to specific low-rank parameterizations, in this work we only study its integration with LoRA under standard single-task downstream adaptation settings. Its compatibility with other PEFT methods or broader fine-tuning paradigms has not yet been systematically explored. In particular, we have not investigated more complex scenarios such as multi-task adaptation, continual learning, or lifelong learning, where task boundaries may evolve over time and routing decisions can become substantially more challenging. Owing to its lightweight and plug-and-play nature, TDCR has the potential to extend to these broader settings, but validating such generality remains an important direction for future research.

Finally, the *semantic buffering* explanation for TDCR's resilience to routing errors remains preliminary and task-dependent. In the TextVQA case study, although 27.54% of TextVQA queries are misrouted to the LoRA branch, the overall score decreases by only 2.11 points relative to zero-shot, suggesting that semantic overlap with IconQA may sometimes alleviate the effects of misrouting. However, this observation does not establish semantic buffering as a general robustness mechanism, and its effectiveness under different task relationships remains unclear. Future work will investigate this hypothesis more systematically, especially under challenging settings with substantial upstream–downstream distributional overlap.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction highlight underexplored challenges in the LoRA fine-tuning of MLLMs and clearly position the contributions of this work in addressing them.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss limitations in the Appendix G.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [Yes]

Justification: Detailed theoretical and mathmatical justification are provided in Appendix A

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: All the experimental details are provided in Appendix D.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: Yes

Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: All the experimental details are provided in Appendix D.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [N/A]

   Justification: Due to the high computational cost of MLLM fine-tuning, we do not report error bars or repeated-run statistical tests.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Discussed in Appendix D.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We are sure that our research conforms NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [N/A]

    Justification: Our work does not pose any known societal risks or negative impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: Our research is purely fundamental research about LoRA, which cannot pose such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All models and datasets referenced in this paper are fully and appropriately cited.

    Guidelines:

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

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [N/A]

    Justification: This paper does not introduce or release new datasets, code, model checkpoints, or other assets.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

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