
000 FINE-TUNING MLLMs WITHOUT FORGETTING 001 002 IS EASIER THAN YOU THINK 003 004

005 **Anonymous authors**

006 Paper under double-blind review

007 008 ABSTRACT 009

010 The paper demonstrate that simple adjustments of the fine-tuning recipes of mul-
011 timodal large language models (MLLM) are sufficient to mitigate catastrophic
012 forgetting. On visual question answering, we design a 2x2 experimental frame-
013 work to assess model performance across in-distribution and out-of-distribution
014 image and text inputs. Our results show that appropriate regularization, such
015 as constraining the number of trainable parameters or adopting a low learning
016 rate, effectively prevents forgetting when dealing with out-of-distribution images.
017 However, we uncover a distinct form of forgetting in settings with in-distribution
018 images and out-of-distribution text. We attribute this forgetting as task-specific
019 overfitting and address this issue by introducing a data-hybrid training strategy
020 that combines datasets and tasks. Finally, we demonstrate that this approach natu-
021 rally extends to continual learning, outperforming existing methods with complex
022 auxiliary mechanisms. In general, our findings challenge the prevailing assump-
023 tions by highlighting the inherent robustness of MLLMs and providing practical
024 guidelines for adapting them while preserving their general capabilities.

025 026 1 INTRODUCTION 027

028 The remarkable success of **multimodal large language models (MLLMs)** in general-purpose visual
029 reasoning (Alayrac et al., 2022; Liu et al., 2023; Achiam et al., 2023) has spurred significant inter-
030 est in adapting them to specialized downstream applications. Compared to large language models
031 (LLMs), **MLLM** fine-tuning is not merely beneficial, but often necessary, as visual data presents
032 distinct challenges compared to text. Visual inputs are exceptionally high-dimensional, and many
033 specialized domains are poorly represented in the data used for pre-training. Consequently, out-of-
034 the-box **MLLMs** can struggle in critical applications, whether it is a robot not able to generalize to
035 unseen rooms (Shi et al., 2025), a web agent misinterpreting novel screenshot layouts (Xie et al.,
036 2024), or a biological application unable to identify specific cell types (Burgess et al., 2025).

037 However, the prevailing wisdom suggests that fine-tuning **MLLMs** is risky due to catastrophic for-
038 getting, a phenomenon in which specialization on a new task severely degrades a model’s general
039 capabilities (Zhai et al., 2024; Shuttleworth et al., 2024). To address this, previous work has pro-
040 posed a suite of complex solutions, ranging from sophisticated regularization schemes and param-
041 eter isolation techniques to intricate methods (Wang et al., 2023; Shuttleworth et al., 2024; Chen
042 et al., 2023; Li et al., 2025). These approaches often introduce significant architectural or training
043 overhead, reinforcing the notion that preserving general **MLLM** knowledge is an inherently difficult
044 problem (McCloskey & Cohen, 1989; Andreassen et al., 2021).

045 Surprisingly, our systematic study reveals that for **MLLMs**, catastrophic forgetting is largely not
046 a problem. We fine-tune state-of-the-art **MLLMs**, Qwen2.5-VL-3B (Bai et al., 2025), on the
047 ImageNet image classification task and evaluate them on a comprehensive 2x2 matrix, testing per-
048 formance on both in-distribution (ID) and out-of-distribution (OOD) image and text inputs (§2). Our
049 central finding is that with a simple and proper fine-tuning recipe, such as using a low learning rate
050 or employing parameter-efficient fine-tuning, **MLLMs** maintain their general-purpose performance,
051 especially when handling OOD visual inputs (§3.1, §3.2). We verify that this conclusion holds
052 across **MLLM** architectures, including LLaVA1.5-7B (Liu et al., 2023) and Qwen2.5-VL-7B,
053 as well as in extremely OOD fine-tuning domains, such as surgery and microscopy, challenging the
idea that a trade-off between specialization and generalization is inevitable (§3.3).

054
055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070

	ID Image	OOD Image
ID Text	 <p>What is the class of this image? A. bee eater B. hummingbird C. jacamar D. whiptail</p> <p>ImageNet V2 ImageNet-val</p>	 <p>What is the class of this image? A. passion flower B. thorn apple C. foxglove D. globe flower</p> <p>Flowers 102 Stanford Cars Caltech 101</p>
OOD Text	 <p>In which habitats is this object predominantly found? A. Low-altitude woodlands and forest edges B. Polar and subpolar zones C. Temperate forests and grasslands D. Whiptail</p> <p>ImageWikiQA</p>	 <p>VMCBench MMMU</p> <p>What is NOT exhibited in the painting? A. hierarchical scale B. graphic representation of horror and despair C. a sense of immediacy and drama D. use of sharply contrasting light and shade</p>

071 Figure 1: **Evaluation matrix.** A 2×2 design crossing *text* and *image*. In this work, for both text and
072 images, we define in-distribution (ID) data as samples drawn from the same probability distribution
073 as the training set. Conversely, out-of-distribution (OOD) data originates from a distribution not
074 encountered during training; during evaluation, we report average accuracy within each quadrant.
075 This setup allows us to systematically evaluate a comprehensive range of training and evaluation
076 scenarios. Further details on the datasets are provided in Appendix B.1.

077 However, our investigation revealed one specific and important failure mode (§4.1): forgetting occurs
078 on tasks involving ID images paired with OOD text (e.g., the same ImageNet image but with
079 different questions about the objects than classification). We determine that this scenario reduces the
080 problem to a uni-modal language task; since the images are familiar, the model’s behavior is dictated
081 by its language component. Here, the model overfits to the linguistic patterns of the training prompts
082 and fails to follow new instructions at inference time, which we call task-specific overfitting (§4.2).
083 We demonstrate that this issue can be resolved with a simple data-hybrid training strategy, which
084 involves mixing a small amount of general-purpose data with the task-specific fine-tuning dataset to
085 prevent this narrow overfitting (§4.3).

086 Armed with this complete understanding of **MLLM** fine-tuning, we extend our findings from single
087 fine-tuning to the challenging continual learning setting (Luo et al., 2025; Chen et al., 2024b).
088 In the newly created continual learning benchmark, which requires the **MLLM** to learn five challenging
089 remote sensing, medical, autonomous driving, science, and finance knowledge, we show that our
090 straightforward approach allows **MLLMs** to sequentially learn new tasks while preserving
091 prior knowledge (§5), outperforming all complex methods that rely on mechanisms like data
092 replay buffers (Zhao et al., 2025). This result underscores that the intrinsic capacity of **MLLMs** for
093 continual learning is much greater than previously understood.

094 **We believe our primary contribution is to reframe the community’s understanding of MLLM adap-**
095 **tation.** We demonstrate that the perceived threat of catastrophic forgetting has been overstated and
096 that effective, robust fine-tuning can be achieved with a remarkably simple recipe. We hope these
097 findings encourage practitioners to move beyond unnecessarily complex solutions and adopt this
098 parsimonious approach to unlock the full potential of **MLLMs** in diverse real-world applications.

100 2 MLLM FINE-TUNING: EVALUATION PROTOCOLS AND TRAINING RECIPES

101 This section specifies *how* we evaluate and *how* we fine-tune **multimodal large language models**
102 (**MLLMs**). We first define a controlled protocol built around a 2×2 distribution shift matrix, then
103 describe the models, training setup, and prompting templates used throughout.

104 2.1 EVALUATION PROTOCOLS

105 **Fine-tuning task and dataset.** We establish a consistent starting point by fine-tuning a multiple-
106 choice visual question answering task constructed from ImageNet, which we call **ImageNet-VQA**.

108 For each ImageNet image, we pose a single question asking for its class label with four options
109 (A–D): one ground-truth label and three distractors. To increase the difficulty of the fine-tuning
110 task, we employ CLIP (Radford et al., 2021) to select the most challenging distractors, with the
111 methodology detailed in Appendix B.1.

112 We choose ImageNet because it provides (i) large-scale, diverse, natural images with standard-
113 ized labels; (ii) a clean mapping to unambiguous multiple-choice questions; and (iii) a familiar
114 in-distribution (ID) reference point for studying shifts in either text or image domains.
115

116 **Axes of variation: text and image.** Our evaluation isolates two sources of distribution shift: *Text*
117 (the question form) and *Image* (the visual domain). ID text is the same classification question format
118 used for fine-tuning; OOD text uses question styles that require different reasoning skills or exter-
119 nal knowledge. ID images are natural photographs similar to ImageNet; OOD images come from
120 different object sets or visual domains (e.g., flowers or stylized drawings).

121 **The 2×2 evaluation matrix.** Crossing the two axes yields four standardized scenarios (Figure 1):

- 122 • **ID Text + ID Image (ID^T – ID^I)**: in-distribution questions on in-distribution images. Datasets:
123 ImageNet (Deng et al., 2009) (validation split) and ImageNetV2 (Recht et al., 2019).
- 124 • **ID Text + OOD Image (ID^T – OOD^I)**: in-distribution questions on out-of-distribution images.
125 Datasets: Flowers102 (Nilsback & Zisserman, 2008), Caltech101 (Fei-Fei et al., 2004), Stanford
126 Cars (Krause et al., 2013).
- 127 • **OOD Text + ID Image (OOD^T – ID^I)**: novel questions on in-distribution images. Dataset:
128 ImageWikiQA (Zhang et al., 2024).
- 129 • **OOD Text + OOD Image (OOD^T – OOD^I)**: novel questions on out-of-distribution images.
130 Datasets: MMMU (Yue et al., 2024), VMCBench (Zhang et al., 2025).

131 Unless otherwise noted, we report the accuracy averaged within each quadrant for clarity.
132

133 2.2 TRAINING RECIPES 134

135 **Base models.** We study two widely used **MLLM** families, Qwen2.5–VL (Bai et al., 2025) and
136 LLaVA (Liu et al., 2023). Our main ablations in §3 and §4 use Qwen2.5–VL–3B; we additionally
137 validate our findings on Qwen2.5–VL–7B and LLaVA–1.5–7B. For comparisons on the MLLM-
138 CL benchmark in §5, we adopt LLaVA–1.5–7B to align with previous work (Zhao et al., 2025).

139 **Codebase and hyperparameters.** We train with LLaMA-Factory (Zheng et al., 2024). Unless
140 specified, we use a batch size of 16 and ablate the learning rate of $\{1e-5, 1e-6\}$. Training runs
141 for one epoch on ImageNet-VQA (approximately 80,000 steps). We compare different trainable
142 parameters and keep other settings fixed for fair comparison; full configurations are listed in Ap-
143 pendix C. **Since LLM Backbone Fine-tuning is redundant and not commonly used, we donate**
144 **the recipe that unfreezing all LLM backbone parameters while freezing all the vision encoder and**
145 **project parameters as Full Fine-tuning.**

146 **Prompts and templates.** We use the system templates provided by LLaMA-Factory for
147 Qwen2.5–VL and LLaVA. All evaluations in §3 and §4 follow the multiple-choice format. To
148 avoid formatting confounding, the prompts explicitly instruct the model to output a single option
149 letter (A–D). Illustrative prompt templates are included in Appendix D.1.
150

151 3 FINE-TUNING WITHOUT FORGETTING: A SIMPLE RECIPE WITHOUT 152 PERFORMANCE TRADE-OFF 153

154 Can a **multimodal large language model (MLLM)** be specialized to a new task *without* erasing its
155 general capabilities? Using the 2×2 evaluation matrix (§2), we vary the trainable components (LLM
156 backbone, vision encoder, projector), optimization method (full fine-tuning vs. LoRA), and learning
157 rate.
158

159 Three consistent findings emerge: (I) with simple regularization (small learning rate or LoRA),
160 forgetting on OOD images is *nearly absent* as ID accuracy increases; (II) avoiding forgetting does
161 *not* reduce target-task accuracy; and (III) these patterns hold across model sizes/families, rare visual
domains, and low-data regimes.

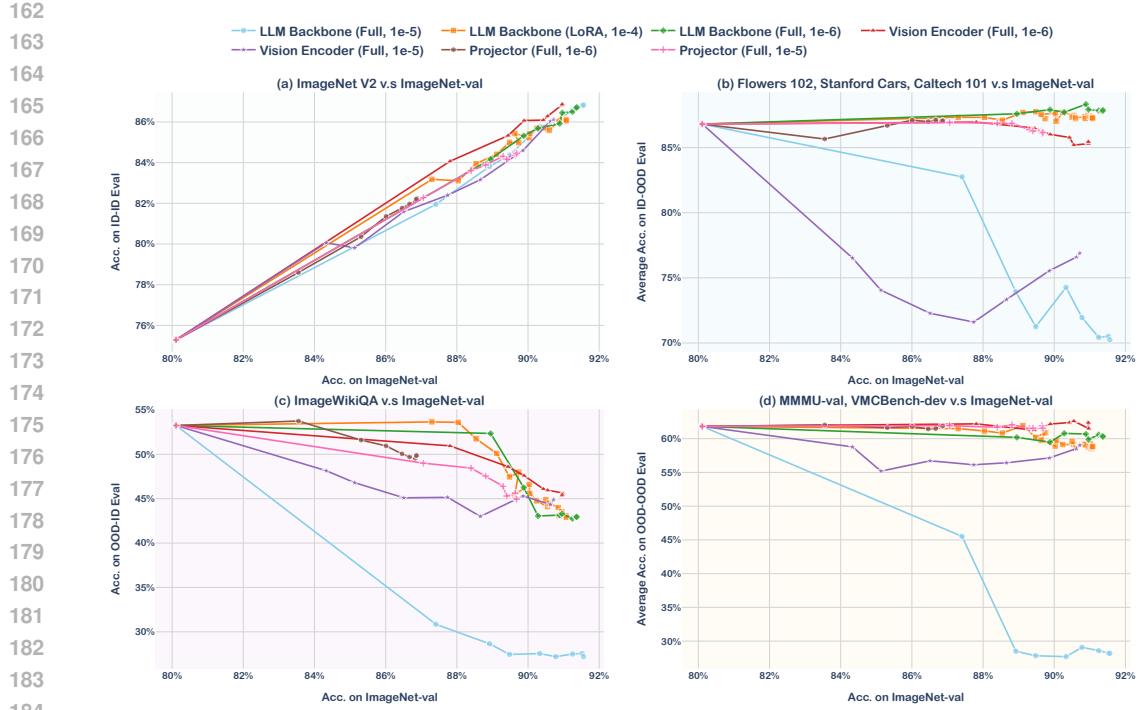


Figure 2: **Single-task fine-tuning across the evaluation matrix.** Each curve traces checkpoints during fine-tuning: x-axis = ID accuracy on ImageNet validation (the fine-tuned task), y-axis = accuracy on an ID/OOD evaluation. Layout and colors follow Figure 1. Legends show **trainable part (method, learning rate)**. Performance is largely maintained in **ID^T–OOD^I** and **OOD^T–OOD^I** with simplest regularization on parameter update, with a notable drop only in **OOD^T–ID^I**. Full hyperparameters are in Appendix C.1.

3.1 FINDING I: SIMPLE REGULARIZATION PREVENTS (NEARLY ALL) FORGETTING

Research question. Catastrophic forgetting is often attributed to architectural limits: specializing on a new task is thought to overwrite broad, pre-trained knowledge. If that were the case, the gains on the ID data should come with the losses on the OOD data.

Results. In Figure 2, high-learning-rate full fine-tuning (1e-5) increases ID accuracy but substantially degrades OOD performance, consistent with catastrophic forgetting: relative to zero-shot, **LLM Backbone, Full, 1e-5** yields -16.56 pp on **OOD^T–ID^I** and -33.64 pp on **OOD^T–OOD^I** (Table 1). In contrast, conservative settings (small learning rate or LoRA) keep the OOD accuracy essentially stable as the ID accuracy increases. Restricting the magnitude and scope of parameter updates eliminates these drops: **LLM Backbone, Full, 1e-6** changes are $+1.06$ pp (**OOD^T–ID^I**) and -1.51 pp (**OOD^T–OOD^I**); **LLM Backbone, LoRA, 1e-4** changes are $+0.46$ pp and -2.97 pp, respectively.

Takeaway I: Forgetting is *not* inevitable; it arises from over-optimization. Simple regularization (small learning rate or parameter-efficient training) preserves capabilities.

3.2 FINDING II: NO TRADE-OFF BETWEEN SPECIALIZATION AND PRESERVATION

Research question. Prior reports suggest a performance gap between full fine-tuning and LoRA on the target task. If regularization preserves OOD performance, does it *cost* ID accuracy?

Results. Table 1 shows that the regularized settings match the aggressive baseline on the ID task while avoiding OOD forgetting. Validation accuracy differences relative to **LLM Backbone, Full, 1e-5** are ≤ 0.6 pp for **LLM Backbone, Full, 1e-6** (-0.19 pp), **LLM Backbone, LoRA, 1e-4** (-0.48 pp), and **Vision Encoder, Full, 1e-6** (-0.60 pp). Projector-only fine-tuning is the sole exception (e.g., -4.70 pp at 1e-6) and is therefore not recommended when target-task accuracy is critical.

Trainable Part	Settings	Final Acc.	Δ vs. zero-shot	
		Validation (%)	$\text{OOD}^T\text{-ID}^I$ (pp)	$\text{OOD}^T\text{-OOD}^I$ (pp)
LLM Backbone	Full, 1e-5	91.56	-16.56	-33.64
LLM Backbone	LORA, 1e-4	91.08 (-0.48)	0.46	-2.97
LLM Backbone	Full, 1e-6	91.37 (-0.19)	1.06	-1.51
Vision Encoder	Full, 1e-6	90.96 (-0.60)	-1.36	0.49
Vision Encoder	Full, 1e-5	91.08 (-0.48)	-9.90	-2.76
Projector	Full, 1e-6	86.86 (-4.70)	0.26	0.05
Projector	Full, 1e-5	89.68 (-1.88)	-0.64	-0.26

Table 1: **ID accuracy and robustness deltas across recipes.** “Final Acc” is ImageNet-VQA validation accuracy; in parentheses we show the difference to **LLM Backbone, Full, 1e-5**. “ Δ vs. zero-shot” reports percentage-point change relative to the pre-trained model on $\text{OOD}^T\text{-ID}^I$ and $\text{OOD}^T\text{-OOD}^I$. To enhance visual clarity, we use **red** to highlight performance degradations $>3\text{pp}$ and **blue** for changes within a $\pm 3\text{pp}$ margin. Rows corresponding to settings where all results fall within this margin are shaded **gray**. This suggests that most of regularization strategies mitigate catastrophic forgetting without compromising the model’s learning capacity.

(a) Model size and family.				
Model Version	Validation (%)	ImageNetV2 (%)	ID–OOD (%)	OOD–OOD (%)
Qwen2.5-VL-3B	80.11→91.37	75.29→86.72	86.80→87.87	61.82→60.31
Qwen2.5-VL-7B	83.20→92.66	78.61→88.05	90.35→91.24	62.57→62.62
LLaVA-7B	65.53→91.43	61.55→86.76	66.44→70.05	41.45→37.73

(b) Rare domains.				
Dataset	Validation (%)	OOD–OOD (%)	(c) Dataset size.	
Dataset	Validation (%)	OOD–OOD (%)	Dataset fraction	Validation (%)
ImageNet	80.11→89.88	61.82→59.48	100%	91.42 91.60
BSCCM	18.15→84.34	61.82→61.19	25%	90.18 89.08
PitVis	25.61→51.33	61.82→61.56	2.5%	86.99 87.46
			0.25%	82.03 81.82

Table 2: **Generalization of the recipe.** The default setting referenced in §3.2 is shaded in **gray**. The results show that all findings in §3.1 are consistent across: (a) different model sizes and families; (b) rare domains including surgery and microscopy; (c) different fine-tuning datasets size; Full training details are in Appendix C.2.

Takeaway 2: Specialization and preservation are *not* at odds: Under regularized fine-tuning, ID and OOD performance do not trade off.

3.3 FINDING III: CONSISTENCY ACROSS MODELS, DOMAINS, AND DATA REGIMES

Research question. If the recipe is principled, it should transfer across architectures, uncommon visual domains, and data-scarce settings.

Results. Models. The trends persist across sizes and families (Table 2a): Qwen2.5-VL-3B improves ImageNet validation $80.11 \rightarrow 91.37$ with $\text{OOD}^T\text{-OOD}^I$ $61.82 \rightarrow 60.31$ (-1.51pp); Qwen2.5-VL-7B improves $83.20 \rightarrow 92.66$ with $\text{OOD}^T\text{-OOD}^I$ $+0.05\text{pp}$; LLaVA-1.5-7B improves $65.53 \rightarrow 91.43$ with a modest $\text{OOD}^T\text{-OOD}^I$ drop (-3.72pp).

Rare domains. The same recipe holds for microscopy (BSCCM (Pinkard et al., 2024)) and surgical (PitVis (Das et al., 2025)) data (Table 2b), keeping $\text{OOD}^T\text{-OOD}^I$ within $\leq 2.5\text{pp}$ while yielding large ID gains ($+66\text{pp}$ on BSCCM, $+26\text{pp}$ on PitVis).

Data size. Even at 0.25% of the data, a small learning rate (1e-6) remains competitive in the ID task (82.03 vs. 81.82 at 1e-5; Table 2c).

Takeaway 3: These findings generalize across architectures, domains, and data regimes, implying that forgetting in **MLLM** fine-tuning is generally not a concern.

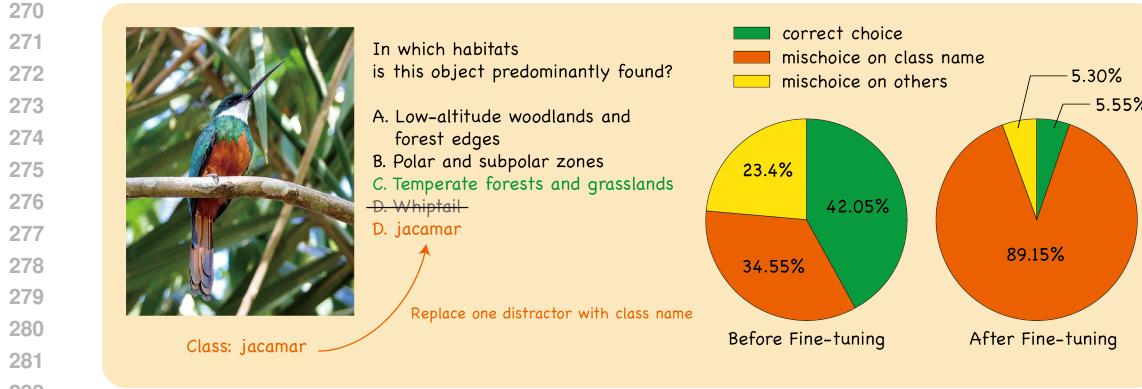


Figure 3: **ImageWikiQA with class-label distractors.** *Left:* an example transformation where one distractor is replaced by the correct class name. *Right:* accuracy with/without a class-name distractor, before fine-tuning and after fine-tuning, using **LLM Backbone, Full, 1e-6**. The substantial decrease in accuracy and the concurrent increase in “mischoice on class name” after fine-tuning indicate that the model ceases to follow prompt instructions, instead defaulting to outputting the choice with class label directly. Therefore, the primary issue is *task-specific overfitting* rather than catastrophic forgetting.

4 OOD TEXT MEETS ID IMAGES: DIAGNOSIS AND SIMPLE REMEDY

Our 2×2 evaluation reveals a single weak spot: $\text{OOD}^T\text{-ID}^I$ (novel text over familiar images), exemplified by ImageWikiQA. In contrast to $\text{ID}^T\text{-OOD}^I$ and $\text{OOD}^T\text{-OOD}^I$, where regularization preserves performance, Figure 2c shows a clear drop on OOD text with ID images. We (i) diagnose this failure as *task-specific overfitting* in the ID image distribution and (ii) demonstrate that a simple *data-hybrid* recipe prevents it with minimal impact on the target task.

4.1 FINDING IV: FORGETTING APPEARS ONLY WITH OOD TEXT OVER ID IMAGES

Research question. In $\text{OOD}^T\text{-ID}^I$, the image distribution matches fine-tuning (ID), but the text distribution shifts. The test set, ImageWikiQA (Zhang et al., 2024), asks the model to link an ImageNet image to external knowledge (e.g., the habitat of a species or the use of an artifact) rather than to perform the ImageNet classification task. This setup closely parallels standard LLM fine-tuning, where inputs remain in-domain while the instruction distribution changes. Prior work on LLMs has shown that single-task fine-tuning can impair other capabilities and encourage instruction-ignoring (Luo et al., 2025; Ung et al., 2024; Lyu et al., 2024).

Results. Even with regularized fine-tuning (e.g., small learning rates or LoRA), ImageWikiQA performance drops relative to zero-shot (Figure 2c). For example, the **LLM Backbone, Full, 1e-6** configuration falls from 53.35% to 42.95% (-10.40pp) after fine-tuning on ImageNet. This contrasts sharply with $\text{ID}^T\text{-OOD}^I$ and $\text{OOD}^T\text{-OOD}^I$, where performance remains stable under the same settings.

Takeaway 4: The sole exception in our study is the ID-image/OOD-text setting, where forgetting persists and is not remedied by standard regularized fine-tuning, mirroring findings from LLM fine-tuning.

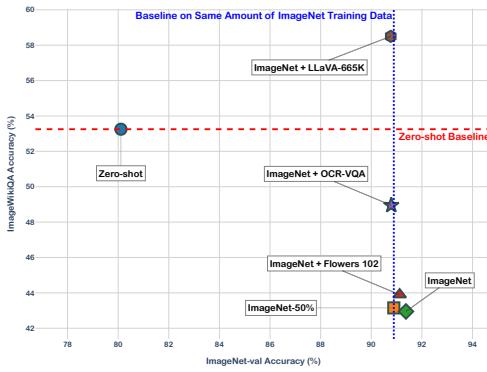
4.2 FINDING V: $\text{OOD}^T\text{-ID}^I$ FORGETTING ARISES FROM TASK-SPECIFIC OVERFITTING

Research question. We hypothesize that the model becomes over-attuned to the “classify-this-image” template when trained on ID images. To test this, we construct **ImageWikiQA with class-label distractors** by replacing one standard distractor with the correct class label (Figure 3, left). If the model has memorized the task, it should over-select the class label rather than the correct answer.

Results. Using the **LLM Backbone, Full, 1e-6** model, we observe severe *task-specific overfitting*: before fine-tuning, accuracy drops moderately when the class-name distractor is present (53.25% \rightarrow

324
325

(a) Mixing different datasets (50%).

326
327
328
329
330
331
332
333
334
335
336
337
338

(b) Varying the mixing ratio (LLaVA-665K).

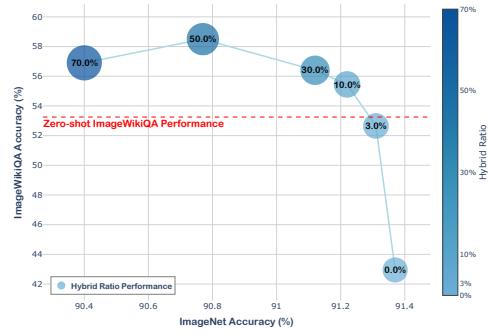


Figure 4: **Ablations for data-hybrid training.** (a) Mixing ImageNet-VQA with Flowers102, OCR-VQA, or LLaVA-665K (each at 50% of training instances). (b) Varying the LLaVA-665K mix from 0% to 70%; larger, darker markers denote higher ratios. Augmenting the training data with diverse textual inputs helps to alleviate *task-specific overfitting*. Consequently, this data-hybrid method improves model robustness in the $\text{OOD}^T\text{-ID}^I$ setting with minimal trade-offs for ID performance. Training details are in Appendix C.3.

42.05%, -11.2 pp); *after* fine-tuning, the drop is drastic ($42.95\% \rightarrow 5.55\%$, -37.4 pp) (Figure 3, right). The much larger change after fine-tuning indicates a learned bias to “pick the class label,”, that is, prompt-ignoring rather than knowledge deletion.

Takeaway 5: Forgetting in the ID-image/OOD-text case stems from *task-specific overfitting*: the model memorizes the image-specific classification template during fine-tuning and ignores the prompt.

353
354 4.3 FINDING VI: DATA-HYBRID TRAINING PREVENTS TASK OVERFITTING
355
356 **Research question.** If overfitting arises from repeatedly pairing ID images with a single classification template, mixing in *diverse* tasks should force the model to attend to the prompt and avoid the shortcut. We therefore ablate both **dataset type** and **mixing ratio**.

359 **Results.** *Dataset type (50% mix).* Figure 4a compares mixing ImageNet-VQA with: (i) Flowers102 (ID-style text on OOD images), (ii) OCR-VQA (OOD text), and (iii) LLaVA-665K (broad OOD instructions). Hybrid training consistently improves $\text{OOD}^T\text{-ID}^I$ while keeping ImageNet-VQA strong. Flowers102 yields only marginal gains on ImageWikiQA (another classification-style dataset, hence weak against task overfitting). OCR-VQA helps more by requiring text-based reasoning. LLaVA-665K performs best, likely due to its breadth of instructions and reasoning styles.

365 *Mixing ratio (with LLaVA-665K).* Figure 4b shows that increasing the proportion of LLaVA-665K to 50% keeps ImageNet-VQA within ~ 1 pp of the pure-ImageNet condition while markedly improving ImageWikiQA; at 70%, we see no further $\text{OOD}^T\text{-ID}^I$ gains. This suggests that the effect is not just “more data,” but specifically *task diversity* mitigating overfitting.

370 Finally, the effectiveness of co-training with OCR-VQA and LLaVA-665K indicates that, although 371 overfitting manifests on ID images, the remedy does not require additional ID images. Greater task 372 diversity alone is sufficient to counteract the bias, regardless of the image distribution. **In addition,** 373 **we have also shown that the synthetic data (LLaVA-665K) is effective, which furthermore provides** 374 **a positive result on the robustness of hybrid training.**

Takeaway 6: Data-hybrid fine-tuning—mixing diverse instruction data (without requiring ID images)—preserves ID-task accuracy while overcoming ID-image/OOD-text forgetting.

Method	RS (%)		Med (%)		AD (%)		Sci (%)		Fin (%)	
	Last	Average								
Zero-shot	32.29	-	28.28	-	15.59	-	35.55	-	62.56	-
<i>w/ replay buffer</i>										
LoRA	29.57	<u>80.87</u>	29.19	58.60	7.09	38.95	19.55	36.41	63.60	36.78
MoELoRA	40.23	80.00	23.58	56.91	5.19	34.69	18.35	31.70	74.89	31.36
O-LoRA	76.21	80.13	51.34	70.23	36.50	61.35	42.64	53.34	90.20	59.38
L2P	75.21	80.09	38.50	68.64	32.31	54.79	41.05	48.68	88.05	55.02
ModalPrompt	64.77	80.11	38.60	60.99	20.61	50.67	29.98	41.97	88.22	48.44
HiDe-LLaVA	75.36	81.51	39.23	62.37	37.17	49.37	45.02	50.61	81.89	55.73
MR-LoRA	79.87	80.82	62.71	72.19	<u>51.89</u>	<u>65.41</u>	52.48	62.52	89.69	67.31
IncLoRA (Ours)	77.43	78.30	<u>62.57</u>	71.93	52.00	65.38	52.48	62.12	<u>90.41</u>	66.98
SeqFull (Ours)	78.94	75.62	62.45	72.16	51.50	65.77	52.08	62.32	91.21	67.24
<i>w/o replay buffer</i>										
LoRA	26.75	<u>80.72</u>	25.76	59.68	0.79	40.51	18.69	18.64	70.44	28.49
MoELoRA	21.42	80.05	25.29	57.26	0.79	37.03	17.01	19.65	60.34	24.97
O-LoRA	62.68	80.22	35.17	67.56	16.93	<u>51.51</u>	34.44	44.28	92.16	48.28
L2P	63.82	80.02	34.63	68.86	22.96	51.57	38.58	45.12	92.98	50.59
ModalPrompt	65.99	80.11	37.35	59.66	23.27	46.86	37.61	42.97	87.60	50.36
HiDe-LLaVA	41.17	80.91	30.33	65.47	18.73	39.78	37.08	32.92	<u>92.21</u>	43.90
IncLoRA (Ours)	<u>77.20</u>	77.59	<u>58.97</u>	71.59	<u>51.43</u>	<u>64.40</u>	<u>47.44</u>	<u>60.22</u>	90.24	<u>65.06</u>
SeqFull (Ours)	79.10	77.06	61.22	72.75	52.36	66.09	50.52	62.49	91.29	67.44

Table 3: **Continual learning on the MLLM-CL benchmark.** We highlight best and second best separately for *with* and *without* replay. Our simple methods (IncLoRA, SeqFull) are competitive with specialized approaches under replay, and dominate most columns without replay.

5 FROM SINGLE TO MULTIPLE: SIMPLE STRATEGIES RIVAL SOTA

Our single-task study shows that catastrophic forgetting can be substantially reduced with regularization (§3) and data hybrid training (§4). The natural question is whether these observations carry over from one task to a sequence of tasks. We therefore turn to *continual learning*, where a model learns tasks one after another while preserving performance on earlier tasks. Perhaps unexpectedly, we find that very simple updates, either LoRA or a small learning rate, match or outperform prior methods purpose-built for continual learning, both *with* and *without* a replay buffer.

5.1 BENCHMARK AND EVALUATION

Benchmark. We use the **MLLM** continual learning benchmark introduced by MLLM-CL (Zhao et al., 2025), spanning five domains in a fixed order: **Remote Sensing** → **Medicine** → **Autonomous Driving** → **Science** → **Finance**. See §B.3 for more details.

Evaluation. Continual learning reframes forgetting from “does fine-tuning erase zero-shot skills?” to “does learning the next task erase the previous one?”. We therefore report two standard metrics: *Last* (performance on each task after training on the full sequence) and *Average* (mean performance across tasks at the time each task is learned). Details appear in §D.4.

Experimental setup. For comparability, we follow the MLLM-CL recipe exactly (optimizer, prompts, and models), adopt their evaluation protocol, and use the same dataset splits. The zero-shot row in Table 3 provides the pre-training baseline before any fine-tuning.

5.2 FINDING VII: SIMPLE STRATEGIES COMPETE WITH SOTA IN CONTINUAL LEARNING

Method. We evaluate two simple continual learning strategies: incremental LoRA (**IncLoRA**) and sequential full fine-tuning (**SeqFull**). For **IncLoRA**, we train a new LoRA *adapter* for each task and, after training, merge the adapter weights into the base model, which then initializes the next task. **SeqFull** simply fine-tunes all model parameters for each task in sequence, without additional mechanisms.

We refer our **IncLoRA** and **SeqFull** as *simple* because all the other method shown in Table 3 either use a router to select an appropriate LoRA instead of merging, or add additional regularization during fine-tuning the new LoRA. Our framework is essentially a simplified version of them.

432 **Results.** With a replay buffer (a bounded memory that retains a small sample of past tasks’
433 examples and replays them alongside the current task’s data to reduce catastrophic forgetting),
434 many prior methods introduce sophisticated components to control forgetting, yet our simple
435 approaches achieve performance comparable to state-of-the-art techniques. For example, **SeqFull**
436 attains 78.94% on **RS** under the *Last* metric, closely matching **MR-LoRA** (79.87%) while outper-
437 forming it in **Fin**.

438 The gap widens in the more restrictive no-replay setting, which is important for privacy-sensitive
439 applications (e.g., medicine) where replay is infeasible. Except for the *Average* metric in the first task
440 (**RS**) and the *Last* metric on the final task (**Fin**)—both of which do not reflect forgetting—**IncLoRA**
441 and **SeqFull** outperform all competing methods in the remaining eight comparisons, establishing
442 new state-of-the-art results in most domains.

443 **Takeaway 7:** Simple update policies rival or exceed specialized continual-learning meth-
444 ods, work in privacy-sensitive no-replay settings, and avoid additional complexity.

447 6 RELATED WORK

449 **Vision language models.** Multimodal large language models (**MLLMs**) such as Flamingo (Alayrac
450 et al., 2022), LLaVA (Liu et al., 2023), and GPT-4V (Achiam et al., 2023) demonstrate strong
451 visual–linguistic understanding and reasoning (Xu et al., 2024). A typical **MLLM** couples a vision
452 encoder with a language backbone—through a projector or cross-attention module—is trained in
453 large image-text corpora and is subsequently adjusted to instruction (Liu et al., 2023). Recent work
454 has emphasized scaling, architectural refinements, and training strategies to improve zero-/few-shot
455 generalization Tong et al. (2024); Chen et al. (2024c); Bai et al. (2025). In this work, we study how
456 to adapt strong base **MLLMs** to diverse downstream tasks while preserving zero-shot performance,
457 a problem that is arguably more acute for **MLLMs** than for LLMs, yet comparatively underexplored.

458 **Catastrophic forgetting.** Catastrophic forgetting is the loss of previously acquired knowledge when
459 a model is trained on new tasks (Kemker et al., 2018; Chen & Liu, 2022; Goodfellow et al., 2013).
460 In LLMs, catastrophic forgetting has been extensively studied—empirically (Kalajdzievski, 2024;
461 Scialom et al., 2022), theoretically (Shuttleworth et al., 2024), methodologically (Chen et al., 2023;
462 Li et al., 2025), and from an evaluation point of view (Ung et al., 2024). In contrast, catastrophic
463 forgetting in **MLLMs** has received less attention (Zhai et al., 2024). Previous work always shows a
464 result of learning less and forgetting less, while we are presenting the phenomenon of learning the
465 same amount without forgetting.

466 **Continual learning.** Continual learning aims to acquire new capabilities without erasing prior
467 knowledge (Wang et al., 2024; Chen & Liu, 2022; Hadsell et al., 2020). It is critical in real-world
468 settings where data distributions and taxonomies evolve, centralized retraining may be impractical
469 due to cost or privacy, and preserving generalist abilities (e.g., zero-shot performance) is important
470 for safety and robustness. To mitigate forgetting, previous work explores replay, regularization, and
471 parameter isolation approaches, but these often add considerable compute, memory, and engineering
472 complexity (Zhao et al., 2025; Van de Ven & Tolias, 2019). Although continual learning for **MLLMs**
473 has begun to be explored (Chen et al., 2024a; Huang et al., 2024), we show that—with appropriate
474 training recipes—forgetting can be largely mitigated, yielding state-of-the-art results with simple
475 and compute-efficient methods.

476 7 CONCLUSION

478 By rethinking and re-evaluating the design space of multimodal adaptation, this paper reframes how
479 to fine-tune multimodal large language models. We find that concerns about catastrophic forgetting
480 are often overstated. In practice, a simple recipe—using small learning rates or parameter-efficient
481 updates—yields specialized models that remain strong generalists. Our analysis isolates a single
482 failure mode: overfitting to linguistic patterns rather than visual content. We address this with a
483 straightforward hybrid-data mix. On a challenging continual learning benchmark, this recipe per-
484 forms on par with or better than more complex alternatives, suggesting that vision language models
485 are more intrinsically robust than commonly assumed. We hope these results encourage simpler,
486 more transparent adaptation methods and provide a stable foundation for future work.

486 **ETHICS STATEMENT**
487

488 We, the authors of this work, confirm our adherence to the ICLR Code of Ethics. Our research is
489 primarily methodological in nature and does not raise significant ethical concerns regarding data
490 privacy, fairness, or potential misuse, as it does not involve sensitive datasets or direct real-world
491 applications involving individuals.

492
493 **REPRODUCIBILITY STATEMENT**
494

495 To ensure the full reproducibility of our work, we provide our code, adapted datasets, and detailed
496 hyperparameter specifications, which include all scripts to generate the necessary configuration files
497 and perform the training and evaluations presented in this paper.

498 **Code:** <https://anonymous.4open.science/r/VLM-Forgetting-C1CE/>.

500 **Datasets:** <https://huggingface.co/datasets/VLM-Forgetting/vlm-forgetting-datasets>.

501 **Hyperparameters:** Appendix C.

502
503 **REFERENCES**

504 Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Ale-
505 man, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical
506 report. *arXiv preprint arXiv:2303.08774*, 2023.

507 Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel
508 Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language
509 model for few-shot learning. In *NeurIPS*, 2022.

510 Anders Andreassen, Yasaman Bahri, Behnam Neyshabur, and Rebecca Roelofs. The evolution of
511 out-of-distribution robustness throughout fine-tuning. *arXiv preprint arXiv:2106.15831*, 2021.

512 Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang,
513 Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*,
514 2025.

515 James Burgess, Jeffrey J Nirschl, Laura Bravo-Sánchez, Alejandro Lozano, Sanket Rajan Gupte,
516 Jesus G Galaz-Montoya, Yuhui Zhang, Yuchang Su, Disha Bhowmik, Zachary Coman, et al. Mi-
517 crovqa: A multimodal reasoning benchmark for microscopy-based scientific research. In *CVPR*,
518 2025.

519 Cheng Chen, Junchen Zhu, Xu Luo, Heng T Shen, Jingkuan Song, and Lianli Gao. Coin: A bench-
520 mark of continual instruction tuning for multimodel large language models. In *NeurIPS*, 2024a.

521 Jie Chen, Zhipeng Chen, Jiapeng Wang, Kun Zhou, Yutao Zhu, Jinhao Jiang, Yingqian Min,
522 Wayne Xin Zhao, Zhicheng Dou, Jiaxin Mao, et al. Towards effective and efficient continual
523 pre-training of large language models. *arXiv preprint arXiv:2407.18743*, 2024b.

524 Wuyang Chen, Yanqi Zhou, Nan Du, Yanping Huang, James Laudon, Zhifeng Chen, and Claire Cui.
525 Lifelong language pretraining with distribution-specialized experts. In *ICML*, 2023.

526 Zhe Chen, Jiannan Wu, Wenhui Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong
527 Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning
528 for generic visual-linguistic tasks. In *CVPR*, pp. 24185–24198, 2024c.

529 Zhiyuan Chen and Bing Liu. Continual learning and catastrophic forgetting. In *Lifelong Machine*
530 *Learning*. Springer, 2022.

531 Adrito Das, Danyal Z Khan, Dimitrios Psychogios, Yitong Zhang, John G Hanrahan, Francisco
532 Vasconcelos, You Pang, Zhen Chen, Jinlin Wu, Xiaoyang Zou, et al. Pitvis-2023 challenge:
533 Workflow recognition in videos of endoscopic pituitary surgery. *Medical Image Analysis*, 2025.

540 Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale
541 hierarchical image database. In *CVPR*, 2009.

542

543 Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong,
544 Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating
545 large multi-modality models. In *MM*, 2024.

546

547 Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training ex-
548 amples: An incremental bayesian approach tested on 101 object categories. In *CVPR Workshops*,
549 2004.

550

551 Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An empiri-
552 cal investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint*
553 *arXiv:1312.6211*, 2013.

554

555 Haiyang Guo, Fanhu Zeng, Ziwei Xiang, Fei Zhu, Da-Han Wang, Xu-Yao Zhang, and Cheng-Lin
556 Liu. Hide-llava: Hierarchical decoupling for continual instruction tuning of multimodal large
557 language model. *arXiv preprint arXiv:2503.12941*, 2025.

558

559 Raia Hadsell, Dushyant Rao, Andrei A Rusu, and Razvan Pascanu. Embracing change: Continual
560 learning in deep neural networks. *Trends in cognitive sciences*, 2020.

561

562 Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang,
563 Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.

564

565 Wenke Huang, Jian Liang, Zekun Shi, Didi Zhu, Guancheng Wan, He Li, Bo Du, Dacheng Tao,
566 and Mang Ye. Learn from downstream and be yourself in multimodal large language model
567 fine-tuning. *arXiv preprint arXiv:2411.10928*, 2024.

568

569 Damjan Kalajdzievski. Scaling laws for forgetting when fine-tuning large language models. *arXiv*
570 *preprint arXiv:2401.05605*, 2024.

571

572 Ronald Kemker, Marc McClure, Angelina Abitino, Tyler Hayes, and Christopher Kanan. Measuring
573 catastrophic forgetting in neural networks. In *AAAI*, 2018.

574

575 Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained
576 categorization. In *ICCV Workshops*, 2013.

577

578 Xinlong Li, Weijieying Ren, Wei Qin, Lei Wang, Tianxiang Zhao, and Richang Hong. Analyzing
579 and reducing catastrophic forgetting in parameter efficient tuning. In *ICASSP*. IEEE, 2025.

580

581 Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*,
582 2023.

583

584 Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of
585 catastrophic forgetting in large language models during continual fine-tuning. *TASLP*, 2025.

586

587 Kaifeng Lyu, Haoyu Zhao, Xinran Gu, Dingli Yu, Anirudh Goyal, and Sanjeev Arora. Keeping llms
588 aligned after fine-tuning: The crucial role of prompt templates. In *NeurIPS*, 2024.

589

590 Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The
591 sequential learning problem. In *Psychology of learning and motivation*. Elsevier, 1989.

592

593 Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number
594 of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*,
595 2008.

596

597 Henry Pinkard, Cherry Liu, Fanice Nyatigo, Daniel A Fletcher, and Laura Waller. The berkeley
598 single cell computational microscopy (bsccm) dataset. *arXiv preprint arXiv:2402.06191*, 2024.

599

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal,
Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual
models from natural language supervision. In *ICML*, 2021.

594 Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers
595 generalize to imagenet? In *ICML*, 2019.
596

597 Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. Fine-tuned language models are
598 continual learners. *arXiv preprint arXiv:2205.12393*, 2022.
599

600 Lucy Xiaoyang Shi, Michael Robert Equi, Liyiming Ke, Karl Pertsch, Quan Vuong, James Tanner,
601 Anna Walling, Haohuan Wang, Niccolo Fusai, Adrian Li-Bell, et al. Hi robot: Open-ended
602 instruction following with hierarchical vision-language-action models. In *ICML*, 2025.
603

604 Reece Shuttleworth, Jacob Andreas, Antonio Torralba, and Pratyusha Sharma. Lora vs full fine-
605 tuning: An illusion of equivalence. *arXiv preprint arXiv:2410.21228*, 2024.
606

607 Peter Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Adithya Jairam Vedagiri IYER, Sai Charitha
608 Akula, Shusheng Yang, Jihan Yang, Manoj Middepogu, Ziteng Wang, et al. Cambrian-1: A fully
609 open, vision-centric exploration of multimodal llms. In *NeurIPS*, 2024.
610

611 Megan Ung, Alicia Sun, Samuel J Bell, Bhaktipriya Radharapu, Levent Sagun, and Adina Williams.
612 Chained tuning leads to biased forgetting. *arXiv preprint arXiv:2412.16469*, 2024.
613

614 Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint
arXiv:1904.07734*, 2019.
615

616 Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual
617 learning: Theory, method and application. *TPAMI*, 2024.
618

619 Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and
620 Xuanjing Huang. Orthogonal subspace learning for language model continual learning. *arXiv
preprint arXiv:2310.14152*, 2023.
621

622 Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vin-
623 cent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *CVPR*,
624 2022.
625

626 Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua,
627 Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents
628 for open-ended tasks in real computer environments. In *NeurIPS*, 2024.
629

630 Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. Llava-cot: Let
631 vision language models reason step-by-step. *arXiv preprint arXiv:2411.10440*, 2024.
632

633 Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens,
634 Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal
635 understanding and reasoning benchmark for expert agi. In *CVPR*, 2024.
636

637 Fanhu Zeng, Fei Zhu, Haiyang Guo, Xu-Yao Zhang, and Cheng-Lin Liu. Modalprompt: Dual-
638 modality guided prompt for continual learning of large multimodal models. *arXiv preprint
arXiv:2410.05849*, 2024.
639

640 Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. Investigating
641 the catastrophic forgetting in multimodal large language model fine-tuning. In *CPAL*, 2024.
642

643 Yuhui Zhang, Alyssa Unell, Xiaohan Wang, Dhruba Ghosh, Yuchang Su, Ludwig Schmidt, and
644 Serena Yeung-Levy. Why are visually-grounded language models bad at image classification? In
645 *NeurIPS*, 2024.
646

647 Yuhui Zhang, Yuchang Su, Yiming Liu, Xiaohan Wang, James Burgess, Elaine Sui, Chenyu Wang,
648 Josiah Aklilu, Alejandro Lozano, Anjiang Wei, et al. Automated generation of challenging
649 multiple-choice questions for vision language model evaluation. In *CVPR*, 2025.
650

651 Hongbo Zhao, Fei Zhu, Rundong Wang, Gaofeng Meng, and Zhaoxiang Zhang. Mllm-cl: Continual
652 learning for multimodal large language models. *arXiv preprint arXiv:2506.05453*, 2025.
653

654 Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyuan Luo, Zhangchi Feng, and
655 Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv
preprint arXiv:2403.13372*, 2024.
656

648 A STATEMENT ON THE USE OF LARGE LANGUAGE MODELS (LLMs)
649

650 In the preparation of this manuscript, the authors utilized a Large Language Model (LLM) as a
651 general-purpose writing assistance tool. The LLM’s role was strictly limited to improving the clarity,
652 grammar, and readability of the text. Specific tasks included rephrasing sentences for better flow,
653 correcting grammatical errors, and ensuring consistent terminology.
654

655 B DATASETS DETAILS
656

657 B.1 2x2 EVALUATION MATRIX DETAILS
658

659 **Classification Datasets (Deng et al., 2009; Nilsback & Zisserman, 2008; Fei-Fei et al., 2004; Krause et al., 2013).** For all classification datasets in the evaluation matrix, we follow the same protocol to turn them into multiple-choice questions. The question text are fixed to *What is the class of this image? Please answer with a single letter (A, B, C, or D).*, where the formatting instructions are concatenated to ensure the evaluation result will not be greatly influenced by output format of the model.

660 To increase the difficulty of the task and test the model’s fine-grained discrimination ability, distractors are strategically selected. We use CLIP (Radford et al., 2021) to identify the five incorrect class labels with the highest semantic similarity scores to the image. From this pool of five candidates, we randomly sample three to serve as distractors. This methodology ensures that incorrect options are semantically plausible, requiring the model to perform a more precise identification. By fine-tuning on ImageNet-VQA, the model is trained to perform a standard, in-distribution (ID) image classification task.

661 **ImageWikiQA (Zhang et al., 2024).** Since the ImageWikiQA dataset is already in a format of multiple-choice question, we directly use adapt it.

662 **MMMU and VMCBench (Yue et al., 2024; Zhang et al., 2025).** Since the MMMU and VM-CBench datasets are already in a format of multiple-choice question, we directly use adapt them. For all the numbers reported in this paper, we use the MMMU-val split for the evaluation.

663 B.2 RARE DATASETS DETAILS
664

665 **BSCCM.** We use the original BSCCM (Pinkard et al., 2024) dataset and follow the official guide at https://github.com/Waller-Lab/BSCCM/blob/main/Getting_started.ipynb to create a classification question-answering dataset. We collect images from all 23 available channels, including:

- 666 • Brightfield
- 667 • DF_50, DF_50_Bottom, DF_50_Right,
- 668 • DF_55,
- 669 • DF_60, DF_60_Bottom, DF_60_Right,
- 670 • DF_65,
- 671 • DF_70, DF_70_Bottom, DF_70_Right,
- 672 • DF_75,
- 673 • DF_80, DF_80_Bottom, DF_80_Right,
- 674 • DF_85,
- 675 • DF_90,
- 676 • DPC_Bottom, DPC_Left, DPC_Right, DPC_Top,
- 677 • LED119

678 There are 10 classes in total, and for each question we ask the model to choose from 6 possible choices. The 5 distractors are randomly sampled from all possible choices and we provide the list of classes as follows:

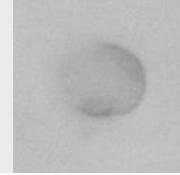
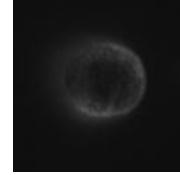
702 1. neutrophil 5. basophil 9. b_lymphocyte
 703 2. nk_lymphocyte 6. monocyte 10. t_lymphocyte
 704 3. eosinophil 7. plasma_cell
 705 4. lymphocyte 8. blast_cell

707
 708 To increase the type of questions, we provide multiple choices of prompt while all of them are
 709 sharing the same semantic meaning.

710
 711 • What type of white blood cell is shown in this *{channel_type}* microscopy image?
 712 • Based on the morphological features visible in this *{channel_type}* image, what is the cell type?
 713 • What is the most likely classification of this blood cell captured with *{channel_type}* illumina-
 714 tion?
 715 • Which white blood cell type does this *{channel_type}* image represent?
 716 • What type of immune cell is depicted in this *{channel_type}* microscopy image?
 717 • Looking at the cell morphology in this *{channel_type}* image, which cell type is this?
 718 • What is the identity of this cell captured using *{channel_type}* in LED array microscopy?

719 We provide the following samples in Table 4 from curated dataset. During training and inference, a
 720 prompt of "Please answer with a single letter (A, B, C, D, E or F)" is appended at the end to avoid
 721 the influence from model response formatting.

722 Table 4: VQA Dataset Curated from BSCCM
 723

724 Image	725 Question	726 Choices
726 	727 What is the identity of this cell captured us- 728 ing brightfield in LED array microscopy?	729 A. eosinophil 730 B. neutrophil 731 C. t_lymphocyte 732 D. plasma_cell 733 E. debris_or_artifact 734 F. unclassified_cell
731 	732 What type of white blood cell is shown 733 in this dark field (50 illumination) mi- 734 croscopy image?	735 A. plasma_cell 736 B. nk_lymphocyte 737 C. b_lymphocyte 738 D. neutrophil 739 E. unclassified_cell 740 F. debris_or_artifact
738 	739 Based on the morphological features visi- 740 ble in this differential phase contrast (left 741 illumination) image, what is the cell type?	742 A. basophil 743 B. unclassified_cell 744 C. debris_or_artifact 745 D. t_lymphocyte 746 E. blast_cell 747 F. lymphocyte

748 **PitVis.** We use the PitVis Challenge (Das et al., 2025) to create a classification dataset aiming to
 749 categorize the frame sampled from video according to the surgical instrument appeared. We fix the
 750 sample rate to be 1 out of every 6 frames. The total 21 instrument classes are as follows.

751 Fixed choices:

752 1. no_secondary_instrument
 753 2. out_of_patient
 754 3. no_visible_instrument/occluded_image_inside_patient

755 Other choices:

756 1. bipolar_forceps 7. irrigation_syringe 13. ring_curette
757 2. cottle 8. kerrisons 14. spatula_dissector
758 3. cup_forceps 9. micro_doppler 15. stealth_pointer
759 4. dural_scissors 10. nasal_cutting_forceps 16. suction
760 5. freer_elevator 11. pituitary_rongeurs 17. surgical_drill
761 6. haemostatic_foam 12. retractable_knife 18. tissue_glue
762
763

We still ask the model to choose from 6 possible choices. For every question, there will be 3 fixed choices to be *no_secondary_instrument*, *out_of_patient*, *no_visible_instrument* and we will randomly sample 2 or 3 distractors from all other classes (2 if the ground truth is not one of the 3 fixed classes).

We provide the following samples in Table 5 from curated dataset. During training and inference, a prompt of "Please answer with a single letter (A, B, C, D, E or F)" is appended at the end to avoid the influence from model response formatting.

Table 5: VQA Dataset Curated from BSCCM

Image	Question	Choices
	What is the major surgical instrument being used in this frame?	A. tissue_glue B. retractable_knife C. haemostatic_foam D. no_secondary_instrument E. out_of_patient F. no_visible_instrument/occluded_image_inside_patient
	What is the major surgical instrument being used in this frame?	A. plasma_cell B. nk_lymphocyte C. b_lymphocyte D. neutrophil E. unclassified_cell F. debris_or_artifact
	What is the major surgical instrument being used in this frame?	A. out_of_patient B. ring_curette C. no_visible_instrument/occluded_image_inside_patient D. freer_elevator E. micro_doppler F. no_secondary_instrument

B.3 MLLM-CL DETAILS

This sequential learning benchmark MLLM-CL contains:

- **RS**: Remote Sensing Data **RSVQA** (60k Training Data)
- **Med**: Medical Data **PathVQA** (23k Training Data)
- **AD**: Auto-Driving Data **DriveLM** (60k Training Data)
- **Sci**: Science Data **AI2D**, **SciVerse**, **MapQA**, **TQA** (33k Training Data)
- **Fin**: Financial Data **StockQA** (60k Training Data).

More details about the dataset can be found in MLLM-CL paper (Zhao et al., 2025). We adapt the number reported in original MLLM-CL paper, including LoRA Hu et al. (2022), MoELORA Chen et al. (2024a), O-LoRA Wang et al. (2023), L2P Wang et al. (2022), ModalPrompt Zeng et al. (2024), HiDe-LLaVA* Guo et al. (2025), MR-LoRA Zhao et al. (2025).

C TRAINING HYPER-PARAMETERS AND DETAILS

C.1 TRAINING HYPER-PARAMETERS FOR FIGURE 2

In this section, we align the table caption with Figure 2.

Config	Value
Optimizer	AdamW
Batch Size	32
Learning Rate Schedule	cosine decay
Warmup Ratio	0.1
Learning Rate	1×10^{-4}
Training Steps	40000
LoRA Rank	8
Freeze Vision Tower	True
Freeze Multi Modal Projector	True
Freeze Language Model	False

(a) LLM Backbone (LoRA, 1e-4)

Config	Value	Config	Value
Optimizer	AdamW	Optimizer	AdamW
Batch Size	16	Batch Size	16
Learning Rate Schedule	cosine decay	Learning Rate Schedule	cosine decay
Warmup Ratio	0.1	Warmup Ratio	0.1
Learning Rate	1×10^{-5}	Learning Rate	1×10^{-6}
Training Steps	80000	Training Steps	80000
Freeze Vision Tower	True	Freeze Vision Tower	True
Freeze Multi Modal Projector	True	Freeze Multi Modal Projector	True
Freeze Language Model	False	Freeze Language Model	False

(b) LLM Backbone (Full, 1e-5)

Config	Value	Config	Value
Optimizer	AdamW	Optimizer	AdamW
Batch Size	16	Batch Size	16
Learning Rate Schedule	cosine decay	Learning Rate Schedule	cosine decay
Warmup Ratio	0.1	Warmup Ratio	0.1
Learning Rate	1×10^{-5}	Learning Rate	1×10^{-6}
Training Steps	80000	Training Steps	80000
Freeze Vision Tower	False	Freeze Vision Tower	False
Freeze Multi Modal Projector	True	Freeze Multi Modal Projector	True
Freeze Language Model	True	Freeze Language Model	True

(d) Vision Encoder (Full, 1e-5)

Config	Value	Config	Value
Optimizer	AdamW	Optimizer	AdamW
Batch Size	16	Batch Size	16
Learning Rate Schedule	cosine decay	Learning Rate Schedule	cosine decay
Warmup Ratio	0.1	Warmup Ratio	0.1
Learning Rate	1×10^{-5}	Learning Rate	1×10^{-6}
Training Steps	80000	Training Steps	80000
Freeze Vision Tower	True	Freeze Vision Tower	True
Freeze Multi Modal Projector	False	Freeze Multi Modal Projector	False
Freeze Language Model	True	Freeze Language Model	True

(f) Projector (Full, 1e-5)

(g) Projector (Full, 1e-6)

864 **C.2 TRAINING HYPER-PARAMETERS FOR TABLE 2**
 865

866 In the ablation across different setting, we study the fine-tuning receipt of full fine-tuning LLM back-
 867 bone (learning rate 1e-6) since this is the most surprising result in our paper. Since LoRA fine-tuning
 868 or fine-tuning other parts (vision encoder or projector) is more regularized, doing validation study
 869 on the simplest fine-tuning LLM backbone is the most convincible choice.

Config	Value	Config	Value
Optimizer	AdamW	Optimizer	AdamW
Batch Size	16	Batch Size	16
Learning Rate Schedule	cosine decay	Learning Rate Schedule	cosine decay
Warmup Ratio	0.1	Warmup Ratio	0.1
Learning Rate	1×10^{-6}	Learning Rate	1×10^{-6}
Training Steps	80000	Training Steps	20000
Freeze Vision Tower	True	Freeze Vision Tower	True
Freeze Multi Modal Projector	True	Freeze Multi Modal Projector	True
Freeze Language Model	False	Freeze Language Model	False

881 (a) Configuration for ablation study across model size
 882 and model family, all the 3 models share the above
 883 hyper-parameters.

Config	Value	Config	Value
Optimizer	AdamW	Optimizer	AdamW
Batch Size	16	Batch Size	16
Learning Rate Schedule	linear	Learning Rate Schedule	linear
Warmup Ratio	0.1	Warmup Ratio	0.0025
Learning Rate	1×10^{-6}	Learning Rate	1×10^{-6}
Training Steps	2000	Training Steps	80000
Freeze Vision Tower	True	Freeze Vision Tower	True
Freeze Multi Modal Projector	True	Freeze Multi Modal Projector	True
Freeze Language Model	False	Freeze Language Model	False

895 (c) Ablation study across dataset size, 2000 training
 896 steps corresponding to 2.5% dataset, the warmup
 897 steps is $2000 \times 0.1 = 200$. This configuration produce
 898 the results of 0.25% and 2.5%.

500 (b) Configuration for ablation study across rare
 501 datasets, all the 3 datasets share the above hyper-
 502 parameters.

Config	Value	Config	Value
Optimizer	AdamW	Optimizer	AdamW
Batch Size	16	Batch Size	16
Learning Rate Schedule	linear	Learning Rate Schedule	linear
Warmup Ratio	0.1	Warmup Ratio	0.0025
Learning Rate	1×10^{-6}	Learning Rate	1×10^{-6}
Training Steps	2000	Training Steps	80000
Freeze Vision Tower	True	Freeze Vision Tower	True
Freeze Multi Modal Projector	True	Freeze Multi Modal Projector	True
Freeze Language Model	False	Freeze Language Model	False

595 (d) Ablation study across dataset size, 80000 training
 600 steps corresponding to the 100% dataset, the warmup
 601 steps is $80000 \times 0.0025 = 200$. This configuration produce
 602 the results of 25% and 100%.

690 **C.3 TRAINING HYPER-PARAMETERS FOR FIGURE 4**
 691

702 In this part, we still use full fine-tuning LLM backbone (learning rate 1e-6) as the default setting for
 703 the same reason with Appendix C.2. For hybridizing different datasets, we use a fixed hybridizing ratio
 704 of 0.5. The datasets will be oversampling if all the samples has been used at least once.

Config	Value
Optimizer	AdamW
Batch Size	16
Learning Rate Schedule	cosine decay
Warmup Ratio	0.1
Learning Rate	1×10^{-6}
Training Steps	80000
Freeze Vision Tower	True
Freeze Multi Modal Projector	True
Freeze Language Model	False

916 (a) Configuration for ablation study across hybridizing different datasets and different hybrid ratio, all experiments share the above hyper-parameters.

918 C.4 TRAINING HYPER-PARAMETERS FOR TABLE 3
919

920 We follow the configuration from MLLM-CL(Zhao et al., 2025) to achieve a fair comparison with
921 their results.
922

923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1888
1889
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2098
2099
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2198
2199
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2298
2299
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2398
2399
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2498
2499
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2598
2599
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2698
2699
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2798
2799
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2898
2899
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2969
2970
2971
2972
2973
2974
2975
2976
2977
2978
2979
2979
2980
2981
2982
2983
2984
2985
2986
2987
2988
2989
2989
2990
2991
2992
2993
2994
2995
2996
2997
2998
2998
2999
2999
3000
3001
3002
3003
3004
3005
3006
3007
3008
3009
3009
3010
3011
3012
3013
3014
3015
3016
3017
3018
3019
3019
3020
3021
3022
3023
3024
3025
3026
3027
3028
3029
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3039
3040
3041
3042
3043
3044
3045
3046
3047
30

972 **LLaVA-1.5.** We use the default `LLaMA-Factory` prompt, which is also the official prompt from
973 LLaVA-1.5 repository.
974

975 **System Prompt**

976 A chat between a curious user and an artificial intelligence assistant. The assistant gives
977 helpful, detailed, and polite answers to the user’s questions.
978 USER: *{User’s prompt}*
979 ASSISTANT: *{Model’s response}*
980

981

982 **D.2 EVALUATION OF 2x2 EVALUATION MATRIX**
983

984 **Result Matcher.** We use a `result_matcher.py` file to evaluate the answer accuracy of pre-
985 dictions. All the questions in this part are multiple-choice questions and the answer is a single letter.
986 All predictions are stored in a json file `f`, each entry has a `predict` key containing the model’s
987 output to the question and a `label` key containing a single letter as the ground truth. The logic is
988 as follows:

```
1 correct_predictions = 0
2 total_predictions = len(f)
3 for entry in f:
4     predict = str(entry['predict']).strip()
5     label = str(entry['label']).strip()
6
7     if ":" in predict:
8         predict = predict.split(":")[-1].strip()
9
10    predict = predict.upper()
11    label = label.upper()
12
13    if predict == label or predict.startswith(f"{{label}}."):
14        correct_predictions += 1
15
16 accuracy = correct_predictions / total_predictions
```

1003 Listing 1: Pseudo code snippet for `result_matcher.py`.

1004

1005 This above script is adapted for evaluations curated from **ImageNet**, **Flowers 102**, **Caltech 101**,
1006 **Stanford Cars**, **ImageWikiQA**.

1007

1008 **VLMEvalKit.** For evaluation of **MMMU** and **VMCBench**, we directly use the code in
1009 VLMEvalKit (Duan et al., 2024) to get the results.

1010

1011 **D.3 EVALUATION OF RARE DATASETS**

1012

1013 Since the questions we curated from **BSCCM** and **PitVis** are all multiple-choice questions, we use
1014 the same protocols as Appendix D.2, adapting the **Result Matcher** code in Listing 1.

1015

1016 **D.4 EVALUATION OF MLLM-CL**

1017

1018 **Last and Average.** *Last* is the accuracy of all seen tasks after learning the last task. *Average* is the
1019 average accuracy of each task during the training process, *i.e.*, $Average = \sum_{i=1}^t acc_i$, where t is the
1020 task that the model is learning, acc_i is the accuracy of the i -th previous learned task.

1021

1022 **Result Matching.** For turning the generation result, we directly adapt the script from MLLM-CL
1023 to ensure the fair comparison. The only change is in the `Sci` script. The original script use the
1024 image storage path to distinguish different kind of types of questions, we find that this is detecting
1025 the multiple-choice question with one single choice letter as the ground truth. Thus, we replace the
1026 judge condition of `image.split(' ') [-1].split('_') [0] == "AI2D" or image.split(' ') [-1].split('_') [0]`

```

1026     == "VQA" or image.split('/')[-1].split('_')[0] == "SciVerse" with len(gt)
1027     == 1.
1028
1029 Evaluation code snippet for evaluating RS and AD. All the namings follows Appendix D.2.
1030
1031 1 right = 0
1032 2 total = len(f)
1033 3 for entry in f:
1034 4     ground_truth = entry['label']
1035 5     if 'Unanswerable' in entry['predict'] :
1036 6         continue
1037 7
1038 8     pred: str = entry['predict'].lower()
1039 9     gt: str = ground_truth.lower()
1040 10
1041 11     score = 0
1042 12     if ' ' in gt:
1043 13         if gt in pred:
1044 14             right += 1
1045 15     else:
1046 16         gt = gt.replace('.', '')
1047 17         if ' ' in pred:
1048 18             if (' ' + gt) in pred or (gt + ' ') in pred or (gt + '.') in pred
1049 19             or (gt + ',') in pred:
1050 20                 right += 1
1051 21             else:
1052 22                 if gt in pred:
1053 23                     right += 1
1054 24
1055 25 accuracy = right / total

```

Listing 2: Pseudo code snippet for evaluating RS and AD.

1054 **Evaluation code snippet for evaluating Med.** All the namings follows Appendix D.2.

```

1056 1 right = 0
1057 2 total = len(f)
1058 3 for entry in f:
1059 4     ground_truth = entry['label'].lower()
1060 5     pred = entry['predict'].lower()
1061 6     if 'Unanswerable' in entry['predict'] :
1062 7         continue
1063 8
1064 9     if ground_truth in pred:
1065 10         right += 1
1066 11
1067 12 accuracy = right / total

```

Listing 3: Pseudo code snippet for evaluating Med.

1068 **Evaluation code snippet for evaluating Sci.** All the namings follows Appendix D.2, the prompt key containing the question description.

```

1071 1 right = 0
1072 2 total = len(f)
1073 3 for entry in f:
1074 4     ground_truth = entry['label'].strip()
1075 5     problem = entry['prompt']
1076 6
1077 7     pred: str = entry['predict'].strip().lower().replace('.', '').replace
1078 8         (' ', '').replace('neither', 'no')
1079 9     gt: str = ground_truth.strip().lower().replace('.', '').replace(' ', '')
1080 10         .replace('neither', 'no')

```

```

1080 10     if len(gt) == 1:
1081 11         if gt == pred:
1082 12             right += 1
1083 13     else:
1084 14         if 'Which states' in problem:
1085 15             gt_list = gt.split(',')
1086 16             len_gt = len(gt_list)
1087 17             pred_map_list = pred.split(',')
1088 18
1089 19             count = 0
1090 20             for gt in gt_list:
1091 21                 if gt in pred_map_list:
1092 22                     count += 1
1093 23             right += count / len_gt
1094 24         elif gt in pred or pred in gt:
1095 25             right += 1
1096 26
1097 27 accuracy = right / total

```

Listing 4: Pseudo code snippet for evaluating Sci.

1098 **Evaluation code snippet for evaluating Fin.** All the namings follows Appendix D.2.

```

1100 1 right = 0
1101 2 total = len(f)
1102 3 for entry in f:
1103 4     ground_truth = entry['label']
1104 5
1105 6     pred: str = entry['predict'].lower().replace(' ', '').replace('.', '')
1106 7     gt: str = ground_truth.lower()
1107 8     score = 0
1108 9     if gt == pred:
1109 10        right += 1
1110 11
1111 12 accuracy = right / total

```

Listing 5: Pseudo code snippet for evaluating Fin.

E FINE-TUNING ON PATH VQA

1116 Experiment for full fine-tuning Qwen-3B-Instruct for 80,000 steps on the PathVQA dataset,
1117 which is an open-ended pathology question answering dataset. Since the fine-tuning process is
1118 independent of the ImageNet dataset, we think the only reasonable evaluation is the OOD^T-OOD^I
1119 case, and the results are as follows:

	MMMU-dev (%)	MMMU-val (%)	VMCBench (%)
Before Fine-tuning	48.00%	48.44%	75.20%
After Fine-tuning	48.67%	45.00%	74.20%