How Foundation Skills Boost Continual Vision-Language Learning: A Skill-Evolution Perspective

Anonymous authors
Paper under double-blind review

Abstract

Continual learning can empower large vision-language models (VLMs) to acquire new knowledge in non-stationary environments without replaying historical downstream-task data, yet continual post-training often causes catastrophic forgetting. We argue that such forgetting also stems from the degradation of reusable foundation capabilities inherited from pretraining. This paper therefore revisits a fundamental yet underexplored problem: how can foundation skills in VLMs boost continual vision-language learning performance? We propose a Skill-Centric Adaptation model for Lifelong Evolution, termed *SCALE*, which reinterprets pretrained VLMs as a repertoire of reusable foundation skills. As continual downstream tasks come, downstream task adaptation follows supervised fine-tuning (SFT), while SCALE adopts skill-specific drift-aware gates to identify drifting foundation skills and task-prioritized co-evolution to maintain them. These selected skills can evolve with the model through verifier-guided behavioral supervision, without constraining current-task adaptation. This co-evolution strategy allows foundation skills to better support new-task learning while mitigating forgetting associated with foundation-skill drift during task-biased post-training. A fixed maintenance budget is dynamically allocated across foundation skills according to capability drift, independent of downstream task history. We evaluate SCALE on VQA-v3 under the SS and PS settings of UCo-VQA, with analyses of continual-learning performance, generalization, sensitivity, ablations and foundation-skill evolution. Our SCALE improves final average accuracy by 8.7% and reduces final forgetting by 9.1% points over matched continual SFT.

1 Introduction

Vision-language models (VLMs) have emerged as powerful general-purpose models with strong multimodal understanding and generalization across diverse downstream tasks (Radford et al., 2021). In practical deployments, changing data distributions and task demands require these models to continually incorporate new knowledge and adapt over time. Continual learning (CL) studies how models can learn from such non-stationary task streams while retaining previously acquired knowledge and capabilities (Shi et al., 2025; Razdaibiedina et al., 2023; Xu et al., 2024). However, continual post-training can induce catastrophic forgetting, leading to performance degradation on previously learned tasks (French, 1999; Kirkpatrick et al., 2017). For VLMs, this challenge extends beyond downstream task retention, as post-training can also impair the generalization capabilities inherited from large-scale pretraining. Addressing these challenges is important for maintaining the effectiveness and broad applicability of VLMs in evolving deployment environments.

To address this challenge, continual-learning has gradually expanded from retaining downstream task knowledge to preserving capabilities inherited from pretraining. For example, classical methods mitigate catastrophic forgetting through replay, regularization, and parameter isolation or expansion (Rebuffi et al., 2017; Zenke et al., 2017; Yu et al., 2024a); More recent work uses pretrained VLMs as behavioral references, preserves pretrained cross-modal invariants, or constrains updates according to pretrained parameter structure (Zheng et al., 2023; Zhang et al., 2026b; Jo & Kim, 2026). Amongst these developments above, the prevailing strategy remains to stabilize previously acquired knowledge through replay, parameter expansion, or constraints on model updates to limit interfer-

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

ence from subsequent learning. However, as shown in Figure 1, replay and parameter expansion can induce additional memory and computational overhead, while model constraints can reduce the plasticity needed to acquire new knowledge (Marouf et al., 2025; Wu et al., 2025). These observations motivate a insight toward understanding how the role of pretrained capabilities in VLMs can remain reusable as new downstream tasks comes. We hypothesize that task-biased post-training can induce drift in some reusable foundation skills, whose functional degradation weakens their support for both previously learned and new tasks. This hypothesis motivates foundation-skill co-evolution: preserving skill functionality while allowing its behavioral realizations to evolve with downstream tasks. We then ask: can maintaining foundation skills through co-evolution improve knowledge retention and new task adaptation?

To answer this question, we propose SCALE (**S**kill-**C**entric **A**daptation for **L**ifelong **E**volution), a continual learning framework that maintains the functionality of reusable foundation skills for VLMs, while allowing their behavioral realizations to co-evolve with the adapting model. SCALE uses held-out behavioral probes and drift-aware gates to identify skills whose functionality has degraded. For these skills, SCALE regenerates responses across continual stages and retains only those satisfying skill-specific verification criteria, allowing maintenance supervision to evolve with the model. At each stage, the incoming downstream task is learned through supervised fine-tuning (SFT) using only current-task data. To make this maintenance compatible with new-task learning, SCALE jointly optimizes the current-task and skill-maintenance objectives through a task-priority update: the current-task gradient remains unchanged, while only conflicting components of the maintenance gradient are removed. This strategy supports flexible adaptation to new tasks while selectively mitigating foundation-skill degradation associated with forgetting. Moreover, skill maintenance operates under a fixed total budget, which is dynamically redistributed across activated skills according to their measured drift. The total maintenance budget does not grow with downstream task number. Finally, experimental evaluations on VQA-v3 under the SS and PS settings of UCo-VQA demonstrate improved knowledge retention and model performance without historical downstream VQA replay.

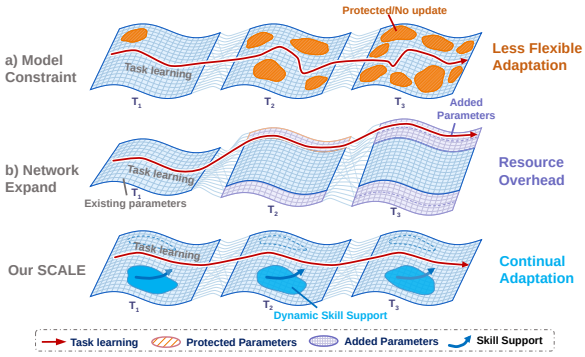


Figure 1: Comparison of continual learning mechanisms.

Our contributions are as follows: **(i) Foundation-skill perspective.** We introduce a foundation-skill perspective on continual VLM learning, which treats pretrained VLM capabilities as reusable skills defined independently of downstream task history and examines their role in retention and adaptation. **(ii) Task-prioritized co-evolution.** We enable foundation-skill behaviors to co-evolve with the adapting model through behavior regeneration and task-priority updates, while mitigating functional degradation from foundation-skill drift during task-biased post-training. **(iii) Drift-adaptive maintenance with a fixed budget.** We selectively activate degraded skills and dynamically redistribute a fixed total maintenance budget according to measured functional degradation. The total budget does not grow with number of downstream tasks.

2 Related Work

Pretrained Capabilities and Foundation Skills. Early foundation-model research primarily characterized pretrained models through the broad transfer, in-context learning, and scale-dependent abilities they exhibit, as exemplified by GPT-3 and studies of emergent abilities (Brown et al., 2020; Wei et al., 2022). Subsequent work examined model competence at a finer granularity: Skill-It and Skills-in-Context characterize skills through their dependencies and compositions (Chen et al., 2023; 2024b), while Skill Neurons and Function Vectors identify internal representations associated with particular tasks or functions (Wang et al., 2022; Todd et al., 2024). In multimodal foundation models, this capability-level view has become more concrete: MM-Vet, MMBench, SEED-Bench, and BLINK decompose overall VLM performance into fine-grained functional dimensions, such as recognition, OCR, spatial understanding, reasoning, and visual perception, primarily for capa-

bility evaluation and diagnosis (Yu et al., 2024b; Liu et al., 2024; Li et al., 2024; Fu et al., 2024). More recently, Vlaser and VLM4VLA have begun to investigate how different capabilities of pretrained VLMs relate to subsequent downstream learning and adaptation (Yang et al., 2026; Zhang et al., 2026a). However, these studies mainly focus on capability characterization, diagnosis, or one-shot downstream adaptation, leaving the role of distinct pretrained capabilities in continual learning largely unexplored. SCALE organizes selected pretrained capabilities into a set of reusable *foundation skills* and studies how they can be leveraged throughout continual learning to jointly support adaptation and retention.

Continual Learning for VLMs. Continual learning has traditionally addressed catastrophic forgetting by preserving previously acquired downstream knowledge through three major strategies: replaying past examples or experiences, regularizing changes to previously learned functions or important parameters, and isolating or allocating task-specific capacity. Representative methods include iCaRL, GEM, and DER for replay-based retention, LwF and EWC for functional or parameter regularization, and PackNet and HAT for parameter isolation (Rebuffi et al., 2017; Lopez-Paz & Ranzato, 2017; Buzzega et al., 2020; Li & Hoiem, 2016; Kirkpatrick et al., 2017; Mallya & Lazebnik, 2018; Serra et al., 2018). These principles have subsequently been extended to VLM continual learning: Symbolic Replay, VQACL, GaB, and CoMem retain historical task knowledge through rehearsal, generation, or structured memory (Lei et al., 2023; Zhang et al., 2023; Das et al., 2025; Zhou et al., 2026), whereas MoE-Adapters, PROOF, and SMoLoRA reduce interference through specialized adapters, routing mechanisms, or task-specific capacity (Yu et al., 2024a; Zhou et al., 2025; Wang et al., 2025). More recently, the preservation target has expanded beyond downstream history to the pretrained foundation itself. ZSCL and C-CLIP explicitly preserve pretrained zero-shot or multimodal capabilities (Zheng et al., 2023; Liu et al., 2025), while DIKI, DFA-MoE, DMNSP, NuSA-CL, and Pi-CCA retain pretrained knowledge through frozen or separated pathways, restricted update directions, or preserved representation and alignment structures (Tang et al., 2024; Kang et al., 2026; 2025; Jo & Kim, 2026; Zhang et al., 2026b). Despite their different mechanisms, these approaches largely pursue retention by replaying prior information, separating capacity, preserving pretrained references or invariants, or restricting subsequent updates, with the common goal of maintaining previously established knowledge and functionality. SCALE instead introduces *co-evolving foundation maintenance*: it maintains the functionality of foundation skills while allowing their behavioral realizations to adapt together with the continually learning model.

3 METHOD

At each continual stage, as illustrated in Fig. 2, SCALE monitors foundation skills using fixed held-out behavioral probes and activates only the skills whose functionality degrades. A fixed maintenance budget is then dynamically reallocated across the activated skills according to their measured drift. For each selected skill, a behavior policy along the evolving model trajectory regenerates candidate behaviors on foundation prompts, and verifier-accepted behaviors provide adaptive behavioral supervision for skill maintenance rather than fixed targets tied to the pretrained model. Current-task supervised fine-tuning and foundation-skill maintenance are jointly optimized under a task-priority update that removes only maintenance-gradient components conflicting with the current task, thereby prioritizing new-task learning while retaining reusable capabilities. Through this process, foundation skills co-evolve with the adapting model: their valid behavioral realizations can change to better support new-task learning, while functionality degradation induced by task-biased post-training is selectively mitigated, reducing forgetting associated with foundation-skill drift.

3.1 PRELIMINARIES AND FOUNDATION-SKILL SETUP

We consider a continual stream of downstream tasks $\mathcal{D}_1, \ldots, \mathcal{D}_T$ following the continual VQA setting of UCo-VQA (Gao et al., 2026). We use Molmo2-4B (Clark et al., 2026) as the pretrained VLM. At stage t, the VLM has access only to the current dataset $\mathcal{D}_t = \{(x_i^t, y_i^t)\}_{i=1}^{N_t}$ and does not store examples from $\mathcal{D}_{<t}$. At every stage, the current downstream task is learned through supervised fine-tuning (SFT) on $\mathcal{D}_t$. Let $\pi_{\theta_t}(y \mid x)$ denote the VLM at the current adaptation state. We use LoRA (Hu et al., 2022) for parameter-efficient adaptation and write $\theta_t = (\bar{\theta}, \phi_t)$, where $\bar{\theta}$ denotes the frozen pretrained backbone and ϕ_t the trainable low-rank parameters.

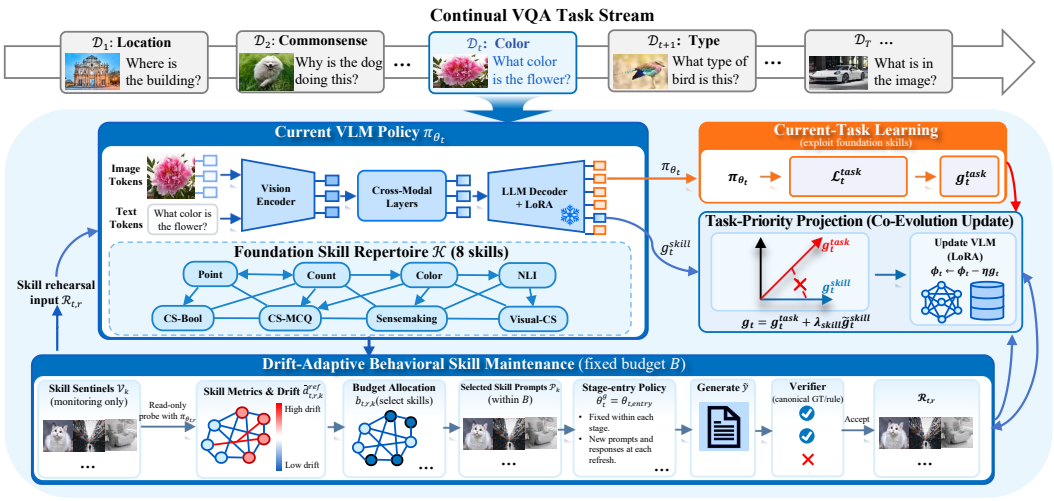


Figure 2: Overview of SCALE. **Foundation Skill Repertoire:** SCALE treats reusable capabilities inherited from the pretrained VLM as foundation skills supporting adaptation and retention. **Drift-Adaptive Behavioral Skill Maintenance:** behavioral probes identify degraded skills and dynamically allocate a fixed budget B. The stage-entry policy $\theta_t^g = \theta_{t,\text{entry}}$ regenerates candidate behaviors on foundation prompts, with verifier-accepted responses providing evolving supervision. **Task-Priority Projection:** current-task SFT and skill maintenance are jointly optimized. The task gradient g_t^{task} remains unchanged, while conflicting components of g_t^{skill} are projected: $g_t = g_t^{\text{task}} + \lambda_{\text{skill}}\widetilde{g}_t^{\text{skill}}$. This task-priority update maintains skill functionality while enabling co-evolution with the adapting VLM.

From the training data underlying the pretrained VLM, we organize eight foundation-skill categories: Point, Count, Attribute/Color, NLI, Commonsense Bool, Commonsense MCQ, Sensemaking, and Visual Commonsense. We index these skills by $k \in \mathcal{K} = \{1, \ldots, 8\}$. For each skill k, we construct a foundation prompt pool $\mathcal{P}_k$ for regenerating skill behaviors along the evolving VLM trajectory, and reserve a disjoint fixed monitoring subset $\mathcal{V}_k$ from the corresponding skill data source to track the skill state during continual adaptation. The monitoring subset $\mathcal{V}_k$ is never used for behavior generation or optimization. Within stage t, let r index the maintenance refreshes and $\theta_{t,r}$ denote the current VLM parameters at refresh r. A deterministic skill-specific evaluation function M_k evaluates the model outputs on $\mathcal{V}_k$, yielding

$$m_{t,r,k} = M_k(\theta_{t,r}; \mathcal{V}_k), \tag{1}$$

as a diagnostic estimate of the current functionality of skill k, while $m_{0,k}$ denotes the corresponding score of the pretrained model. For example, M_k uses Point F1 for grounding, exact accuracy for counting, normalized answer matching for color, and task-specific accuracy for the semantic skills.

3.2 Drift-Guided Skill Selection and Budget Allocation

Rather than allocating maintenance uniformly across all foundation skills, SCALE allocates it according to each skill's observed functionality degradation. For skill k at refresh r of stage t, we measure its relative degradation with respect to the fixed pretrained reference score $m_{0,k}$:

$$d_{t,r,k}^{\text{ref}} = \max\left(0, \frac{m_{0,k} - m_{t,r,k}}{m_{0,k} + \epsilon}\right), \tag{2}$$

where $\epsilon > 0$ avoids numerical instability. At every refresh, $d_{t,r,k}^{\text{ref}}$ compares the current skill score $m_{t,r,k}$ with the same pretrained reference $m_{0,k}$; it does not measure the change from the previous refresh. To characterize whether such degradation persists across refreshes, we maintain an exponential moving average over the sequence of reference-degradation measurements:

$$\bar{d}_{t,r,k}^{\text{ref}} = \beta \bar{d}_{t,r-1,k}^{\text{ref}} + (1-\beta) d_{t,r,k}^{\text{ref}}, \tag{3}$$

where $\beta \in [0, 1)$ is the smoothing coefficient. Thus, $\bar{d}_{t,r,k}^{\text{ref}}$ aggregates the current and previous degradation measurements relative to the same pretrained reference, assigning larger weight to more recent refreshes. It retains persistent degradation while gradually discounting older signals as the skill recovers, providing a stable basis for skill selection and maintenance allocation.

Based on this smoothed degradation signal, we activate skill k for maintenance when its smoothed reference degradation exceeds the activation threshold:

$$a_{t,r,k} = \mathbb{I}\left[\bar{d}^{\mathrm{ref}}_{t,r,k} > \tau_{\mathrm{ref}}\right], \tag{4}$$

where τ_{ref} denotes the activation threshold and $\mathbb{I}[\cdot]$ is the indicator function. Thus, $a_{t,r,k} = 1$ indicates that skill k is selected for maintenance. Importantly, the drift signal measures degradation in skill functionality rather than arbitrary behavioral change from the pretrained model. We define the maintenance weight of each skill as

$$w_{t,r,k} = a_{t,r,k}\bar{d}^{\mathrm{ref}}_{t,r,k}. \tag{5}$$

Using these drift-aware weights, and given a fixed total maintenance budget B, SCALE dynamically redistributes the budget across the activated skills in proportion to their measured drift:

$$b_{t,r,k} = \mathrm{Round}\left(B\frac{w_{t,r,k}}{\sum_{j\in\mathcal{K}} w_{t,r,j}}\right), \qquad \sum_{k\in\mathcal{K}} b_{t,r,k} \le B, \tag{6}$$

with $b_{t,r,k} = 0$ for inactive skills. If no skill is activated, no foundation-skill maintenance batch is constructed. Because the total budget B is fixed, the amount of foundation maintenance does not grow with the number of previously observed downstream tasks.

3.3 Verifier-Guided Behavioral Maintenance

Once a foundation skill is activated, SCALE regenerates verified skill behaviors instead of replaying fixed responses from the pretrained model, allowing its maintenance supervision to evolve with the VLM. Specifically, at stage t, we use the model checkpoint before adapting to the current downstream task as the stage-entry generation policy:

$$\theta^g_t = \theta_{t,\mathrm{entry}}. \tag{7}$$

Here, $\theta_{t,\mathrm{entry}}$ denotes the VLM state at the beginning of stage t, before any updates on the current downstream task. The generation policy remains fixed within the stage, while each maintenance refresh samples new prompts from $\mathcal{P}_k$ and regenerates candidate responses. We choose the stage-entry policy because a current model whose skill has already degraded may yield too few valid responses after verification, whereas the stage-entry checkpoint remains close to the evolving model while providing more stable behavioral coverage. We further examine this design in the *Policy design ablation* under Sec. 4.3.

For an activated skill k at refresh r, we sample a foundation prompt:

$$x \sim \mathcal{P}_k, \tag{8}$$

and generate a candidate response from the stage-entry policy:

$$\tilde{y} \sim \pi_{\theta^g_t}(\cdot \mid x). \tag{9}$$

The same deterministic skill-specific evaluation function M_k introduced above is used to verify whether the generated response satisfies the corresponding skill criterion. Only verified behaviors are retained, up to the maximum number $b_{t,r,k}$ allocated to each skill:

$$\mathcal{R}_{t,r,k} = \left\{(x,\tilde{y}) \;\middle|\; x\sim\mathcal{P}_k,\ \tilde{y}\sim\pi_{\theta^g_t}(\cdot\mid x),\ M_k(x,\tilde{y})=1\right\}, \qquad |\mathcal{R}_{t,r,k}| \le b_{t,r,k}. \tag{10}$$

The verified behaviors from all activated skills are combined as

$$\mathcal{R}_{t,r} = \bigcup_{k\in\mathcal{K}} \mathcal{R}_{t,r,k}, \tag{11}$$

and used to optimize the current model with

$$\mathcal{L}^{\mathrm{skill}}_{t,r} = -\frac{1}{|\mathcal{R}_{t,r}|}\sum_{(x,\tilde{y})\in\mathcal{R}_{t,r}}\sum_{\ell=1}^{|\tilde{y}|}\log p_{\theta_{t,r}}\left(\tilde{y}_\ell \mid x, \tilde{y}_{<\ell}\right). \tag{12}$$

Thus, foundation-skill maintenance is based on verified behaviors regenerated by stage-entry policies that are refreshed along the continual stream, rather than fixed responses tied to the pretrained checkpoint.

Table 1: Continual-learning performance on VQA-v3 under SS and PS. All metrics are reported in percentage (%).

Method	Venue	SS			PS		
		FAA↑	CAA↑	FFM↓	FAA↑	CAA↑	FFM↓
Joint Training (Upper Ref.)	–	63.16	–	–	26.83	–	–
MaDQ-QReplay	CVPR 2026	62.22	59.91	2.18	21.36	**36.00**	28.23
GaB	WACV 2025	56.81	59.37	10.14	16.93	33.10	39.65
DGG	CVPR 2026	45.73	55.61	14.13	27.77	32.99	38.79
CL-MoE	CVPR 2025	43.77	50.19	**0.09**	16.04	27.32	**0.56**
MoELoRA	NeurIPS 2024	51.67	59.08	14.05	25.03	31.55	39.34
D-MoLE	ICML 2025	44.19	50.78	1.22	21.52	33.12	6.48
SCALE (Ours)	–	**63.53**	**62.46**	2.91	**29.19**	34.63	37.66

3.4 Task-Priority Joint Update

The current downstream task is learned with standard supervised fine-tuning,

$$\mathcal{L}_t^{\text{task}} = -\frac{1}{|\mathcal{B}_t|} \sum_{(x,y)\in\mathcal{B}_t} \log \pi_{\theta_t}(y \mid x), \qquad \mathcal{B}_t \subset \mathcal{D}_t. \tag{13}$$

We separately compute the gradients of the current-task and foundation-skill objectives with respect to the trainable parameters ϕ_t:

$$g_t^{\text{task}} = \nabla_{\phi_t} \mathcal{L}_t^{\text{task}}, \qquad g_t^{\text{skill}} = \nabla_{\phi_t} \mathcal{L}_t^{\text{skill}}. \tag{14}$$

When the two gradients conflict, directly combining them can reduce progress on the newly arriving task. Following the gradient-projection principle used to mitigate conflicting objectives (Yu et al., 2020), we use an asymmetric task-priority projection:

$$\widetilde{g}_t^{\text{skill}} = \begin{cases} g_t^{\text{skill}} - \dfrac{\langle g_t^{\text{skill}}, g_t^{\text{task}} \rangle}{\|g_t^{\text{task}}\|_2^2 + \epsilon} g_t^{\text{task}}, & \langle g_t^{\text{skill}}, g_t^{\text{task}} \rangle < 0, \\ g_t^{\text{skill}}, & \text{otherwise.} \end{cases} \tag{15}$$

Unlike symmetric gradient surgery, the current-task gradient is left unchanged; only the conflicting component of the foundation-skill gradient is projected, so that skill maintenance remains compatible with current-task learning.

The final update direction is

$$g_t = g_t^{\text{task}} + \lambda_{\text{skill}} \widetilde{g}_t^{\text{skill}}, \tag{16}$$

and the trainable parameters are updated as

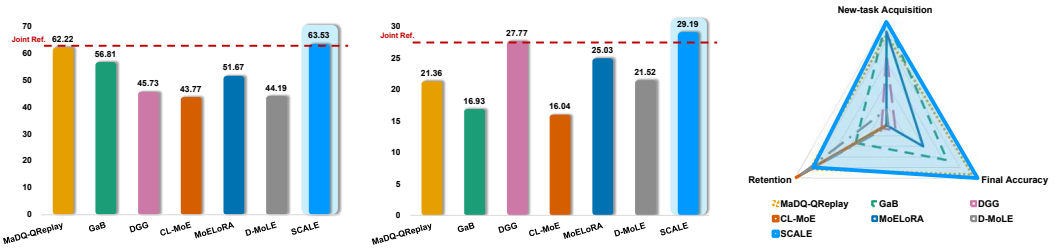
$$\phi_t \leftarrow \phi_t - \eta g_t. \tag{17}$$

Thus, the task-priority update allows foundation skills to co-evolve with the model while maintaining their functionality, enabling the VLM to continually adapt to new downstream tasks.

4 Experiments

4.1 Experimental Setup

We evaluate SCALE on VQA-v3 from UCo-VQA (Gao et al., 2026) under Standard Splits (SS) and Proposed Splits (PS), comprising eight sequential VQA tasks. Following UCo-VQA, we report Final Average Accuracy (FAA), Continual Average Accuracy (CAA), and Final Forgetting Measure (FFM). Except for the sensitivity analysis, all experiments use the official task order and seed 96817. All methods use Molmo2-4B step2000 (Clark et al., 2026) as the initialization with LoRA adaptation (Hu et al., 2022). SCALE maintains eight foundation skills: Point, Count, Attribute/Color, NLI, Commonsense Bool, Commonsense MCQ, Sensemaking, and Visual Commonsense, constructed



(a) Final Accuracy on SS (b) Final Accuracy on PS (c) Stability–Plasticity Profile

Figure 3: Main continual-learning comparison on VQA-v3. (a–b) Final Average Accuracy (FAA) under SS and PS. (c) Stability–plasticity profile on SS.

Table 2: Policy-design ablation on VQA-v3 under SS and PS. All metrics are reported in percentage (%). The left columns summarize the PT behavior design. Values in parentheses denote the difference relative to SCALE.

Setting	PT	Evolve	Policy	Rollout	SS			PS		
					FAA↑	CAA↑	FFM↓	FAA↑	CAA↑	FFM↓
No PT	✗	✗	–	–	54.85 (-8.68)	59.91 (-2.55)	11.99 (+9.08)	24.04 (-5.15)	32.28 (-2.35)	43.09 (+5.43)
Direct PT Replay	✓	✗	–	–	55.66 (-7.87)	60.00 (-2.46)	10.32 (+7.41)	24.59 (-4.60)	33.93 (-0.70)	39.96 (+2.30)
Latest	✓	✓	Latest	Refresh	63.05 (-0.48)	61.30 (-1.16)	3.68 (+0.77)	25.88 (-3.31)	33.35 (-1.28)	41.94 (+4.28)
Stale	✓	✓	Stage-entry	Once	59.05 (-4.48)	59.88 (-2.58)	5.00 (+2.09)	28.69 (-0.50)	33.75 (-0.88)	**33.10** (-4.56)
SCALE	✓	✓	Stage-entry	Refresh	**63.53**	**62.46**	**2.91**	**29.19**	**34.63**	37.66

from public pretraining data associated with Molmo2. Each stage contains eight maintenance refreshes with a fixed budget $B = 32$ and $\lambda_{\text{skill}} = 0.1$, without historical downstream VQA replay. We compare with MaDQ-QReplay (Gao et al., 2026), GaB (Das et al., 2025), DGG (Li et al., 2026), CL-MoE (Huai et al., 2025), MoELoRA (Chen et al., 2024a), and D-MoLE (Ge et al., 2025), and report Joint Training as an upper reference. Full configurations are provided in the appendix.

4.2 Continual Learning Performance

Table 1 summarizes the main continual-learning results on VQA-v3 under SS and PS, while Figure 3 visualizes final performance and the stability–plasticity trade-off. Complete stage-wise results are provided in the appendix.

Under SS, SCALE achieves the highest FAA (63.53%) and CAA (62.46%) among continual-learning methods. Compared with MaDQ-QReplay, SCALE improves FAA and CAA by 1.31 and 2.55 percentage points, respectively, while maintaining a low FFM of 2.91%. Although CL-MoE and D-MoLE obtain lower FFM, their FAA is substantially lower (43.77% and 44.19%). In contrast, SCALE combines strong task acquisition with substantially lower forgetting than GaB, DGG, and MoELoRA, reducing FFM from 10.14–14.13% to 2.91%.

A similar pattern holds under the more challenging PS setting. SCALE achieves the highest FAA of 29.19%, exceeding MaDQ-QReplay, DGG, and MoELoRA by 7.83, 1.42, and 4.16 percentage points, respectively. Notably, CL-MoE reaches an FFM of only 0.56%, but this is accompanied by a much lower FAA of 16.04%, indicating substantially weaker task acquisition. Meanwhile, SCALE achieves lower FFM than GaB, DGG, and MoELoRA (37.66% vs. 39.65%/38.79%/39.34%) while maintaining the highest final accuracy. Overall, the SS and PS results show that SCALE consistently supports new-task adaptation while mitigating forgetting, yielding a favorable balance between adaptation and retention.

4.3 Ablation and Sensitivity Analysis

Policy design ablation. Table 2 studies the design of PT maintenance. To isolate the effect of PT behavior design, all settings use the same downstream SFT procedure and differ only in how PT supervision is introduced. Here, **PT** refers to foundation-skill data constructed from the public pre-training data associated with the base Molmo2 VLM. We first compare No PT with Direct PT Replay, showing that PT rehearsal benefits continual SFT, while directly replaying fixed PT targets provides only limited gains. We then study how PT behaviors should evolve with the model. **Latest** uses the most recent checkpoint at every refresh to generate PT responses, whereas **Stale** and

Table 3: Sensitivity of SCALE to random seeds and task orders on SS. Results are mean ± std over three runs.

Variation	FAA↑	CAA↑	Final-stage forgetting↓
Seed (official order)	62.78 ± 0.66	62.31 ± 0.14	2.96 ± 0.91
Task order (seed 96817)	63.51 ± 0.43	60.66 ± 2.50	2.91 ± 0.43

Table 4: Budget sensitivity on VQA-v3 under SS. All metrics are reported in percentage (%). The selected configuration is highlighted in blue, and the best overall results are in bold.

Cap	Stage-wise AA↑								Overall		
	Loc.	Common.	Type	Action	Color	Count	Recog.	Judge	FAA↑	CAA↑	FFM↓
24	58.95	74.60	56.98	61.75	59.45	54.76	54.99	57.28	57.28	59.85	7.44
32 (SCALE)	59.27	74.51	58.24	62.19	64.72	57.08	60.17	63.53	**63.53**	**62.46**	**2.91**
40	55.95	80.96	56.59	68.30	64.05	59.31	58.52	60.95	63.08	62.24	3.69

SCALE both use the model checkpoint saved at the beginning of each stage as a fixed generation policy throughout that stage, which we refer to as the stage-entry policy. **Stale** generates behaviors once at stage entry and reuses them throughout the stage, whereas SCALE resamples prompts and regenerates verified behaviors at every refresh using the same stage-entry policy. As shown in Table 2, SCALE improves FAA over Latest by 0.48 and 3.31 percentage points on SS and PS, respectively, and consistently achieves higher FAA and CAA than Stale. These results suggest that an overly fresh policy can be affected by drift in the current model, whereas repeatedly reusing fixed rollouts leads to increasing staleness. SCALE therefore provides a better balance between policy stability and behavioral evolution.

Seed and order sensitivity. As shown in Table 3, we further evaluate the robustness of SCALE to random seeds and task orders. For each sensitivity setting, we conduct three runs and report the mean ± standard deviation. Under the official task order, SCALE achieves 62.78 ± 0.66 FAA and 62.31 ± 0.14 CAA across different random seeds. When fixing seed 96817 and varying the task order, FAA remains stable at 63.51 ± 0.43. The small standard deviations indicate that the final continual-learning performance of SCALE is robust to both random initialization and task ordering, suggesting that its gains are not tied to a particular experimental configuration.

Maintenance budget sensitivity. Table 4 evaluates sensitivity to the maintenance budget, where **cap** denotes the maximum number of verifier-accepted foundation-skill samples used at each refresh. Reducing the cap from 32 to 24 lowers FAA and CAA and increases FFM, suggesting that an overly restrictive budget limits foundation-skill maintenance. However, increasing the cap from 32 to 40 does not improve performance: both FAA and CAA decrease, while FFM increases. These results suggest that simply allocating a larger maintenance budget does not necessarily yield better continual-learning performance. We therefore adopt a cap of 32 for SCALE, which achieves the best performance among the tested budgets while keeping the per-refresh maintenance budget fixed independently of downstream task-history length.

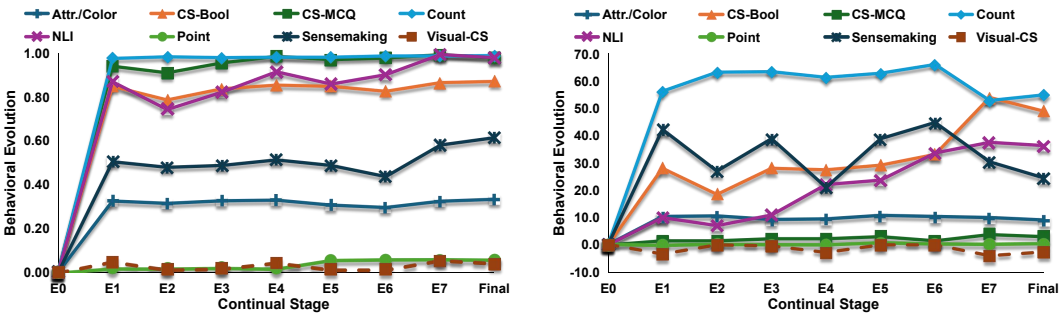
4.4 Foundation-Skill Co-Evolution Analysis

To examine how foundation skills evolve during continual learning, we compare their behavioral realizations with the pretrained state on the fixed monitoring subsets $\mathcal{V}_k$ introduced in Sec. 3.1, using identical prompting, decoding, parsing, and evaluation settings across all checkpoints. For skill k, let B_0^k and B_t^k denote the parsed behaviors of the pretrained checkpoint and the checkpoint after stage t, respectively. We define behavioral evolution relative to pretraining as

$$\Delta_{t,k}^{\mathrm{beh}} = D_k(B_t^k, B_0^k), \tag{18}$$

where D_k is a skill-specific distance between parsed outputs. As shown in Fig. 4(a), all eight foundation skills develop behavioral realizations different from their pretrained state over the continual stream. Differences between consecutive stages further show that these behaviors continue to evolve throughout continual learning rather than changing only at the beginning of adaptation.

Behavioral change alone does not indicate whether skill functionality is preserved. Using the same skill-specific evaluation function M_k defined in Sec. 3.1, we measure the stage-level functionality



(a) Behavioral change of foundation skills. (b) Functionality change of foundation skills.

Figure 4: **Behavioral change and functionality of foundation skills under SCALE.** (a) Stage-wise behavioral change of each foundation skill relative to its pretrained behavior. (b) Stage-wise functionality change of each foundation skill relative to its pretrained state.

Table 5: Behavioral evolution and functionality changes of foundation skills. E1–E8 report behavioral change $D_k(B_t^k, B_0^k)$ relative to pretraining after stages 1–8. Q_0 and Q_8 denote functionality before continual learning and after stage 8, respectively, with $\Delta Q = Q_8 - Q_0$. Blue shading highlights the final-stage columns.

Skill	**Behavioral change**								**Functionality**		
	E1	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	Q_0	Q_8	ΔQ
Attr./Color	0.324	0.313	0.325	0.328	0.306	0.295	0.323	0.332	0.575	0.663	+0.088
CS Bool	0.848	0.787	0.837	0.854	0.848	0.826	0.865	0.871	0.169	0.657	+0.489
CS MCQ	0.940	0.910	0.955	0.985	0.970	0.978	0.993	0.978	0.657	0.687	+0.030
Count	0.976	0.983	0.979	0.981	0.981	0.987	0.987	0.989	0.000	0.548	+0.548
NLI	0.870	0.744	0.822	0.915	0.859	0.902	0.993	0.979	0.017	0.377	+0.360
Point	0.018	0.017	0.020	0.017	0.053	0.056	0.056	0.054	0.549	0.552	+0.003
Sensemaking	0.504	0.479	0.487	0.513	0.487	0.437	0.580	0.613	0.336	0.580	+0.244
Visual CS	0.043	0.012	0.016	0.039	0.012	0.012	0.049	0.037	0.963	0.936	−0.027

as

$$Q_t^k = M_k(\theta_{t,\text{end}}; \mathcal{V}_k), \qquad Q_0^k = M_k(\theta_0; \mathcal{V}_k), \tag{19}$$

where $\theta_{t,\text{end}}$ denotes the checkpoint after stage t. As shown in Fig. 4(b), most foundation skills maintain or improve their functionality while their behavioral realizations evolve beyond the pretrained state. Visual Commonsense shows a modest decline under task-biased post-training while retaining high absolute functionality. Table 5 further reports the behavioral changes of foundation skills and compares their pretrained and final functionality scores. More detailed functionality results are provided in the appendix.

Together, these results show that foundation skills can evolve toward behavioral realizations different from their pretrained state while maintaining or improving their functionality, providing evidence of functional co-evolution.

5 Conclusion

This work revisits what should be preserved in continual VLM learning. We show that forgetting can stem not only from the loss of downstream knowledge, but also from the degradation of reusable foundation skills under task-biased post-training. Motivated by this insight, SCALE shifts the preservation target from fixed past states to foundation-skill functionality and studies how reusable foundation skills contribute to adaptation and retention throughout continual learning.

Empirically, SCALE improves adaptation and retention on VQA-v3 under SS and PS without historical downstream VQA replay, while foundation-skill analysis shows that behaviors can evolve as functionality is maintained or improved. These results suggest that continual VLM learning can adapt to new tasks without being limited to preserving fixed past states, instead maintaining and evolving reusable foundation skills to mitigate forgetting associated with foundation-skill drift. Future work may explore automatic skill decomposition, organization, and allocation, as well as fixed-budget maintenance over longer and more diverse continual-learning streams.

AI Use Statement. Generative AI tools were used to assist with manuscript drafting, language polishing, and improving the clarity and organization of the presentation. All AI-assisted content was reviewed and verified by the authors, who take full responsibility for the final text, claims, and presentation of this work.

Ethics Statement. This work adheres to the ICLR Code of Ethics. Our study does not involve the collection of new human-subject data or private user data. Experiments are conducted on publicly available vision–language benchmarks and pretraining data associated with the base VLM, following their intended research use. We are not aware of any additional ethical risks specific to the proposed continual-learning methodology.

Reproducibility Statement. We describe the proposed method and continual-training procedure in Sec. 3. The benchmark setting, task order, evaluation metrics, model checkpoint, and main continual-learning results are reported in Sec. 4.2, with ablation and sensitivity analyses in Sec. 4.3 and foundation-skill co-evolution analysis in Sec. 4.4. Additional implementation and optimization details, together with complete stage-wise results, ablation settings, and sensitivity results, are provided in the appendix and supplementary material to facilitate reproducibility.

REFERENCES

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: A strong, simple baseline. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Cheng Chen, Junchen Zhu, Xu Luo, Heng T Shen, Jingkuan Song, and Lianli Gao. Coin: A benchmark of continual instruction tuning for multimodel large language models. *Advances in neural information processing systems*, 37:57817–57840, 2024a.

Jiaao Chen, Xiaoman Pan, Dian Yu, Kaiqiang Song, Xiaoyang Wang, Dong Yu, and Jianshu Chen. Skills-in-context: Unlocking compositionality in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 13838–13890. Association for Computational Linguistics, 2024b. doi: 10.18653/v1/2024.findings-emnlp.812.

Mayee F. Chen, Nicholas Roberts, Kush Bhatia, Jue Wang, Ce Zhang, Frederic Sala, and Christopher Ré. Skill-it! a data-driven skills framework for understanding and training language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, et al. Molmo2: Open weights and data for vision-language models with video understanding and grounding, 2026. *URL https://arxiv. org/abs/2601.10611*, 2026.

Deepayan Das, Davide Talon, Massimiliano Mancini, Yiming Wang, and Elisa Ricci. One vlm to keep it learning: Generation and balancing for data-free continual visual question answering. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025.

Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, pp. 148–166. Springer, 2024. doi: 10.1007/978-3-031-73337-6_9.

Zijian Gao, Zicheng Sun, Xingxing Zhang, Kele Xu, and Huaimin Wang. Re-evaluating continual vqa: Toward fair and robust evaluation for multimodal continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18021–18031, 2026.

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Tianyu Huai, Jie Zhou, Xingjiao Wu, Qin Chen, Qingchun Bai, Ze Zhou, and Liang He. Cl-moe: Enhancing multimodal large language model with dual momentum mixture-of-experts for continual visual question answering. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19608–19617. IEEE, 2025.

Yujin Jo and Taesup Kim. Memory-free continual learning with null space adaptation for zero-shot vision-language models. In *International Conference on Learning Representations*, volume 2026, pp. 106089–106106, 2026.

Borui Kang, Lei Wang, Zhiping Wu, Tao Feng, Yawen Li, Yang Gao, and Wenbin Li. Dynamic multi-layer null space projection for vision-language continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2077–2086, 2025. doi: 10.1109/ICCV51701.2025.00201.

Borui Kang, Jinrui Gu, Tao Feng, Qi Fan, Yinghuan Shi, Lei Wang, Wenbin Li, and Yang Gao. Don't forget why you started: Tackling dual forgetting in vision-language continual learning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

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

Stan Weixian Lei, Difei Gao, Jay Zhangjie Wu, Yuxuan Wang, Wei Liu, Mengmi Zhang, and Mike Zheng Shou. Symbolic replay: Scene graph as prompt for continual learning on vqa task. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 1250–1259, 2023. doi: 10.1609/aaai.v37i1.25208.

Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. SEED-Bench: Benchmarking multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13299–13308, 2024.

Songze Li, Mingyu Gao, Tonghua Su, Xu-Yao Zhang, and Zhongjie Wang. Multimodal continual instruction tuning with dynamic gradient guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10820–10829, 2026.

Zhizhong Li and Derek Hoiem. Learning without forgetting. In *European Conference on Computer Vision*, pp. 614–629, 2016. doi: 10.1007/978-3-319-46493-0_37.

Wenzhuo Liu, Fei Zhu, Longhui Wei, and Qi Tian. C-CLIP: Multimodal continual learning for vision-language model. In *International Conference on Learning Representations*, 2025.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. MMBench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision*, pp. 216–233. Springer, 2024. doi: 10.1007/978-3-031-72658-3_13.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, volume 30, pp. 6467–6476, 2017.

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*, pp. 7765–7773, 2018.

Imad Eddine Marouf, Enzo Tartaglione, Stéphane Lathuilière, and Joost Van De Weijer. Ask and remember: A questions-only replay strategy for continual visual question answering. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18078–18089. IEEE, 2025.

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 *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021.

Anastasia Razdaibiedina, Yuning Mao, Rui Hou, Madian Khabsa, Mike Lewis, and Amjad Almahairi. Progressive prompts: Continual learning for language models. *arXiv preprint arXiv:2301.12314*, 2023.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 5533–5542. IEEE, 2017.

Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4548–4557. PMLR, 2018.

Haizhou Shi, Zihao Xu, Hengyi Wang, Weiyi Qin, Wenyuan Wang, Yibin Wang, Zifeng Wang, Sayna Ebrahimi, and Hao Wang. Continual learning of large language models: A comprehensive survey. *ACM Computing Surveys*, 58(5):1–42, 2025.

Longxiang Tang, Zhuotao Tian, Kai Li, Chunming He, Hantao Zhou, Hengshuang Zhao, Xiu Li, and Jiaya Jia. Mind the interference: Retaining pre-trained knowledge in parameter efficient continual learning of vision-language models. In *European Conference on Computer Vision*, pp. 346–365, 2024. doi: 10.1007/978-3-031-72764-1_20.

Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. Function vectors in large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Xiaozhi Wang, Kaiyue Wen, Zhengyan Zhang, Lei Hou, Zhiyuan Liu, and Juanzi Li. Finding skill neurons in pre-trained transformer-based language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 11132–11152. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.emnlp-main.765.

Ziqi Wang, Chang Che, Qi Wang, Yangyang Li, Zenglin Shi, and Meng Wang. SMoLoRA: Exploring and defying dual catastrophic forgetting in continual visual instruction tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 177–186, 2025.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022.

Bin Wu, Wuxuan Shi, Jinqiao Wang, and Mang Ye. Synthetic data is an elegant gift for continual vision-language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2813–2823. IEEE, 2025.

Yicheng Xu, Yuxin Chen, Jiahao Nie, Yusong Wang, Huiping Zhuang, and Manabu Okumura. Advancing cross-domain discriminability in continual learning of vision-language models. *Advances in Neural Information Processing Systems*, 37:51552–51576, 2024.

Ganlin Yang, Tianyi Zhang, Haoran Hao, Weiyun Wang, Yibin Liu, Dehui Wang, Guanzhou Chen, Zijian Cai, Junting Chen, Weijie Su, Wengang Zhou, Yu Qiao, Jifeng Dai, Jiangmiao Pang, Gen Luo, Wenhai Wang, Yao Mu, and Zhi Hou. Vlaser: Vision-Language-Action Model with Synergistic Embodied Reasoning. In *International Conference on Learning Representations*, 2026. URL https://mlanthology.org/iclr/2026/yang2026iclr-vlaser/.

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

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. MM-vet: Evaluating large multimodal models for integrated capabilities. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57730–57754. PMLR, 2024b.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pp. 3987–3995. Pmlr, 2017.

Jianke Zhang, Xiaoyu Chen, Yanjiang Guo, Yucheng Hu, and Jianyu Chen. VLM4VLA: Revisiting vision-language-models in vision-language-action models. In *The Fourteenth International Conference on Learning Representations*, 2026a.

Jiayu Zhang, Chuangxin Zhao, Canran Xiao, Ruibo Duan, Wenyi Mo, Haoyu Gao, and Wenshuo Wang. Pi-cca: Prompt-invariant cca certificates for replay-free continual multimodal learning. In *International Conference on Learning Representations*, volume 3, 2026b.

Xi Zhang, Feifei Zhang, and Changsheng Xu. VQACL: A novel visual question answering continual learning setting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19102–19112, 2023.

Zangwei Zheng, Mingyuan Ma, Kai Wang, Ziheng Qin, Xiangyu Yue, and Yang You. Preventing zero-shot transfer degradation in continual learning of vision-language models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19068–19079. IEEE, 2023.

Da-Wei Zhou, Yuanhan Zhang, Yan Wang, Jingyi Ning, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Learning without forgetting for vision-language models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(6):4489–4504, 2025. doi: 10.1109/TPAMI.2025.3540889.

Heng Zhou, Jing Tang, Jusheng Zhang, Yanshu Li, Canran Xiao, Liwei Hou, Zong Ke, and Jiawei Yao. CoMem: Compositional concept-graph memory for vision–language adaptation. In *International Conference on Learning Representations*, 2026.

APPENDIX

OVERVIEW

This supplementary material provides additional details and results for SCALE that complement the main paper.

(1) Additional algorithmic details (Sec. A). We provide the complete continual-adaptation procedure, including the stage-entry generation policy, periodic foundation-skill monitoring, maintenance-budget allocation, behavior regeneration, and data separation.

(2) Experimental configuration (Sec. B). We describe the VQA-v3 continual-learning setup, optimization configuration, and the construction and verification of the foundation-skill repertoire.

(3) Additional results (Sec. C). We report complete stage-wise results and additional analyses of policy design, hyperparameter sensitivity, maintenance budget, and foundation-skill evolution.

(4) Limitations (Sec. D). We discuss the scope of the current evaluation and the limitations of foundation-skill construction and maintenance.

A ADDITIONAL ALGORITHMIC DETAILS

At the beginning of stage t, the checkpoint before adaptation to the current downstream task is retained as the stage-entry generation policy and remains fixed throughout the stage. At each maintenance refresh, foundation skills are evaluated on their fixed monitoring subsets. Skills exhibiting functionality degradation are activated for maintenance, and the fixed budget B is distributed among the activated skills according to their measured degradation. Algorithm 1 summarizes the complete stage-level procedure.

A.1 REFRESH-LEVEL BEHAVIOR REGENERATION

The generation policy is fixed within a downstream stage, whereas the prompts and generated behaviors are resampled at every maintenance refresh. Thus, different refreshes can expose the evolving model to different valid realizations of the same foundation skill without maintaining a persistent behavior buffer. Each verifier-accepted response is consumed only in the refresh in which it is generated. When the number of accepted responses is smaller than the allocated budget $b_{t,r,k}$, the unfilled portion of the budget is left unused; accepted responses are not duplicated to meet the nominal budget.

A.2 DATA SEPARATION AND VERIFICATION

Foundation-skill monitoring, behavior generation, and downstream adaptation use disjoint roles of the available data. The monitoring subset $\mathcal{V}_k$ is fixed for each skill and is used only to estimate its functionality during continual adaptation; it does not provide prompts or optimization targets. Foundation prompts used for behavior generation are drawn separately from the monitoring subset. Their canonical annotations are accessible only to the skill-specific verifier, which determines whether a generated response is valid, and are never used directly as maintenance targets. Throughout the continual stream, examples from previous downstream VQA tasks $\mathcal{D}_{<t}$ are neither stored nor replayed.

B EXPERIMENTAL CONFIGURATION

B.1 BENCHMARK AND TRAINING SETUP

We evaluate SCALE on VQA-v3 from UCo-VQA under both the Standard Splits (SS) and Proposed Splits (PS). All continual-learning experiments start from the Molmo2-4B step2000 checkpoint and use the same eight-stage downstream stream: *Location, Commonsense, Type, Action, Color, Count,*

Algorithm 1 SCALE Continual Adaptation at Stage t

Require: Current-task data $\mathcal{D}_t$; foundation-skill monitoring sets $\{\mathcal{V}_k\}_{k=1}^{K}$; foundation-skill data and skill-specific verifiers; maintenance budget B; activation threshold τ_{ref}; EMA coefficient β; skill-loss weight λ_{skill}

```
Store the checkpoint before learning D_t as the stage-entry generation policy
for each maintenance refresh r do
    Evaluate the current model on each monitoring set V_k and update the monitored functionality using EMA coefficient β
    Measure the functionality degradation of each foundation skill as defined in the main paper
    Activate skills whose functionality degradation exceeds τ_ref
    Allocate the fixed maintenance budget B across the activated skills, yielding b_{t,r,k} for each activated skill k
    for each activated skill k do
        Sample foundation prompts for skill k
        Generate candidate behaviors using the fixed stage-entry generation policy
        Retain at most b_{t,r,k} verifier-accepted behaviors
    end for
    Distribute the accepted behaviors across the optimization updates of the current refresh without reuse
    for each optimization update do
        Sample the next current-task mini-batch from D_t
        Compute the current-task objective
        Compute the foundation-skill maintenance objective using the assigned verified behaviors
        Update the current model using the task-priority update with λ_skill
    end for
end for
```

Ensure: Updated model after adaptation to stage t

Recognition, and *Judge*. At stage t, training uses only the current downstream dataset $\mathcal{D}_t$; no example from $\mathcal{D}_{<t}$ is retained for subsequent stages. Unless otherwise stated, the task order, initialization, optimization schedule, and maintenance configuration are shared across methods and experiments.

B.2 Optimization Configuration

We adapt Molmo2-4B using LoRA while keeping the pretrained backbone frozen. Each downstream stage contains eight maintenance refreshes, with four optimization updates per refresh. A refresh uses 128 current-task samples and a global batch size of 32. The total foundation-maintenance budget is fixed to $B = 32$ per refresh and is dynamically distributed across activated skills. Importantly, B is independent of the number of downstream tasks already observed, so the maintenance budget does not grow with the continual-learning horizon. The remaining optimization and maintenance hyperparameters are reported in Table 6.

B.3 Foundation-Skill Construction

The foundation-skill repertoire is constructed from public data sources associated with the Molmo2 pretraining mixture. We select eight skills that cover visual grounding, counting, visual attributes, language reasoning, and visual commonsense. The repertoire is defined independently of the downstream VQA task partition: a foundation skill is treated as a reusable capability inherited from pretraining rather than as a previous downstream task to be rehearsed. For every skill, its data are separated according to their role in SCALE: monitoring examples measure functionality, generation prompts elicit candidate behaviors, and canonical annotations are used only by the corresponding verifier. Table 7 summarizes the data source, task form, and verification criterion for each skill.

C Complete Experimental Results

This section reports the complete results underlying the main comparisons in the paper. For each method included in the main comparison, we report the full task-by-stage evaluation matrix over the eight downstream VQA tasks, together with the stage-wise average accuracy (AA) and stage-wise forgetting. Joint Training is reported by its single joint-evaluation row because it serves as an

Table 6: Optimization and continual-learning configuration used for SCALE.

Configuration	Value
Base VLM	Molmo2-4B
Initialization checkpoint	Molmo2-4B step2000
Adaptation method	LoRA
Frozen parameters	Pretrained backbone
Benchmark	VQA-v3 from UCo-VQA
Evaluation settings	SS and PS
Number of downstream stages	8
Refreshes per stage	8
Optimization updates per refresh	4
Current-task samples per refresh	128
Global batch size	32
Maintenance budget B	32
Skill-loss weight λ_{skill}	0.1
EMA coefficient β	0.5
Activation threshold τ_{ref}	0.05
Point verification threshold	0.75
Historical downstream replay	None

Table 7: Construction of the foundation-skill repertoire from data sources associated with Molmo2 pretraining. Each skill is defined independently of downstream VQA task identities.

Skill	Source / Task Form	Verification
Point	PixMo: PixMo-Points and CoSyn-Point; coordinate and point-set grounding	Point F1
Count	PixMo: PixMo-Count; integer short-answer counting	Exact count accuracy
Attribute/Color	Molmo2 visual pretraining data: PixMo caption QA and COCO/VQA color QA	Normalized answer matching
NLI	FLAN-v2: nli_3way; semantic reasoning	Task-specific accuracy
Commonsense Bool	FLAN-v2: bool; semantic reasoning	Task-specific accuracy
Commonsense MCQ	FLAN-v2: mcq; semantic reasoning	Task-specific accuracy
Sensemaking	FLAN-v2: sentence_ab; semantic reasoning	Task-specific accuracy
Visual Commonsense	Molmo2 visual-commonsense data; scene, action, and relation commonsense	Normalized deterministic answer accuracy

upper reference and does not follow the sequential stage-wise protocol. Unobserved future tasks are marked by "–". All values are reported in percentage (%).

C.1 Main Comparison on SS

Tables 8–15 report the complete results for the methods compared in the main paper under the Standard Splits (SS). Each continual-learning method is reported separately to expose its full stage-wise trajectory, while Joint Training is shown as the upper reference.

C.2 Main Comparison on PS

Tables 16–23 report the corresponding complete results under the Proposed Splits (PS), using the same task order and evaluation protocol as in SS. Each continual-learning method is reported separately, while Joint Training is shown as the upper reference.

Table 8: Complete results of Joint Training (Upper Ref.) on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
Joint	59.07	92.38	29.24	69.52	65.21	65.56	33.69	90.57	63.16	–

Table 9: Complete stage-wise results of MaDQ-QReplay on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	58.58	–	–	–	–	–	–	–	58.58	0.00
S2: Commonsense	48.63	90.22	–	–	–	–	–	–	69.43	9.94
S3: Type	57.05	90.62	24.63	–	–	–	–	–	57.43	0.76
S4: Action	48.73	86.44	28.42	66.08	–	–	–	–	57.42	4.67
S5: Color	54.59	89.25	34.30	69.12	77.43	–	–	–	64.94	1.34
S6: Count	50.93	73.68	29.69	69.21	53.12	53.80	–	–	55.07	10.70
S7: Recognition	51.88	64.79	30.24	64.77	75.90	57.31	34.60	–	54.21	7.09
S8: Judge	55.94	79.10	31.30	68.05	75.93	61.92	35.07	90.47	62.22	2.83

C.3 Policy-Design Ablation

Tables 24–28 and Tables 29–33 report the complete stage-wise results underlying the policy-design ablation in the main paper under SS and PS, respectively. All settings use the same downstream SFT procedure and differ only in how foundation-skill supervision is introduced or regenerated. **No PT** performs only continual SFT, **Direct PT Replay** directly replays fixed PT targets, **Latest** regenerates behaviors from the most recent checkpoint at each refresh, and **Stale** generates behaviors once from the stage-entry policy and reuses them throughout the stage. SCALE uses the same fixed stage-entry policy within a stage but resamples prompts and regenerates verifier-accepted behaviors at every refresh.

C.4 Seed and Task-Order Sensitivity

We evaluate the sensitivity of SCALE to random seeds and downstream task order under SS. The seed analysis uses the official task order, *Location* → *Commonsense* → *Type* → *Action* → *Color* → *Count* → *Recognition* → *Judge*, with seeds 96817, 49048, and 96093. The task-order analysis fixes seed 96817 and compares the official order with two alternative orders: Order B, *Location* → *Type* → *Commonsense* → *Action* → *Color* → *Count* → *Recognition* → *Judge*, and Order C, *Location* → *Commonsense* → *Type* → *Color* → *Action* → *Count* → *Recognition* → *Judge*. Tables 34–38 report the complete stage-wise trajectories for all five runs. All other training and maintenance configurations are kept unchanged.

D Limitations

The current foundation-skill repertoire is organized into eight predefined skill categories, leaving open how foundation capabilities should be structured for continual adaptation. Future work could investigate finer-grained or hierarchical subskill decompositions, the appropriate granularity and overlap among subskills, and whether the skill repertoire should be expanded, merged, or reorganized as downstream demands evolve. Relatedly, the relative maintenance proportions assigned to different skills and subskills could be adapted online rather than determined solely by the current allocation rule.

Several algorithmic choices in SCALE also admit further study, including how long the stage-entry generation policy should remain fixed, the frequency of skill monitoring and behavior regeneration, the smoothing and activation rules for drift detection, and the interaction between skill allocation and task-priority optimization. Jointly adapting these components to the observed continual-learning dynamics may provide a more flexible alternative to using a single configuration throughout the continual stream.

Finally, SCALE maintains foundation skills under a fixed total budget that does not grow with downstream task-history length. While this design controls the maintenance cost in the current setting, its scalability should be further validated on substantially longer continual streams, larger

Table 10: Complete stage-wise results of GaB on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	58.58	–	–	–	–	–	–	–	58.58	0.00
S2: Commonsense	54.40	92.31	–	–	–	–	–	–	73.36	4.17
S3: Type	41.15	90.82	28.63	–	–	–	–	–	53.54	9.46
S4: Action	40.64	91.96	28.75	69.20	–	–	–	–	57.64	6.10
S5: Color	54.66	91.03	26.72	67.06	77.02	–	–	–	63.30	2.34
S6: Count	54.05	91.37	21.97	68.83	51.09	73.39	–	–	60.12	7.71
S7: Recognition	53.03	91.18	30.77	69.43	14.81	68.26	34.11	–	51.66	12.34
S8: Judge	48.58	91.21	30.23	69.19	43.72	47.63	33.32	90.62	56.81	10.25

Table 11: Complete stage-wise results of DGG on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	58.58	–	–	–	–	–	–	–	58.58	0.00
S2: Commonsense	58.09	90.68	–	–	–	–	–	–	74.38	0.48
S3: Type	58.39	90.33	23.95	–	–	–	–	–	57.55	0.27
S4: Action	56.72	89.77	25.15	65.65	–	–	–	–	59.32	0.92
S5: Color	56.50	88.61	27.32	67.27	69.93	–	–	–	61.93	1.04
S6: Count	55.95	88.46	28.58	66.77	5.68	27.43	–	–	45.48	13.92
S7: Recognition	56.37	89.01	25.37	53.35	7.98	28.76	32.28	–	41.87	13.83
S8: Judge	57.32	89.00	27.74	67.69	0.94	0.47	32.63	90.02	45.73	14.44

skill repertoires, and more diverse downstream task collections. Such evaluations could clarify whether a fixed budget remains sufficient as the scale and diversity of continual learning increase, and characterize the trade-off between maintenance budget and long-term retention at larger scales.

Table 12: Complete stage-wise results of CL-MoE on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	58.30	–	–	–	–	–	–	–	58.30	0.00
S2: Commonsense	57.83	75.86	–	–	–	–	–	–	66.85	0.47
S3: Type	58.36	85.35	23.44	–	–	–	–	–	55.72	0.00
S4: Action	58.87	88.91	23.48	52.08	–	–	–	–	55.84	0.00
S5: Color	58.48	88.07	23.85	57.45	0.10	–	–	–	45.59	0.31
S6: Count	58.22	87.91	23.83	57.67	0.19	0.90	–	–	38.12	0.33
S7: Recognition	57.54	87.80	23.86	59.65	0.26	1.03	31.38	–	37.36	0.41
S8: Judge	57.85	88.03	23.95	64.05	0.20	0.72	31.68	83.70	43.77	0.32

Table 13: Complete stage-wise results of MoELoRA on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	59.05	–	–	–	–	–	–	–	59.05	0.00
S2: Commonsense	57.41	91.68	–	–	–	–	–	–	74.54	1.64
S3: Type	57.82	91.24	24.19	–	–	–	–	–	57.75	0.83
S4: Action	58.13	91.56	29.98	70.10	–	–	–	–	62.44	0.34
S5: Color	57.61	91.43	28.51	69.83	76.37	–	–	–	64.75	0.86
S6: Count	57.88	92.02	28.23	70.30	7.01	51.52	–	–	51.16	14.46
S7: Recognition	52.88	92.14	27.45	70.12	18.97	59.51	37.97	–	51.29	11.05
S8: Judge	55.81	91.72	34.04	70.70	9.19	26.48	35.07	90.32	51.67	15.25

Table 14: Complete stage-wise results of D-MoLE on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	57.64	–	–	–	–	–	–	–	57.64	0.00
S2: Commonsense	60.06	92.13	–	–	–	–	–	–	76.09	0.00
S3: Type	60.06	92.13	24.95	–	–	–	–	–	59.04	0.00
S4: Action	60.17	71.29	25.02	59.97	–	–	–	–	54.11	6.95
S5: Color	60.17	71.29	25.02	59.97	0.15	–	–	–	43.32	5.21
S6: Count	60.17	71.29	25.02	59.97	0.15	0.68	–	–	36.21	4.17
S7: Recognition	60.17	71.29	25.02	59.97	0.15	0.68	31.88	–	35.59	3.47
S8: Judge	60.17	83.88	25.01	63.98	0.06	0.50	32.24	87.69	44.19	1.22

Table 15: Complete stage-wise results of SCALE on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	59.27	–	–	–	–	–	–	–	59.27	0.00
S2: Commonsense	57.45	91.56	–	–	–	–	–	–	74.51	1.82
S3: Type	57.52	91.60	25.59	–	–	–	–	–	58.24	0.88
S4: Action	57.15	92.24	30.47	68.92	–	–	–	–	62.19	0.71
S5: Color	58.05	91.65	26.65	71.02	76.22	–	–	–	64.72	1.41
S6: Count	58.03	92.53	24.49	69.96	28.44	69.05	–	–	57.08	11.21
S7: Recognition	57.46	91.92	28.64	70.33	68.74	65.26	38.81	–	60.17	2.70
S8: Judge	56.05	92.07	33.46	68.96	67.94	63.23	35.78	90.74	63.53	3.27

Table 16: Complete results of Joint Training (Upper Ref.) on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
Joint	61.06	16.42	24.89	23.40	35.33	0.63	30.96	21.97	26.83	–

Table 17: Complete stage-wise results of MaDQ-QReplay on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	62.42	–	–	–	–	–	–	–	62.42	0.00
S2: Commonsense	47.18	44.68	–	–	–	–	–	–	45.93	15.24
S3: Type	50.84	44.46	29.40	–	–	–	–	–	41.57	5.90
S4: Action	55.43	1.19	30.50	29.17	–	–	–	–	29.07	16.83
S5: Color	45.71	0.93	35.50	28.35	79.79	–	–	–	38.06	15.32
S6: Count	24.68	0.66	21.91	14.17	58.78	45.31	–	–	27.59	26.27
S7: Recognition	60.85	1.22	24.38	17.42	10.12	8.68	31.14	–	21.97	29.04
S8: Judge	58.97	1.15	25.47	7.75	0.20	0.59	30.18	46.54	21.36	29.10

Table 18: Complete stage-wise results of GaB on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	62.40	–	–	–	–	–	–	–	62.40	0.00
S2: Commonsense	45.04	85.65	–	–	–	–	–	–	65.34	17.36
S3: Type	35.07	1.38	33.94	–	–	–	–	–	23.47	55.80
S4: Action	21.17	1.21	30.91	65.21	–	–	–	–	29.63	42.90
S5: Color	33.99	1.28	35.51	8.55	53.67	–	–	–	26.60	42.36
S6: Count	50.28	1.20	25.14	8.04	0.15	62.02	–	–	24.47	43.53
S7: Recognition	43.09	1.17	23.84	8.35	0.43	0.60	34.00	–	15.93	47.83
S8: Judge	52.92	1.22	25.15	8.23	0.06	0.61	31.11	16.15	16.93	39.88

Table 19: Complete stage-wise results of DGG on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	62.42	–	–	–	–	–	–	–	62.42	0.00
S2: Commonsense	23.49	85.57	–	–	–	–	–	–	54.53	38.93
S3: Type	25.76	1.44	30.84	–	–	–	–	–	19.35	60.39
S4: Action	38.11	1.31	29.45	56.79	–	–	–	–	31.42	36.65
S5: Color	2.44	0.30	18.83	5.53	79.86	–	–	–	21.39	52.13
S6: Count	0.15	0.11	17.63	14.86	27.82	61.33	–	–	20.32	50.98
S7: Recognition	49.03	2.35	24.99	9.37	47.18	22.37	31.76	–	26.72	36.92
S8: Judge	42.01	1.20	27.60	13.94	22.52	0.61	29.17	85.11	27.77	38.79

Table 20: Complete stage-wise results of CL-MoE on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	64.92	–	–	–	–	–	–	–	64.92	0.00
S2: Commonsense	62.69	1.30	–	–	–	–	–	–	31.99	2.23
S3: Type	62.34	1.29	24.14	–	–	–	–	–	29.26	1.29
S4: Action	61.79	1.31	24.02	7.64	–	–	–	–	23.69	1.08
S5: Color	61.06	1.31	24.20	7.66	0.08	–	–	–	18.86	0.97
S6: Count	61.62	1.29	24.29	7.60	0.17	0.59	–	–	15.93	0.68
S7: Recognition	61.15	1.34	24.31	7.68	0.15	0.60	29.80	–	17.86	0.63
S8: Judge	61.01	1.38	24.20	7.71	0.11	0.59	29.87	3.47	16.04	0.58

Table 21: Complete stage-wise results of MoELoRA on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	63.25	–	–	–	–	–	–	–	63.25	0.00
S2: Commonsense	12.83	56.75	–	–	–	–	–	–	34.79	50.42
S3: Type	19.11	1.38	34.09	–	–	–	–	–	18.19	49.75
S4: Action	25.42	1.24	26.66	62.33	–	–	–	–	28.91	33.59
S5: Color	0.79	0.40	13.42	43.14	79.92	–	–	–	27.54	39.66
S6: Count	0.12	0.08	18.45	14.38	35.17	59.39	–	–	21.27	45.63
S7: Recognition	34.28	1.52	21.78	25.57	66.79	49.51	34.74	–	33.46	26.04
S8: Judge	5.00	1.00	24.24	16.06	40.83	0.63	27.31	85.17	25.03	39.34

Table 22: Complete stage-wise results of D-MoLE on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	64.34	–	–	–	–	–	–	–	64.34	0.00
S2: Commonsense	63.55	43.81	–	–	–	–	–	–	53.68	0.79
S3: Type	63.55	43.81	23.85	–	–	–	–	–	43.74	0.40
S4: Action	67.36	1.17	25.21	7.71	–	–	–	–	25.36	14.22
S5: Color	67.36	1.17	25.21	7.71	0.14	–	–	–	20.32	10.66
S6: Count	67.36	1.17	25.21	7.71	0.14	0.59	–	–	17.03	8.53
S7: Recognition	67.36	1.17	25.21	7.71	0.14	0.59	30.72	–	18.99	7.11
S8: Judge	62.88	1.15	24.02	7.62	0.17	0.52	29.62	46.18	21.52	7.08

Table 23: Complete stage-wise results of SCALE on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	62.05	–	–	–	–	–	–	–	62.05	0.00
S2: Commonsense	19.01	76.78	–	–	–	–	–	–	47.89	43.05
S3: Type	25.08	1.58	34.03	–	–	–	–	–	20.23	56.09
S4: Action	33.95	1.26	36.46	61.64	–	–	–	–	33.33	34.54
S5: Color	2.57	0.52	15.56	31.80	79.52	–	–	–	25.99	46.62
S6: Count	0.18	0.12	17.58	16.72	38.92	63.01	–	–	22.75	48.59
S7: Recognition	46.48	1.29	24.70	51.50	48.56	42.89	33.60	–	35.57	27.34
S8: Judge	19.89	1.11	30.26	17.31	49.23	0.69	28.48	86.51	29.19	38.01

Table 24: Complete stage-wise results of No PT on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	58.58	–	–	–	–	–	–	–	58.58	0.00
S2: Commonsense	57.22	91.37	–	–	–	–	–	–	74.30	1.35
S3: Type	57.82	91.24	24.38	–	–	–	–	–	57.81	0.44
S4: Action	56.62	91.98	31.26	70.81	–	–	–	–	62.67	0.65
S5: Color	56.94	92.23	30.87	72.65	76.70	–	–	–	65.88	0.51
S6: Count	57.05	92.07	22.58	68.66	7.66	63.59	–	–	51.94	16.68
S7: Recognition	53.97	91.80	24.05	69.87	22.66	71.28	39.10	–	53.25	11.51
S8: Judge	55.83	91.04	31.86	70.63	19.00	42.82	36.89	90.74	54.85	13.48

Table 25: Complete stage-wise results of Direct PT Replay on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	59.17	–	–	–	–	–	–	–	59.17	0.00
S2: Commonsense	57.34	92.01	–	–	–	–	–	–	74.67	1.84
S3: Type	57.09	91.70	24.70	–	–	–	–	–	57.83	1.19
S4: Action	56.70	92.15	29.09	68.76	–	–	–	–	61.67	0.83
S5: Color	56.81	92.05	28.59	69.70	73.11	–	–	–	64.05	0.74
S6: Count	55.68	92.31	26.74	70.17	13.30	62.94	–	–	53.52	13.13
S7: Recognition	56.87	92.21	24.48	70.07	19.36	71.80	39.20	–	53.43	10.15
S8: Judge	56.75	91.72	31.07	69.65	19.02	50.94	35.74	90.41	55.66	11.71

Table 26: Complete stage-wise results of Latest on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	59.20	–	–	–	–	–	–	–	59.20	0.00
S2: Commonsense	57.91	91.71	–	–	–	–	–	–	74.81	1.30
S3: Type	58.25	91.03	24.99	–	–	–	–	–	58.09	0.82
S4: Action	57.05	90.57	32.83	70.78	–	–	–	–	62.81	1.10
S5: Color	56.95	90.19	24.99	69.99	76.18	–	–	–	63.66	3.10
S6: Count	56.85	91.08	25.53	69.30	10.55	68.77	–	–	53.68	15.48
S7: Recognition	56.47	89.55	26.92	69.94	34.60	69.65	38.51	–	55.09	8.87
S8: Judge	57.26	89.89	33.82	69.50	68.16	58.45	36.13	91.19	63.05	3.81

Table 27: Complete stage-wise results of Stale on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	58.95	–	–	–	–	–	–	–	58.95	0.00
S2: Commonsense	57.76	91.84	–	–	–	–	–	–	74.80	1.18
S3: Type	58.05	92.26	24.73	–	–	–	–	–	58.35	0.45
S4: Action	56.72	92.66	29.55	70.61	–	–	–	–	62.38	0.74
S5: Color	56.58	91.30	26.40	69.19	76.01	–	–	–	63.90	2.07
S6: Count	56.07	92.36	27.10	68.91	8.80	36.34	–	–	48.26	14.91
S7: Recognition	56.87	92.75	27.30	70.04	17.25	70.26	38.75	–	53.32	10.61
S8: Judge	56.44	92.59	30.57	69.14	47.63	49.05	36.12	90.83	59.05	8.05

Table 28: Complete stage-wise results of SCALE on VQA-v3 under SS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	59.27	–	–	–	–	–	–	–	59.27	0.00
S2: Commonsense	57.45	91.56	–	–	–	–	–	–	74.51	1.82
S3: Type	57.52	91.60	25.59	–	–	–	–	–	58.24	0.88
S4: Action	57.15	92.24	30.47	68.92	–	–	–	–	62.19	0.71
S5: Color	58.05	91.65	26.65	71.02	76.22	–	–	–	64.72	1.41
S6: Count	58.03	92.53	24.49	69.96	28.44	69.05	–	–	57.08	11.21
S7: Recognition	57.46	91.92	28.64	70.33	68.74	65.26	38.81	–	60.17	2.70
S8: Judge	56.05	92.07	33.46	68.96	67.94	63.23	35.78	90.74	63.53	3.27

Table 29: Complete stage-wise results of No PT on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	62.42	–	–	–	–	–	–	–	62.42	0.00
S2: Commonsense	13.73	75.46	–	–	–	–	–	–	44.60	48.69
S3: Type	17.63	1.44	35.66	–	–	–	–	–	18.24	59.41
S4: Action	29.96	1.29	36.59	60.85	–	–	–	–	32.17	35.55
S5: Color	8.92	0.58	20.65	17.60	79.56	–	–	–	25.46	46.90
S6: Count	0.16	0.05	18.56	15.87	27.96	61.58	–	–	20.70	50.46
S7: Recognition	45.77	1.27	24.74	19.39	46.37	43.45	33.37	–	30.62	32.58
S8: Judge	14.47	0.97	28.57	17.74	20.43	0.61	24.46	85.08	24.04	43.23

Table 30: Complete stage-wise results of Direct PT Replay on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	62.58	–	–	–	–	–	–	–	62.58	0.00
S2: Commonsense	28.71	58.06	–	–	–	–	–	–	43.39	33.87
S3: Type	26.03	1.48	33.35	–	–	–	–	–	20.29	46.56
S4: Action	35.36	1.27	32.76	61.64	–	–	–	–	32.76	28.20
S5: Color	6.63	0.86	25.28	56.58	78.97	–	–	–	33.66	31.57
S6: Count	0.16	0.15	18.40	16.36	29.44	62.75	–	–	21.21	46.02
S7: Recognition	48.53	1.30	24.84	51.56	25.90	45.51	33.25	–	32.98	26.62
S8: Judge	17.11	1.20	30.82	17.41	19.63	0.60	24.10	85.86	24.59	39.96

Table 31: Complete stage-wise results of Latest on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	62.42	–	–	–	–	–	–	–	62.42	0.00
S2: Commonsense	14.02	82.27	–	–	–	–	–	–	48.15	48.40
S3: Type	21.71	1.57	34.96	–	–	–	–	–	19.41	60.71
S4: Action	33.97	1.28	35.33	60.88	–	–	–	–	32.87	36.48
S5: Color	9.51	0.77	21.39	25.40	79.58	–	–	–	27.33	45.96
S6: Count	0.26	0.18	19.48	16.51	25.56	61.28	–	–	20.54	51.70
S7: Recognition	44.88	1.25	24.29	21.87	43.79	42.02	33.14	–	30.18	33.95
S8: Judge	7.21	1.11	29.05	17.15	41.56	0.61	24.26	86.07	25.88	41.99

Table 32: Complete stage-wise results of Stale on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	62.65	–	–	–	–	–	–	–	62.65	0.00
S2: Commonsense	21.90	46.80	–	–	–	–	–	–	34.35	40.75
S3: Type	27.65	1.48	34.02	–	–	–	–	–	21.05	40.16
S4: Action	38.09	1.24	31.50	61.41	–	–	–	–	33.06	24.22
S5: Color	4.22	0.78	19.22	52.16	79.14	–	–	–	31.10	32.12
S6: Count	0.26	0.11	17.42	15.28	43.89	63.68	–	–	23.44	41.41
S7: Recognition	47.85	1.28	24.91	52.15	47.96	42.11	33.56	–	35.69	21.91
S8: Judge	27.53	1.25	32.12	12.07	48.86	0.61	27.12	79.92	28.69	33.10

Table 33: Complete stage-wise results of SCALE on VQA-v3 under PS.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	62.05	–	–	–	–	–	–	–	62.05	0.00
S2: Commonsense	19.01	76.78	–	–	–	–	–	–	47.89	43.05
S3: Type	25.08	1.58	34.03	–	–	–	–	–	20.23	56.09
S4: Action	33.95	1.26	36.46	61.64	–	–	–	–	33.33	34.54
S5: Color	2.57	0.52	15.56	31.80	79.52	–	–	–	25.99	46.62
S6: Count	0.18	0.12	17.58	16.72	38.92	63.01	–	–	22.75	48.59
S7: Recognition	46.48	1.29	24.70	51.50	48.56	42.89	33.60	–	35.57	27.34
S8: Judge	19.89	1.11	30.26	17.31	49.23	0.69	28.48	86.51	29.19	38.01

Table 34: Complete stage-wise results of SCALE under Official order, seed 96817.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	59.27	–	–	–	–	–	–	–	59.27	0.00
S2: Commonsense	57.45	91.56	–	–	–	–	–	–	74.51	1.82
S3: Type	57.52	91.60	25.59	–	–	–	–	–	58.24	0.88
S4: Action	57.15	92.24	30.47	68.92	–	–	–	–	62.19	0.71
S5: Color	58.05	91.65	26.65	71.02	76.22	–	–	–	64.72	1.41
S6: Count	58.03	92.53	24.49	69.96	28.44	69.05	–	–	57.08	11.21
S7: Recognition	57.46	91.92	28.64	70.33	68.74	65.26	38.81	–	60.17	2.70
S8: Judge	56.05	92.07	33.46	68.96	67.94	63.23	35.78	90.74	63.53	3.27

Table 35: Complete stage-wise results of SCALE under Official order, seed 49048.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	59.72	–	–	–	–	–	–	–	59.72	0.00
S2: Commonsense	57.93	91.74	–	–	–	–	–	–	74.84	1.78
S3: Type	57.54	91.94	24.87	–	–	–	–	–	58.12	1.09
S4: Action	57.62	91.48	30.78	69.66	–	–	–	–	62.39	0.85
S5: Color	57.05	91.16	25.67	68.64	76.70	–	–	–	63.84	2.39
S6: Count	56.40	91.08	25.36	69.80	48.17	46.23	–	–	56.17	7.63
S7: Recognition	53.29	90.59	27.65	70.14	69.21	69.84	38.87	–	59.94	3.06
S8: Judge	53.36	91.96	33.29	69.09	71.84	52.71	37.16	90.83	62.53	4.45

Table 36: Complete stage-wise results of SCALE under Official order, seed 96093.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	59.33	–	–	–	–	–	–	–	59.33	0.00
S2: Commonsense	57.86	91.74	–	–	–	–	–	–	74.80	1.47
S3: Type	57.61	91.46	25.59	–	–	–	–	–	58.22	1.00
S4: Action	56.25	92.77	30.03	70.93	–	–	–	–	62.50	1.03
S5: Color	56.91	91.70	30.50	71.15	71.13	–	–	–	64.28	0.87
S6: Count	56.78	91.81	27.59	69.09	32.13	71.03	–	–	58.07	9.50
S7: Recognition	56.65	91.81	28.56	70.70	58.52	67.23	38.06	–	58.79	3.74
S8: Judge	56.75	90.81	30.69	68.84	65.55	60.04	34.53	90.97	62.27	3.85

Table 37: Complete stage-wise results of SCALE under Order B, seed 96817.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	58.40	–	–	–	–	–	–	–	58.40	0.00
S2: Type	58.06	–	24.39	–	–	–	–	–	41.23	0.34
S3: Commonsense	59.07	91.62	25.76	–	–	–	–	–	58.82	0.00
S4: Action	56.18	91.45	28.68	71.16	–	–	–	–	61.87	1.02
S5: Color	55.71	90.70	23.29	68.68	76.80	–	–	–	63.04	3.04
S6: Count	55.37	91.17	22.68	68.65	25.62	69.57	–	–	55.51	12.77
S7: Recognition	53.15	90.82	24.59	69.76	69.41	70.59	39.06	–	59.63	3.27
S8: Judge	57.08	91.49	30.99	69.27	73.67	60.91	37.14	90.93	63.94	2.68

Table 38: Complete stage-wise results of SCALE under Order C, seed 96817.

Stage	Loc.	Comm.	Type	Act.	Color	Count	Recog.	Judge	AA↑	Fgt.↓
S1: Location	58.40	–	–	–	–	–	–	–	58.40	0.00
S2: Commonsense	57.19	91.74	–	–	–	–	–	–	74.47	1.21
S3: Type	57.22	91.54	25.12	–	–	–	–	–	57.96	0.69
S4: Color	57.69	91.63	27.96	–	75.89	–	–	–	63.29	0.28
S5: Action	58.29	92.21	28.32	68.26	67.76	–	–	–	62.97	2.06
S6: Count	57.98	93.00	30.39	69.02	12.80	67.03	–	–	55.04	12.70
S7: Recognition	57.24	93.01	27.32	71.75	48.29	72.91	38.18	–	58.39	5.31
S8: Judge	56.58	92.44	32.00	69.65	58.77	68.38	36.04	90.70	63.07	4.04