PACE: PROBABILISTIC EXPERTS ASSIGNMENT FOR MULTIMODAL CONTINUAL INSTRUCTION TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal continual instruction tuning requires models to acquire new capabilities while retaining previously learned ones. Expert-based methods mitigate catastrophic forgetting through parameter isolation, but expert allocation often relies on heuristic or method-specific criteria, while inference-time selection requires a separate routing mechanism. We propose **Probabilistic Assignment for Continual Experts (PACE)**, a router-free method that provides a probabilistic mechanism for selective expert sharing, built from a Chinese Restaurant Process prior and closed-form parametric empirical-Bayes evidence in a frozen multimodal feature space. Before adaptation, PACE determines whether an arriving task should reuse an existing LoRA expert or create a new one, after which only the assigned expert is updated. At inference, the same task statistics enable training-free selection of a single expert, eliminating routing drift and expert interference. We further provide a theoretical analysis of expert sharing, showing that cross-task interference is confined to tasks sharing the same expert and establishing sufficient conditions under which sharing improves generalization over task-wise isolation. Experiments across four continual multimodal benchmarks, CoIN, UCIT, TriGap, and MLLM-CL, demonstrate strong performance and knowledge retention across diverse task sequences. Further analyses show that effective expert sharing depends critically on which tasks are grouped together: PACE preserves task-specific response behavior, avoids highly interfering assignments, and reduces computation through selective adaptation and single-expert generation.

1 INTRODUCTION

Multimodal large language models acquire new capabilities through instruction tuning on image–text–response data (Liu et al., 2024; Dai et al., 2023; Bai et al., 2025). However, in realistic settings, new tasks arrive sequentially (Chen et al., 2024), motivating multimodal continual instruction tuning (MCIT), where models must learn new capabilities while retaining previous ones (Wang et al., 2023; Luo et al., 2024; Chen et al., 2025). A central challenge is *catastrophic forgetting*, where adapting to new tasks degrades performance on earlier ones. Parameter-efficient expert-based methods mitigate forgetting by maintaining lightweight LoRA experts within a frozen backbone and restricting each task's updates to selected experts (Ge et al., 2025a; Xie et al., 2026; Shi et al., 2026; Wang et al., 2025a). Consequently, experts that are not selected for a new task can remain unchanged, reducing direct parameter interference. However, parameter isolation alone does not determine how tasks should share experts or how the appropriate expert should be recovered at inference. Therefore, expert-based MCIT methods require two coupled decisions: whether an arriving task should reuse an existing expert or create a new one, and, when task identity is unavailable, which expert should be selected for a test input. Errors in either decision can cause interference or incorrect expert access.

The first challenge is therefore **inference-time expert selection**. In conventional mixture-of-experts architectures (Jacobs et al., 1991; Fedus et al., 2022), a learned router selects or combines experts according to the current input. However, under continual training, the router itself is repeatedly updated, so assignments and mixture weights for earlier inputs may change even when their corresponding experts remain frozen. Recent methods mitigate this instability through spectral-aware router updates (Xie et al., 2026) or replace router training with similarity to multimodal anchors (Guo et al., 2025a). Nevertheless, expert interference can still introduce interference between experts that encode incompatible task-specific response behaviors: contributions from other experts may alter

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.

the generated answer or its format even when the intended expert remains available. These observations motivate a stronger alternative: remove the learned router entirely and perform training-free selection from a frozen input representation, activating exactly one expert for each query.

The second challenge is **expert allocation during continual training**. Assigning a separate expert to every task offers strong isolation, but makes capacity grow with the number of tasks. Reusing an existing expert controls this growth, but subsequent updates to that expert can modify behaviors learned by tasks already assigned to it. Existing approaches determine allocation using signals such as gradient-based sensitivity (Ge et al., 2025b) or prototype similarity with hand-designed admission criteria (Shi et al., 2026). However, such allocation criteria rely on method-specific signals or thresholds to balance expert reuse against expansion, making the decision sensitive to how similarity or compatibility is defined. Moreover, allocation is typically treated separately from the problem of selecting the appropriate expert at inference, leaving two distinct mechanisms to manage expert sharing and expert access.

To address these two challenges jointly, we introduce **Probabilistic Assignment for Continual Experts (PACE)**, a router-free method for replay-free multimodal continual instruction tuning. The key idea is to use **one frozen feature space** and **one probabilistic task model** to determine both expert allocation during training and expert selection at inference. At each task boundary, PACE summarizes the arriving task using frozen multimodal features and evaluates whether it is better explained by an existing expert group or by a new one. Inspired by the Chinese restaurant process (Blackwell & MacQueen, 1973), this reuse-or-expand decision combines group occupancy with task evidence under a conjugate Gaussian model. By analytically integrating out the model parameters, PACE obtains a closed-form sequential parametric empirical-Bayes rule that determines expert allocation before adapter optimization. After allocation, only the selected LoRA expert is updated. At inference, PACE uses the input portion of the same feature representation and stored task statistics to select a single expert through a closed-form Gaussian classification rule. Consequently, PACE removes the learned router entirely and avoids both continual routing updates and prediction-time expert interference.

Our contributions are summarized: **(1)** We propose PACE, a router-free MCIT method for *selective expert sharing* that uses a CRP prior and closed-form parametric empirical-Bayes evidence for expert allocation, and the same task statistics for training-free expert selection. **(2)** We theoretically characterize the effects of expert sharing on forgetting and generalization, and establish sufficient conditions under which sharing improves generalization over task-wise isolation. **(3)** We evaluate PACE on four continual multimodal benchmarks, showing strong performance and knowledge retention. Further analyses highlight the importance of task grouping, preservation of task-specific response behavior, and avoidance of highly interfering assignments.

2 Related Work

Multimodal Continual Instruction Tuning. Multimodal continual instruction tuning (MCIT) studies sequential adaptation of multimodal large language models while retaining capabilities acquired from earlier tasks (Chen et al., 2024). Replay-free methods commonly reduce parameter interference by restricting updates to task-specific modules. O-LoRA constrains successive low-rank updates to orthogonal subspaces (Wang et al., 2023), SEFE protects important parameters through regularized LoRA updates (Chen et al., 2025), and BranchLoRA selectively freezes previously learned LoRA branches (Zhang et al., 2025). Other approaches isolate knowledge through task-specific prompts or experts, including ModalPrompt (Zeng et al., 2025), HiDe-LLaVA (Guo et al., 2025a), and mixture-of-experts adaptations such as D-MoLE (Ge et al., 2025b). While such modularization reduces direct parameter interference, it additionally requires deciding how modules should be shared across tasks and how the appropriate module should be recovered when task identity is unavailable.

Expert Allocation and Selection. Recent expert-based methods address these problems through dynamic allocation and routing. For allocation, LLaVA-CMoE uses probe-guided expert expansion (Zhao et al., 2025), D-MoLE relies on gradient-based signals (Ge et al., 2025b), while ProtoAda (Shi et al., 2026) and CRAM (Tang et al., 2026) use prototype or centroid similarity to determine expert reuse and expansion. For inference-time selection, existing methods employ learned routers: SAME stabilizes router and expert drift during continual training (Xie et al., 2026), LLaVA-DyMoE constrains token assignments as experts are introduced (Zhao et al., 2026a), and HiDe-LLaVA per-

forms module selection using multimodal prototypes (Guo et al., 2025a). These approaches typically determine expert allocation and inference-time selection using separate criteria or mechanisms. In contrast, PACE uses the same feature representation and probabilistic task model for both decisions, yielding closed-form expert allocation before adaptation and training-free hard expert selection at inference.

3 PRELIMINARIES

3.1 MULTIMODAL CONTINUAL INSTRUCTION TUNING

Multimodal continual instruction tuning (MCIT) considers a multimodal large language model (MLLM) that learns from a sequence of instruction-tuning tasks (Chen et al., 2024; Wang et al., 2023; Luo et al., 2024; Chen et al., 2025). Let $\mathcal{D}_t = \left\{ \left(x_i^{(t)}, y_i^{(t)} \right) \right\}_{i=1}^{N_t}$ $(t = 1, \ldots, T)$, denote the training set of task t, where $x_i^{(t)} = (I_i^{(t)}, q_i^{(t)})$ contains an image $I_i^{(t)}$ and an instruction or question $q_i^{(t)}$, and $y_i^{(t)} = (y_{i,1}^{(t)}, \ldots, y_{i,L_i}^{(t)})$ is the target response. An MLLM with parameters Θ is trained autoregressively by minimizing:

$$\mathcal{L}_t(\Theta) = -\frac{1}{N_t} \sum_{i=1}^{N_t} \sum_{\ell=1}^{L_i} \log p_\Theta \left(y_{i,\ell}^{(t)} \mid x_i^{(t)}, y_{i,<\ell}^{(t)} \right).$$

We consider the replay-free setting: when learning task t, data from earlier tasks are unavailable. A *task boundary* denotes the known transition from task $t-1$ to a newly arriving task t. Task boundaries are provided during training, but task identity is not available at inference (Chen et al., 2024). Thus, a continual learner must both preserve earlier capabilities and determine how an unlabeled test input should access the learned task-specific knowledge.

3.2 THE DIRICHLET PROCESS AND THE CHINESE RESTAURANT PROCESS

The number of task groups required in continual learning is generally unknown in advance. We therefore use a Dirichlet process (DP), $G \sim \mathrm{DP}(\alpha, G_0)$, which allows the number of groups to grow with the observed tasks (Ferguson, 1973). The resulting group assignment follows the Chinese Restaurant Process (CRP) (Blackwell & MacQueen, 1973). If the first $t-1$ tasks occupy K groups and group k contains n_k tasks, then

$$p(z_t = k \mid z_{<t}) = \frac{n_k}{\alpha + t - 1}, \qquad p(z_t = \text{new} \mid z_{<t}) = \frac{\alpha}{\alpha + t - 1}, \tag{1}$$

where $\alpha > 0$ controls the tendency to create new groups.

Given a task representation r_t, a DP mixture combines this prior with predictive evidence (Neal, 2000):

$$p(z_t = k \mid r_t, \mathcal{H}_{t-1}) \propto n_k \, p(r_t \mid \mathcal{H}_k), \qquad p(z_t = \text{new} \mid r_t, \mathcal{H}_{t-1}) \propto \alpha \, p(r_t \mid G_0), \tag{2}$$

where $\mathcal{H}_k$ contains the observations assigned to group k. PACE assigns each task once, to the maximizer of Eq. 2 given the earlier assignments. This is the sequential greedy approximation to DP-mixture inference of Wang & Dunson (2011): it does not maximize the joint posterior of $z_{1:T}$. With a conjugate likelihood, the predictive terms are available in closed form, so expert reuse or expansion is decided without training.

4 PROBABILISTIC ASSIGNMENT FOR CONTINUAL EXPERTS

PACE contains a frozen backbone Θ_0 and a set of LoRA experts $\{\theta_k\}_{k=1}^{K_t}$ that **grows dynamically** as tasks arrive. Each task receives a fixed assignment z_t to one expert. The same collection of task statistics is used for both **expert allocation** and **inference-time expert selection**. Allocation compares existing and new groups through conjugate Bayesian evidence, whereas expert selection uses a pooled-covariance Gaussian rule on the input coordinates. Only the assigned expert is optimized. Fig. 1 provides the overview of PACE.

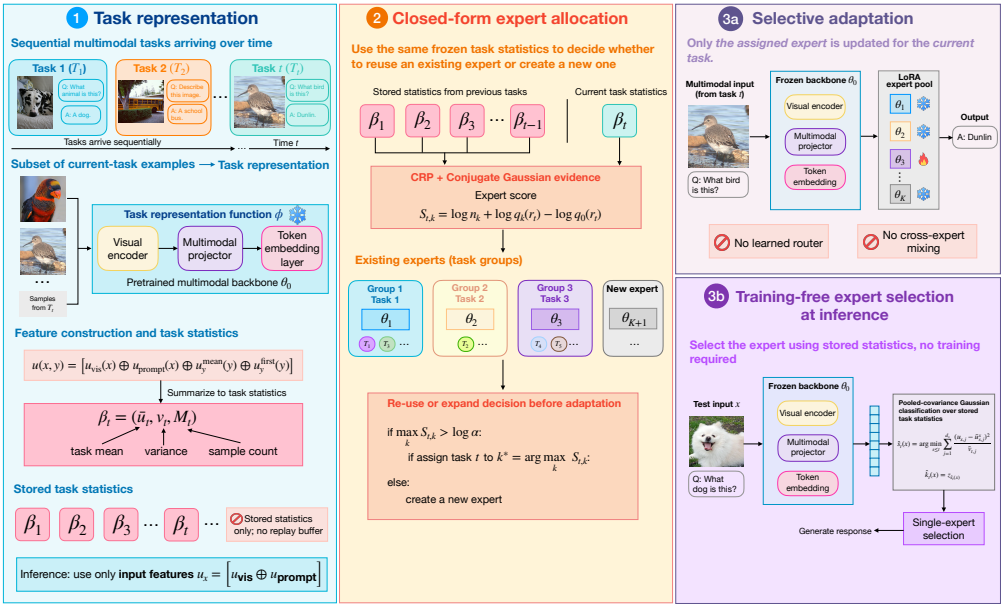

Figure 1: **Overview of PACE.** Frozen task statistics guide closed-form expert reuse or expansion before adaptation, with only the assigned LoRA expert updated. At inference, the input components of the same stored statistics enable training-free selection of a single expert, without task identity, replay data, or a learned router.

4.1 Task Representation and Stored Statistics

For each example (x, y), we extract features from the frozen visual encoder, multimodal projector, and token embedding layer:

$$u(x,y) = \left[u_{\text{vis}}(x) \oplus u_{\text{prompt}}(x) \oplus u_y^{\text{mean}}(y) \oplus u_y^{\text{first}}(y)\right] \in \mathbb{R}^d, \tag{3}$$

where the four components are the mean visual-token representation, mean prompt-token embedding, mean response-token embedding, and first response-token embedding, respectively. Each feature component is independently ℓ_2-normalized. At inference, where the response is unavailable, we use only the input features $u_x = [u_{\text{vis}} \oplus u_{\text{prompt}}] \in \mathbb{R}^{d_x}$. Importantly, u_{prompt} encodes the *task instruction*, which remains available at inference; thus expert selection does not rely on visual content alone. Consequently, tasks with similar visual distributions but different required response behaviors can still be distinguished through their instructions, while response features are used only for training-time task summarization and allocation.

Before adapting to task t, we compute features from M_t current-task examples and store $\beta_t = (\bar{u}_t, v_t, M_t)$, where $\bar{u}_t = \frac{1}{M_t}\sum_{i=1}^{M_t} u_i^{(t)}$ and $v_t = \text{Var}_{i \le M_t}[u_i^{(t)}]$. We define the task representation as $r_t = \bar{u}_t$, i.e., the mean feature vector of task t, which serves as the task-level observation for expert allocation. The stored statistics remain fixed during subsequent adaptation and require $O(Td)$ storage across T tasks, without retaining training examples.

4.2 Closed-Form Bayesian Expert Allocation

When task t arrives, let $\mathcal{G}_k = \{s < t : z_s = k\}$ denote the set of previously observed tasks assigned to expert k, with $n_k = |\mathcal{G}_k|$. We model their task means $\{\bar{u}_s : s \in \mathcal{G}_k\}$ using a diagonal Gaussian likelihood, with an independent normal–inverse-gamma (NIG) prior (Murphy, 2007) for each coordinate:

$$\begin{aligned} r_{t,j} \mid z_t = k, \mu_{kj}, \sigma_{kj}^2 &\sim \mathcal{N}(\mu_{kj}, \sigma_{kj}^2), \\ \sigma_{kj}^2 &\sim \text{InvGamma}(a_0, b_{0t,j}), \qquad \mu_{kj} \mid \sigma_{kj}^2 \sim \mathcal{N}(\mu_{0t,j}, \sigma_{kj}^2/\kappa_0). \end{aligned} \tag{4}$$

To evaluate whether task t should join an existing group or form a new one, we first construct the base distribution $G_{0,t}$ using only previous tasks. We set $\mu_{0,t}$ to the average of the task means observed

before task t, and $b_{0,t}$ to their unbiased coordinate-wise variance. At $t = 2$, only one previous task is available, making the between-task variance unavailable. Therefore, we use its within-task variance v_1 instead. We fix $a_0 = 2$, which makes $b_{0,t}$ equal to the prior expected coordinate-wise variance. The first task forms the initial group by construction. $G_{0,t}$ is thus a parametric empirical-Bayes prior (Morris, 1983; Casella, 1985). Its normal–inverse-gamma form and (κ_0, a_0) are fixed in advance, and only its location $\mu_{0,t}$ and scale $b_{0,t}$ are estimated from data, by matching the moments of the earlier task means. Unlike the fixed base measure of a single DP mixture, it changes with t.

Under the NIG model, integrating out the group-specific mean and variance (μ_k, σ_k^2) yields the closed-form posterior predictive density $q_k(r) = \prod_{j=1}^{d} \mathrm{St}_\nu\Big(r_j; \mu_{kj}, \frac{b_{kj}(\kappa_k+1)}{a_k\kappa_k}\Big)$, where $\mathrm{St}_\nu(\cdot; \mu, s^2)$ denotes a Student-t density with squared scale s^2 and $\nu = 2a_k$. The posterior parameters $(\mu_k, \kappa_k, a_k, b_k)$ are obtained by conditioning the NIG prior specified by $G_{0,t}$ on the task means $\{\bar{u}_s : s \in \mathcal{G}_k\}$. The corresponding closed-form posterior updates are given in Appendix A.3. For a new group, no previous task has been assigned to it, and its predictive density $q_0(\bar{u}_t)$ is obtained directly from $G_{0,t}$.

Combining the predictive densities with the CRP prior (Blackwell & MacQueen, 1973), we score each existing group $\mathcal{G}_k$ by

$$S_{t,k} = \log n_k + \log q_k(r_t) - \log q_0(r_t), \qquad z_t = \begin{cases} K_{t-1} + 1, & \log\alpha > \max_k S_{t,k}, \\ \arg\max_k S_{t,k}, & \text{otherwise.} \end{cases} \tag{5}$$

The score combines the predictive evidence for reusing group k with its CRP occupancy term $\log n_k$. A new expert is created when no existing group provides sufficient evidence relative to a new group. Eq. 5 is a sequential parametric empirical-Bayes MAP rule. At boundary t, z_t maximizes $p_t(z_t \mid r_{1:t}, z_{<t})$ under the CRP mixture in Eq. 2 with base measure $G_{0,t}$ and earlier assignments fixed. Normalizing by $q_0(r_t)$ and taking logarithms yields $S_{t,k}$ for existing groups and $\log\alpha$ for a new group. Since $G_{0,t}$ changes with t and previous assignments are not revisited, the rule is sequential rather than joint and can depend on task order. We fix $\log\alpha = \lambda_0 d$ before the first task arrives, where λ_0 is the expected per-coordinate score of a one-task group whose task is independent of the arriving task. It is computed from (κ_0, a_0) alone and uses no task data. Each assignment z_t therefore depends only on $\beta_1, \ldots, \beta_t$, and the number of experts is not specified in advance. Appendix A.2 gives the definition of λ_0 and a sensitivity analysis.

4.3 Selective Adaptation and Training-Free Expert Allocation

Adaptation. When a new expert is created, its LoRA parameters are initialized by averaging the existing experts; otherwise, the assigned expert is further trained. In both cases, we minimize $\mathcal{L}_t(\Theta_0, \theta_{z_t})$ while keeping the backbone and all other experts fixed. Each adapted layer computes: $h_{\text{out}} = Wh_{\text{in}} + \frac{\alpha_L}{r_{\text{total}}} B_{z_t} A_{z_t} h_{\text{in}}$, where r_{total} is the total LoRA rank allocated across expert slots and α_L is the scaling factor. All tokens are processed by the same expert, matching the single-expert selection used at inference. No learned router or expert interference is used.

Expert selection. After learning t tasks, PACE selects an expert for each input using only the stored input-feature statistics. Let $\bar{u}_s^x$ and v_s^x denote the input-feature mean and variance of task s, respectively. We first compute the pooled coordinate-wise variance across the observed tasks as $\bar{v}_{t,j} = \frac{1}{t}\sum_{s=1}^{t} v_{s,j}^x$. Given an input x, its task and corresponding expert are then identified by

$$\hat{s}_t(x) = \arg\min_{s \le t} \sum_{j=1}^{d_x} \frac{(u_{x,j} - \bar{u}_{s,j}^x)^2}{\bar{v}_{t,j}}, \qquad \hat{k}_t(x) = z_{\hat{s}_t(x)}. \tag{6}$$

Here, $\hat{s}_t(x)$ denotes the identified task and $\hat{k}_t(x)$ denotes its assigned expert. This rule corresponds to Gaussian classification with equal task priors and a shared diagonal covariance. Pooling the variance across tasks avoids unreliable task-specific variance estimates. Tasks are scored individually rather than at the group level, preserving distinct task means when multiple tasks share the same expert. Consequently, confusing two tasks assigned to the same expert does not change the selected adapter. Expert selection requires $O(td_x)$ computation per query, while expert allocation requires $O(K_{t-1}d)$ at each task boundary. Appendix C.2 describes benchmark-specific expert-selection variants.

4.4 THEORETICAL ANALYSIS

We analyze how expert sharing affects forgetting and generalization using the overparameterized linear-regression model commonly adopted in continual-learning theory (Evron et al., 2022; Lin et al., 2023; Li et al., 2025). Each expert is represented by a parameter vector $\theta_k \in \mathbb{R}^p$, where p denotes the adapter dimension. We consider $T \geq 2$ tasks, each encountered once. Before adaptation, task t is assigned to an expert z_t, and only that expert is updated. Supporting results on task statistics, allocation, and expert selection are provided in Appendix B.

Assumption 1. For each task t, let $X_t = [x_{t,1}, \ldots, x_{t,N}] \in \mathbb{R}^{p\times N}$, where $x_{t,1}, \ldots, x_{t,N}$ are i.i.d. samples from $\mathcal{N}(0, I_p)$ and $1 \leq N < p$. The task sample matrices $X_1, \ldots, X_T$ are mutually independent.

Assumption 2. For each task t, $y_t = X_t^\top \theta_t^\star$ for a fixed target $\theta_t^\star \in \mathbb{R}^p$.

Assumption 3. The frozen statistics used to determine the assignments $\{z_t\}_{t=1}^T$ are independent of the task sample matrices $\{X_t\}_{t=1}^T$. Each expert is initialized with finite second moment, and its initial state may depend only on information available before the first task assigned to that expert.

Throughout the analysis, expectations are taken over the task samples $\{X_t\}_{t=1}^T$, while the allocation statistics and the resulting assignments $\{z_t\}_{t=1}^T$ are held fixed. The independence condition in Assumption 3 is imposed for the theoretical analysis and is separate from the frozen-feature construction used by PACE. Under Assumptions 1–3, gradient descent on the assigned expert, initialized from its current state and run to interpolation, converges to the interpolating solution closest to that state:

$$\theta_{z_t}^{(t)} = \theta_{z_t}^{(t-1)} + X_t(X_t^\top X_t)^{-1}\big(y_t - X_t^\top \theta_{z_t}^{(t-1)}\big), \qquad \theta_k^{(t)} = \theta_k^{(t-1)} \ (k \neq z_t). \tag{7}$$

Let $\varrho = 1 - N/p \in (0, 1)$ and define $E_t(\theta) = \|\theta - \theta_t^\star\|_2^2$, which equals the population excess risk on task t under Assumption 1 (Shalev-Shwartz & Ben-David, 2014; Bartlett et al., 2005). Let K denote the final number of experts. For each expert k, define its assigned task set $\mathcal{G}_k = \{t \in \{1, \ldots, T\} : z_t = k\}$ with size $n_k = |\mathcal{G}_k|$, and let $\tau_k(1) < \cdots < \tau_k(n_k)$ denote the corresponding task indices in arrival order. We characterize each group by its maximum adapter distance and the expert's initialization error:

$$\Delta_k = \max_{s,q\in\mathcal{G}_k} \|\theta_s^\star - \theta_q^\star\|_2, \qquad \xi_{k,q}^2 = \mathbb{E}\|\theta_k^{(0)} - \theta_q^\star\|_2^2, \qquad \bar{\xi}_k^2 = \frac{1}{n_k}\sum_{q\in\mathcal{G}_k} \xi_{k,q}^2, \tag{8}$$

where $\theta_k^{(0)}$ denotes the initial state of expert k when it is created. Following Lin et al. (2023) and Li et al. (2025), we measure forgetting and final generalization using the expert assigned to each task:

$$F_T = \frac{1}{T-1}\sum_{t=1}^{T-1}\Big[E_t\big(\theta_{z_t}^{(T)}\big) - E_t\big(\theta_{z_t}^{(t)}\big)\Big], \qquad G_T = \frac{1}{T}\sum_{t=1}^{T} E_t\big(\theta_{z_t}^{(T)}\big). \tag{9}$$

Theorem 1. *Under Assumptions 1–3, let* $c_{ij}^{(n)} = (1-\varrho)(\varrho^{n-i} + \varrho^{n-j} - \varrho^{j-i})$ *for* $1 \leq i < j \leq n$. *Then*

$$\mathbb{E}[F_T] = \frac{1}{T-1}\sum_{k=1}^{K}\left[\sum_{j=1}^{n_k}(\varrho^{n_k} - \varrho^j)\xi_{k,\tau_k(j)}^2 + \sum_{1\leq i<j\leq n_k} c_{ij}^{(n_k)}\|\theta_{\tau_k(i)}^\star - \theta_{\tau_k(j)}^\star\|_2^2\right]. \tag{10}$$

Corollary 1. *Under the assumptions of Theorem 1,*

$$\mathbb{E}[F_T] \leq \frac{1}{T-1}\sum_{k=1}^{K} c_\varrho(n_k)\Delta_k^2, \qquad c_\varrho(n) = n - \frac{1-\varrho^n}{1-\varrho} \leq \min\left\{n-1, \frac{N}{p}\frac{n(n-1)}{2}\right\}. \tag{11}$$

Theorem 1 shows that forgetting can arise only from updates by later tasks assigned to the same expert: tasks assigned to different experts do not affect one another. The upper bound in Corollary 1 is determined by the number of tasks sharing each expert and the maximum distance Δ_k between their optimal adapters, motivating PACE to share an expert among tasks requiring similar adaptations. When $K = T$, each task is assigned its own expert and $F_T = 0$. When all tasks sharing an expert have the same optimal adapter, $\mathbb{E}[F_T] \leq 0$, since subsequent updates introduce no mismatch between their task objectives and may improve earlier tasks.

Theorem 2. *Under Assumptions 1–3,*

$$\begin{aligned}\mathbb{E}[G_T] &= \frac{1}{T}\sum_{k=1}^{K}\sum_{q\in\mathcal{G}_k}\left[\varrho^{n_k}\xi_{k,q}^2 + (1-\varrho)\sum_{i=1}^{n_k}\varrho^{n_k-i}\|\theta^\star_{\tau_k(i)} - \theta^\star_q\|_2^2\right] \\ &\le \frac{1}{T}\sum_{k=1}^{K} n_k\left[\varrho^{n_k}\bar{\xi}_k^2 + (1-\varrho^{n_k})\Delta_k^2\right].\end{aligned} \tag{12}$$

Expert sharing introduces a trade-off between reducing initialization error and increasing within-group interference. When n_k tasks share expert k, repeated updates reduce the residual initialization error by the factor ϱ^{n_k}, compared with ϱ when each task uses an isolated expert, while differences among the tasks' optimal adapters introduce an additional error controlled by Δ_k^2. For $n_k \ge 2$, comparison with n_k isolated experts having the same initialization errors shows that sharing is guaranteed to reduce the group's expected contribution to G_T whenever $\Delta_k^2 < \frac{\varrho-\varrho^{n_k}}{1-\varrho^{n_k}}\bar{\xi}_k^2$ (see Appendix B.5). This condition is sufficient because it is derived from the upper bound in Theorem 2: sharing may still improve generalization when the condition is not satisfied.

Main results. (1) Forgetting is confined to tasks sharing the same expert. Theorem 1 shows that updates from tasks assigned to different experts do not affect one another, while forgetting under expert sharing depends on the number of shared tasks and the distances between their optimal adapters. **(2) Expert sharing can improve generalization.** Theorem 2 shows that repeatedly training of a shared expert reduces residual initialization error by ϱ^{n_k}, at the cost of interference from task mismatch. These results motivate PACE to reuse experts selectively when tasks require similar adaptations, balancing reduced initialization error against forgetting and mismatch.

Table 1: **Comparison on the CoIN benchmark.** All baselines are reproduced with Qwen2.5-VL-7B-Instruct. The protocol is specified in Appendix C.4. The best and second-best results are highlighted in **bold** and underline, respectively.

Method	SciQA	TextVQA	ImageNet	GQA	VizWiz	Grounding	VQAv2	OCR-VQA	Average
Zero-shot	87.64	79.75	18.87	61.80	58.85	81.90	73.48	58.90	65.15
Joint training	94.06	83.98	96.17	65.24	66.08	84.25	73.75	76.28	79.98
FT-LoRA	94.72	81.78	59.65	60.56	45.41	67.54	71.98	**77.15**	69.85
EWC	**95.57**	85.60	24.26	62.69	63.86	81.03	72.60	52.90	67.31
O-LoRA	94.62	84.28	82.43	63.39	43.28	78.70	73.45	75.50	74.46
MoELoRA	94.25	84.42	26.17	62.82	44.02	64.66	73.67	76.30	65.79
CL-MoE	94.34	84.83	19.74	63.77	59.22	80.66	73.93	62.46	67.37
HiDe-LLaVA	80.57	71.33	96.00	60.46	45.04	79.61	62.79	65.29	70.14
SMoLoRA	93.96	84.61	34.09	62.88	44.58	71.41	73.62	**77.15**	67.79
DISCO	94.81	85.42	21.74	63.65	56.81	83.62	**74.54**	67.39	68.50
SEFE	94.81	85.06	95.13	64.12	41.15	79.96	73.84	76.77	76.36
SAME	95.19	85.31	32.17	**64.82**	58.85	78.48	72.08	69.20	69.51
PACE (Ours)	95.38	**85.73**	**96.61**	63.26	**65.34**	**83.64**	74.22	67.55	**78.97**

Table 2: **Comparison on the UCIT benchmark.** All baseline results are reproduced with LLaVA-1.5-7B. The best and second-best results are highlighted in **bold** and underline, respectively.

Method	ImageNet-R	ArxivQA	VizWiz-Cap	IconQA	CLEVR-Math	Flickr30k	Average
Zero-shot	16.80	53.87	38.51	19.90	20.53	41.87	31.91
Joint training	87.17	91.27	60.75	66.97	64.73	56.34	71.21
FT-LoRA	7.23	69.57	43.24	59.97	41.10	**55.90**	46.17
EWC	77.23	80.63	44.62	59.13	54.27	55.48	61.89
O-LoRA	81.20	92.90	31.06	56.37	47.63	49.22	59.73
MoELoRA	60.17	74.50	42.62	60.93	52.23	55.38	57.64
CL-MoE	79.13	80.67	43.38	54.23	42.87	54.16	59.07
HiDe-LLaVA	82.10	88.47	49.83	60.53	40.30	53.09	62.39
SMoLoRA	66.63	75.40	42.87	60.10	56.30	55.14	59.41
DISCO	83.27	**94.20**	42.72	64.80	**59.20**	55.31	66.58
SEFE	80.53	91.47	49.41	59.57	34.50	54.85	61.72
SAME	75.37	52.47	37.50	45.73	13.30	39.56	43.99
PACE (Ours)	**84.07**	93.60	**57.29**	**68.53**	52.63	53.02	**68.19**

5 EXPERIMENTS

Benchmarks. In the main paper, we mainly report our method on two benchmarks: **CoIN** (Chen et al., 2024), which contains 8 sequential tasks constructed from ScienceQA (Lu et al., 2022), TextVQA (Singh et al., 2019), ImageNet (Deng et al., 2009), GQA (Hudson & Manning, 2019), VizWiz (Gurari et al., 2018), VQAv2 (Goyal et al., 2017), OCR-VQA (Mishra et al., 2019), RefCOCO (Kazemzadeh et al., 2014; Mao et al., 2016), and **UCIT** (Guo et al., 2025a), which consists of 6 sequential tasks constructed from ImageNet-R (Hendrycks et al., 2021), ArXivQA (Li et al., 2024), VizWiz-Captions (Gurari et al., 2020), IconQA (Lu et al., 2021), CLEVR-Math (Lindström & Abraham, 2022), and Flickr30k (Young et al., 2014). We report results on the **MLLM-CL** and **TriGap** benchmarks in Appendix D.1-D.2. We scaled the number of tasks from 6 (in UCIT) to 8 (in CoIN) and 10 (in TriGap) to show the scalability of our work.

Baselines. We compare with MCIT baselines: FT-LoRA (Sequential fine-tuning with LoRA), EWC (Kirkpatrick et al., 2017), MoELoRA (Luo et al., 2024), CL-MoE (Huai et al., 2025), O-LoRA (Wang et al., 2023), HiDe-LLaVA (Guo et al., 2025a), DISCO (Guo et al., 2025b), SMoLoRA (Wang et al., 2025b), SEFE (Chen et al., 2025) and SAME (Xie et al., 2026).

Metrics. Following prior works (Wang et al., 2025b; Xie et al., 2026; Guo et al., 2025b), we report four metrics: average accuracy (AA), backward transfer (BWT), and forgetting (FGT). Definitions and benchmark-specific metrics appear in Appendix C.4.

5.1 BENCHMARK COMPARISON

Tabs. 1-2 report task-wise performance on CoIN and UCIT benchmarks, respectively. Zero-shot and joint training provide reference results. On CoIN, PACE achieves the best average performance of 78.97%, outperforming the strongest baseline (SEFE) by 2.62 percentage points. Notably, several baselines even fall below zero-shot performance on individual tasks, e.g., HiDe-LLaVA on SciQA, and SAME on CLEVR-Math. In contrast, PACE consistently outperforms zero-shot across *all* tasks on both benchmarks, demonstrating robust adaptation without sacrificing pretrained capabilities. Furthermore, our method also achieves the best results on the challenging benchmark - TriGap (Appendix D.1).

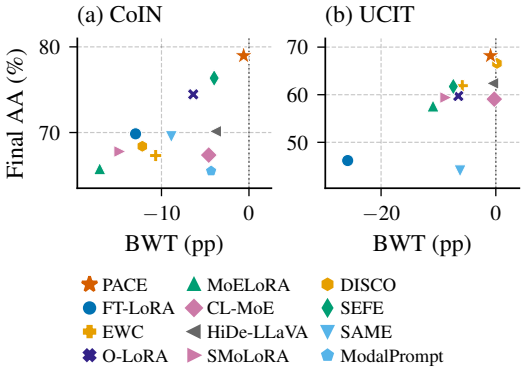

Figure 2: **Trade-off between final average accuracy (AA) and backward transfer (BWT) on CoIN and UCIT.** PACE achieves high final performance while maintaining substantially better backward transfer than most baselines.

Fig. 2 further confirms the advantage of PACE in mitigating catastrophic forgetting by comparing final average accuracy (AA) against backward transfer (BWT). PACE achieves the highest AA on both CoIN (78.97) and UCIT (68.19), with the best BWT on CoIN (-0.61) and near-zero BWT on UCIT (-0.92). Overall, PACE provides a favorable trade-off between learning new tasks and retaining previously acquired knowledge across both benchmarks.

5.2 ABLATION STUDY AND ANALYSIS

Expert Capacity. Tab. 3 shows that selective expert reuse is crucial. PACE achieves AA 78.97 with 6 experts, compared with 78.67 for the one-expert-per-task run, while using 25% fewer populated LoRA parameters. As the performance becomes saturated when close to the upper-bound (Joint), increasing the expert rank provides no improvement. Meanwhile, reducing the pool to 3 experts substantially degrades both AA and forgetting, indicating severe interference from excessive sharing.

Forgetting Root Cause. Fig. 3 decomposes forgetting into expert-update and selection effects. For PACE, the main degradation on GQA (-4.90) is entirely caused by updating its expert shared

Table 3: **Ablation of expert reuse and capacity on CoIN.** Higher AA and BWT and lower FGT are better.

Configuration	K	Expert rank	AA	BWT	FGT
One expert per task	8	24	78.67	**-0.32**	**0.44**
PACE	6	24	**78.97**	-0.61	0.71
PACE w/ higher expert rank	6	32	78.83	-0.63	0.75
PACE w/ fewer experts	3	64	69.06	-11.20	11.23

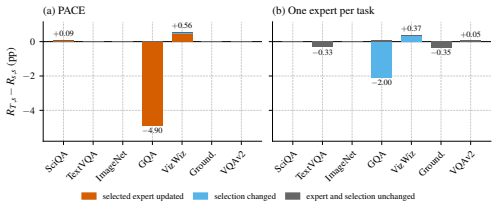

Figure 3: **Forgetting root cause analysis.** Final task changes, $R_{T,s} - R_{s,s}$, are decomposed into effects from expert updates and expert-selection changes for PACE and one expert per task.

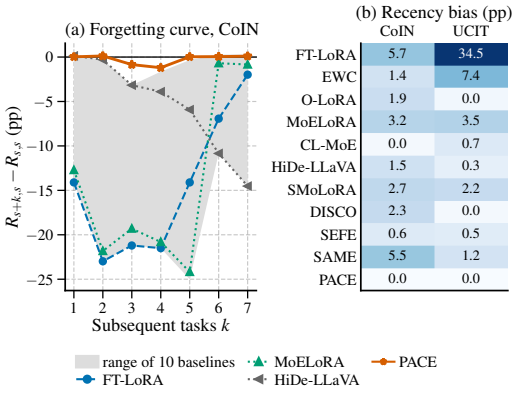

Figure 4: **Forgetting and recency bias.** (a) Mean score change $R_{s+k,s} - R_{s,s}$ after k subsequent tasks; the band shows the range of ten baselines. (b) Increase in earlier-task answers adopting the response format of the newly learned task.

with VQAv2, while isolating each task in a separate expert removes such update interference and leaves forgetting mainly due to selection changes. This empirically supports Theorem 1: parameter-update forgetting is confined to tasks that share an expert, whereas errors from inference-time expert selection constitute a separate source of degradation.

Forgetting and Recency Bias. Fig. 4 shows that PACE maintains earlier-task performance as more tasks are learned, while several baselines exhibit progressively larger degradation. PACE also shows no measurable recency bias on either CoIN or UCIT, whereas other methods increasingly produce answers in the format of recently learned tasks. These results indicate that PACE reduces both performance forgetting and undesirable transfer of recent-task response behavior.

Task Representation Ablation. We compare PACE's full input–response representation with response-only, vision-only and input-only variants. Tab. 4 shows that response information is particularly important for expert allocation. Using only the response-side features preserves the same six-expert partition as full PACE and achieves nearly identical AA (78.98 vs. 78.77), with slightly better BWT and FGT.

Table 4: **Ablation of task representations for expert allocation on CoIN.** Higher AA and BWT and lower FGT are better.

Allocation representation	Experts	Expert rank	Populated params	AA ↑	BWT ↑	FGT ↓
Full (PACE)	6	24	363.3M	78.98	−0.61	0.71
Response-only	6	24	363.3M	78.77	−0.64	0.71
Vision-only	3	64	484.4M	76.80	−2.35	2.42
Input-only	3	64	484.4M	70.02	−12.19	12.12

6 Conclusion

We introduced **Probabilistic Assignment for Continual Experts (PACE)**, a router-free method for replay-free multimodal continual instruction tuning that enables *selective expert sharing* across sequential tasks. Rather than isolating every task in a separate expert or relying on a learned router, PACE uses frozen multimodal representations and stored task statistics to decide whether an arriving task should reuse an existing expert or create a new one. The same statistics are then used for training-free, single-expert selection at inference, providing a unified mechanism for both expert sharing and expert access. Furthermore, we provide a theoretical analysis showing that expert sharing can improve generalization when tasks require similar adaptations, whereas sharing dissimilar tasks can introduce interference and forgetting. Empirically, our analyses show that effective sharing depends critically on task grouping, while PACE preserves task-specific response behavior and mitigates forgetting.

AI USE STATEMENT

Generative AI tools were used to support language editing and manuscript refinement, and to assist with implementation, debugging, and code refactoring. Generative AI tools were not used to generate synthetic datasets; develop the theoretical model or conceptual framework; formulate mathematical claims; provide critical ingredients for, or write, proofs; propose or refine hypotheses; design or provide feedback on the research methodology or experiments; clean or reformat datasets; conduct qualitative or thematic data analysis; or interpret experimental results. No other required AI-use disclosure category was applicable to this work. All AI-assisted content was reviewed and verified by the authors. AI-assisted code was manually inspected, executed, and tested before use in the reported experiments. The authors take full responsibility for the final manuscript, including its text, technical claims, proofs, code, and experimental results.

ETHICS STATEMENT

This work studies continual adaptation of multimodal large language models using existing benchmarks and pretrained models. We do not introduce new datasets or collect new human-subject data. As with other multimodal language models, the resulting systems may inherit biases, limitations, or potentially harmful behaviors from the underlying pretrained models and training data. Our method focuses on improving continual adaptation and knowledge retention, and does not explicitly address these broader risks. Therefore, we encourage appropriate evaluation and safeguards when applying such models in sensitive or real-world settings.

REPRODUCIBILITY STATEMENT

We provide detailed information to support reproduction of the proposed method and experiments throughout the paper and appendix. Sec. 4 describes the proposed PACE method, including its task representation, probabilistic expert allocation, selective adaptation, and inference-time expert selection. Appendix A.1 provides the complete algorithm and closed-form derivations, while Appendix B states the assumptions and gives proofs for the theoretical results. Appendix C details the benchmark settings, training protocols, implementation of PACE and the baselines, and evaluation metrics.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Peter L Bartlett, Olivier Bousquet, and Shahar Mendelson. Local rademacher complexities. 2005.

David Blackwell and James B MacQueen. Ferguson distributions via pólya urn schemes. *The annals of statistics*, 1(2):353–355, 1973.

George Casella. An introduction to empirical Bayes data analysis. *The American Statistician*, 39 (2):83–87, 1985.

Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *European conference on computer vision*, pp. 556–572. Springer, 2018.

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, 2024.

Jinpeng Chen, Runmin Cong, Yuzhi Zhao, Hongzheng Yang, Guangneng Hu, Horace Ip, and Sam Kwong. SEFE: Superficial and essential forgetting eliminator for multimodal continual instruction tuning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 7982–8001. PMLR, 13–19 Jul 2025. URL `https://proceedings.mlr.press/v267/chen25n.html`.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in neural information processing systems*, 36:49250–49267, 2023.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Meng Ding, Kaiyi Ji, Di Wang, and Jinhui Xu. Understanding forgetting in continual learning with linear regression. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10978–11001. PMLR, 21–27 Jul 2024.

Itay Evron, Edward Moroshko, Rachel Ward, Nathan Srebro, and Daniel Soudry. How catastrophic can catastrophic forgetting be in linear regression? In *Conference on Learning Theory*, pp. 4028–4079. PMLR, 2022.

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

Thomas S Ferguson. A bayesian analysis of some nonparametric problems. *The annals of statistics*, pp. 209–230, 1973.

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. In *Proceedings of the 42nd International Conference on Machine Learning*, ICML'25. JMLR.org, 2025a.

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. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 19011–19033. PMLR, 13–19 Jul 2025b. URL https://proceedings.mlr.press/v267/ge25d.html.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6904–6913, 2017.

Haiyang Guo, Fanhu Zeng, Ziwei Xiang, Fei Zhu, Da-Han Wang, Xu-Yao Zhang, and Cheng-Lin Liu. Hide-llava: Hierarchical decoupling for continual instruction tuning of multimodal large language model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13572–13586, 2025a.

Haiyang Guo, Fanhu Zeng, Fei Zhu, Wenzhuo Liu, Da-Han Wang, Jian Xu, Xu-Yao Zhang, and Cheng-Lin Liu. Federated continual instruction tuning. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1325–1335. IEEE, 2025b.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3608–3617. IEEE, 2018.

Danna Gurari, Yinan Zhao, Meng Zhang, and Nilavra Bhattacharya. Captioning images taken by people who are blind. In *European Conference on Computer Vision*, pp. 417–434. Springer, 2020.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8320–8329. IEEE, 2021.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American statistical association*, 58(301):13–30, 1963.

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.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6693–6702. IEEE, 2019.

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

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 787–798, 2014.

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.

Soochan Lee, Junsoo Ha, Dongsu Zhang, and Gunhee Kim. A neural dirichlet process mixture model for task-free continual learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SJxSOJStPr.

Hongbo Li, Sen Lin, Lingjie Duan, Yingbin Liang, and Ness Shroff. Theory on mixture-of-experts in continual learning. In *International Conference on Learning Representations*, volume 2025, pp. 8169–8206, 2025.

Lei Li, Yuqi Wang, Runxin Xu, Peiyi Wang, Xiachong Feng, Lingpeng Kong, and Qi Liu. Multimodal arxiv: A dataset for improving scientific comprehension of large vision-language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14369–14387, 2024.

Sen Lin, Peizhong Ju, Yingbin Liang, and Ness Shroff. Theory on forgetting and generalization of continual learning. In *International Conference on Machine Learning*, pp. 21078–21100. PMLR, 2023.

Adam Dahlgren Lindström and Savitha Sam Abraham. Clevr-math: A dataset for compositional language, visual and mathematical reasoning. *arXiv preprint arXiv:2208.05358*, 2022.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26286–26296. IEEE, 2024.

Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. IconQA: A new benchmark for abstract diagram understanding and visual language reasoning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=uXa9oBDZ9V1.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in neural information processing systems*, 35:2507–2521, 2022.

Tongxu Luo, Jiahe Lei, Fangyu Lei, Weihao Liu, Shizhu He, Jun Zhao, and Kang Liu. Moelora: Contrastive learning guided mixture of experts on parameter-efficient fine-tuning for large language models. *arXiv preprint arXiv:2402.12851*, 2024.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 11–20, 2016.

Colin McDiarmid et al. On the method of bounded differences. *Surveys in combinatorics*, 141(1): 148–188, 1989.

Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *2019 international conference on document analysis and recognition (ICDAR)*, pp. 947–952. IEEE, 2019.

Carl N. Morris. Parametric empirical Bayes inference: Theory and applications. *Journal of the American Statistical Association*, 78(381):47–55, 1983.

Kevin P Murphy. Conjugate bayesian analysis of the gaussian distribution. *def*, 1(2σ2):16, 2007.

Radford M Neal. Markov chain sampling methods for dirichlet process mixture models. *Journal of computational and graphical statistics*, 9(2):249–265, 2000.

Omiros Papaspiliopoulos. High-dimensional probability: An introduction with applications in data science, 2020.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Yu-Cheng Shi, Zhen-Hao Xie, Jun-Tao Tang, and Da-Wei Zhou. Protoada: Prototype-guided adaptive adapter expansion and geometric consolidation for multimodal continual instruction tuning. *arXiv preprint arXiv:2606.02576*, 2026.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8309–8318. IEEE, 2019.

Jun-Tao Tang, Zhen-Hao Xie, Yu-Cheng Shi, and Da-Wei Zhou. Cram: Centroid-routing and adaptive moe for multimodal continual instruction tuning. *arXiv preprint arXiv:2606.02502*, 2026.

Huiyi Wang, Haodong Lu, Lina Yao, and Dong Gong. Self-expansion of pre-trained models with mixture of adapters for continual learning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10087–10098. IEEE, 2025a.

Lianming Wang and David B. Dunson. Fast Bayesian inference in Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 20(1):196–216, 2011.

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

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 *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 177–186. IEEE, 2025b.

Zhen-Hao Xie, Jun-Tao Tang, Yu-Cheng Shi, Han-Jia Ye, De-Chuan Zhan, and Da-Wei Zhou. Same: Stabilized mixture-of-experts for multimodal continual instruction tuning. *arXiv preprint arXiv:2602.01990*, 2026.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the association for computational linguistics*, 2:67–78, 2014.

Fanhu Zeng, Fei Zhu, Haiyang Guo, Xu-Yao Zhang, and Cheng-Lin Liu. Modalprompt: Towards efficient multimodal continual instruction tuning with dual-modality guided prompt. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 12137–12152, 2025.

Duzhen Zhang, Yong Ren, Zhong-Zhi Li, Yahan Yu, Jiahua Dong, Chenxing Li, Zhilong Ji, and Jinfeng Bai. Enhancing multimodal continual instruction tuning with branchlora. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5743–5756, 2025.

Chongyang Zhao, Mingsong Li, Haodong Lu, and Dong Gong. On Token's Dilemma: Dynamic MoE with Drift-Aware Token Assignment for Continual Learning of Large Vision Language Models. *arXiv preprint arXiv:2603.27481*, 2026a.

Hengyuan Zhao, Ziqin Wang, Qixin Sun, Kaiyou Song, Yilin Li, Xiaolin Hu, Qingpei Guo, and Si Liu. Llava-cmoe: Towards continual mixture of experts for large vision-language models. *arXiv preprint arXiv:2503.21227*, 2025.

Hongbo Zhao, Fei Zhu, Haiyang Guo, Meng Wang, Rundong Wang, Gaofeng Meng, and Zhaoxiang Zhang. MLLM-CL: Continual learning for multimodal large language models, 2026b. URL https://openreview.net/forum?id=hxIMpEvyMG.

A Additional Method Details

A.1 Algorithm

Algorithm 1 PACE: allocation, adaptation, and expert selection

Require: Task stream $\{\mathcal{D}_t\}_{t=1}^{T}$, frozen Θ_0 and φ, $\log\alpha = \lambda_0 d$ (Eq. 13), NIG settings, and sufficient expert capacity

Initialize an empty summary bank and $K_0 \leftarrow 0$
for each arriving task $t = 1, \dots, T$ **do**
  Compute $\beta_t = (\bar{u}_t, v_t, M_t)$ using φ; set $r_t \leftarrow \bar{u}_t$
  if $t = 1$ **then**
    Set $z_t \leftarrow 1$ and initialize the first LoRA expert
  else
    Construct $G_{0,t}$ from $\{\beta_s : s < t\}$ and compute $q_0(r_t)$
    for $k = 1, \dots, K_{t-1}$ **do**
      Form $\mathcal{G}_k = \{s < t : z_s = k\}$ and $n_k = |\mathcal{G}_k|$
      Compute $q_k(r_t)$ and $S_{t,k}$ under the current $G_{0,t}$
    end for
    Select z_t by Eq. 5
    if $z_t = K_{t-1} + 1$ **then**
      Initialize the new expert
    end if
  end if
  Set $K_t \leftarrow \max\{K_{t-1}, z_t\}$
  Freeze Θ_0 and all θ_k with $k \neq z_t$
  Minimize $\mathcal{L}_t(\Theta_0, \theta_{z_t})$ on $\mathcal{D}_t$ using only expert z_t
  Store β_t and z_t without revising previous entries
end for
Inference after task t:
Compute u_x and select $\hat{s}_t(x)$ by Eq. 6
Set $\hat{k}_t(x) \leftarrow z_{\hat{s}_t(x)}$ and generate using only $\theta_{\hat{k}_t(x)}$

Algorithm 1 describes the core allocation and adaptation procedure of PACE. More implementation details can be found in Appendix C.2.

A.2 Determine the concentration parameter

In PACE, α enters the allocation only through the comparison of $\log\alpha$ with $S_{t,k}$ in Eq. 5. The diagonal likelihood treats the d coordinates as independent, so $\log q_k(r_t) - \log q_0(r_t)$ is a sum of d per-coordinate terms and $\log\alpha$ must be on the same scale. Read as a CRP prior, a value of order 10^4 would make a new group almost certain before any evidence is seen. We therefore treat $\log\alpha$ as a threshold on the summed evidence: it sets how far an arriving task must be from every existing group before a new expert is created. CN-DPM (Lee et al., 2020) sets α per benchmark to obtain between 2 and 20 experts. PACE instead fixes $\log\alpha$ before the first task arrives, using only the prior settings (κ_0, a_0) and the feature dimension d. The score $S_{t,k}$ is a sum of d per-coordinate terms, and because $G_{0,t}$ standardizes each coordinate by the spread of the previous task means, it is unchanged when a coordinate is rescaled, up to the variance floor. We therefore set $\log\alpha = \lambda_0 d$, where λ_0 is the expected per-coordinate score of a one-task group whose task is independent of the arriving task. In units where $G_{0,t}$ has mean zero and unit between-task variance, let x be the mean of the task in the group and y the mean of the arriving task. For independent tasks, x and y are independent draws from $\mathcal{N}(0,1)$, and

$$\lambda_0 = \mathbb{E}_{x,y\sim\mathcal{N}(0,1)}\left[\log \mathrm{St}_{2a_1}\left(y; \frac{x}{\kappa_0+1}, \frac{b_1(\kappa_0+2)}{a_1(\kappa_0+1)}\right) - \log \mathrm{St}_{2a_0}\left(y; 0, \frac{\kappa_0+1}{a_0\kappa_0}\right)\right], \tag{13}$$

with $a_1 = a_0 + \frac{1}{2}$ and $b_1 = 1 + \kappa_0 x^2/(2\kappa_0+2)$ from the updates in Appendix A.3. The two terms are $\log q_k$ and $\log q_0$ for one coordinate, and $\log n_k = 0$ for a one-task group. With $\kappa_0 = 0.01$

and $a_0 = 2$, numerical integration gives $\lambda_0 = 1.113$ nats per coordinate, so $\log \alpha = 15{,}960.1$ for Qwen2.5-VL ($d = 14{,}336$) and $18{,}240.1$ for LLaVA-1.5 ($d = 16{,}384$). An arriving task therefore reuses an existing expert only when its best score reaches the score expected for an independent one-task group.

Because $\log \alpha$ is fixed before task 1, the decision at boundary t depends only on $\beta_1, \ldots, \beta_t$ and the earlier assignments $z_{<t}$. By induction over t, each z_t is a function of $\beta_1, \ldots, \beta_t$ alone. No statistic of a later task enters any decision, and the number of experts is not specified in advance.

Sensitivity. To measure how much the allocation depends on this choice, we use the following property. For a fixed task order, Eq. 5 depends on α only through comparisons between $\log \alpha$ and the boundary scores $S_t^{\max} = \max_{k \le K_{t-1}} S_{t,k}$, $t = 2, \ldots, T$. Suppose a value c of $\log \alpha$ creates a new expert at the boundaries in $\mathcal{C}$ and reuses an existing expert at those in $\mathcal{R}$. Then every $\log \alpha \in [L, U)$, with $L = \max_{t \in \mathcal{C}} S_t^{\max}$ and $U = \min_{t \in \mathcal{R}} S_t^{\max}$, produces the same assignments as c. This follows by induction over t: if the decisions before t are unchanged, then $S_t^{\max}$ is unchanged, and $\log \alpha$ lies on the same side of it as c. The allocation is therefore piecewise constant in $\log \alpha$, and Algorithm 2 enumerates its intervals by replaying Eq. 5 on the stored statistics. It is an offline analysis of a completed task sequence and is not used to set $\log \alpha$. The replay creates an expert when $c \ge S_t^{\max}$. This differs from Eq. 5 only when c equals a boundary score.

Algorithm 2 Allocation intervals over $\log \alpha$

Require: Task statistics $\{\beta_t\}_{t=1}^{T}$ of a completed task sequence, NIG settings (κ_0, a_0)
Ensure: Every allocation $z_{1:T}^{(j)}$, its number of experts $K^{(j)}$, and its interval $[L_j, U_j)$ of $\log \alpha$

function REPLAY(c) ▷ allocation induced by $\log \alpha = c$
  $z_1 \leftarrow 1$; $K_1 \leftarrow 1$; $\mathcal{C} \leftarrow \emptyset$; $\mathcal{R} \leftarrow \emptyset$
  for $t = 2, \ldots, T$ **do**
    Construct $G_{0,t}$ from $\{\beta_s : s < t\}$ and compute $S_{t,k}$ for $k \le K_{t-1}$ by Eq. 5
    $S_t^{\max} \leftarrow \max_{k \le K_{t-1}} S_{t,k}$
    if $c \ge S_t^{\max}$ **then**
      $z_t \leftarrow K_{t-1} + 1$; $\mathcal{C} \leftarrow \mathcal{C} \cup \{t\}$ ▷ new expert
    else
      $z_t \leftarrow \arg\max_{k \le K_{t-1}} S_{t,k}$; $\mathcal{R} \leftarrow \mathcal{R} \cup \{t\}$ ▷ reuse
    end if
    $K_t \leftarrow \max\{K_{t-1}, z_t\}$
  end for
  $L \leftarrow \max_{t \in \mathcal{C}} S_t^{\max}$ ($L = -\infty$ if $\mathcal{C} = \emptyset$)
  $U \leftarrow \min_{t \in \mathcal{R}} S_t^{\max}$ ($U = +\infty$ if $\mathcal{R} = \emptyset$)
  return $(z_{1:T}, K_T, L, U)$
end function
Enumerate allocations:
$j \leftarrow 0$; $c \leftarrow -\infty$
repeat
  $j \leftarrow j + 1$
  $(z_{1:T}^{(j)}, K^{(j)}, L_j, U_j) \leftarrow$ REPLAY(c) ▷ same allocation for all $\log \alpha \in [L_j, U_j)$
  $c \leftarrow U_j$
until $U_j = +\infty$
return $\{(z_{1:T}^{(j)}, K^{(j)}, L_j, U_j)\}_{j=1}^{J}$ with $J = j$ ▷ $K^{(1)} = 1$ and $K^{(J)} = T$

On CoIN (using Qwen-2.5-VL), every $\log \alpha$ from 12,165.85 to 18,080.25 produces the same assignments as $\lambda_0 d$, which corresponds to every $\lambda = \log \alpha / d$ from 0.849 to 1.261. On UCIT (using LLaVA-1.5), the range is 0.402 to 1.461. Because α affects training only through the assignments, the PACE results in Tabs. 1 and 2 all have $\log \alpha = \lambda_0 d$ fixed before the stream.

A.3 Posterior Predictive Derivation

This appendix provides the closed-form posterior expressions omitted from Section 4.2. At task boundary t, the base parameters $\mu_{0t,j}$ and $b_{0t,j}$ are those of $G_{0,t}$ defined in Sec. 4.2. We adopt the inverse-gamma convention $p(\sigma^2) \propto (\sigma^2)^{-a_0-1}\exp(-b_{0t,j}/\sigma^2)$.

For a nonempty group $\mathcal{G}_k$, let $\bar{r}_{k,j} = n_k^{-1}\sum_{s\in\mathcal{G}_k} r_{s,j}$ denote the mean of its assigned task means in coordinate j. Conditioning the NIG prior in Eq. 4 on these task means gives:

$$\begin{aligned} \kappa_k &= \kappa_0 + n_k, & a_k &= a_0 + \frac{n_k}{2}, \\ \mu_{kj} &= \frac{\kappa_0\mu_{0t,j} + n_k\bar{r}_{k,j}}{\kappa_k}, & & \\ b_{kj} &= b_{0t,j} + \frac{1}{2}\sum_{s\in\mathcal{G}_k}(r_{s,j}-\bar{r}_{k,j})^2 + \frac{\kappa_0 n_k}{2\kappa_k}(\bar{r}_{k,j}-\mu_{0t,j})^2. & & \end{aligned} \tag{14}$$

A.4 Derivation of the Gaussian Selection Rule

We provide the likelihood derivation underlying the expert-selection rule in Eq. 6. For candidate task s, consider the shared-diagonal Gaussian model $u_x \mid s \sim \mathcal{N}(\bar{u}_s^x, \mathrm{diag}(\bar{v}_t))$. Its log-likelihood is:

$$\log p(u_x \mid s) = -\frac{d_x}{2}\log(2\pi) - \frac{1}{2}\sum_{j=1}^{d_x}\log\bar{v}_{t,j} - \frac{1}{2}\sum_{j=1}^{d_x}\frac{(u_{x,j}-\bar{u}_{s,j}^x)^2}{\bar{v}_{t,j}}. \tag{15}$$

Because the covariance is shared across candidate tasks, the first two terms are independent of s. Under the equal task priors assumed in Eq. 6, maximum a posteriori task identification is therefore equivalent to minimizing the final variance-normalized squared distance, which gives the expert-selection rule in Sec. 4.3.

B Additional Theoretical Results and Proofs

B.1 Scope and Setup of the Theoretical Analysis

Throughout the theoretical analysis, the task order and target parameters are fixed. Expectations involving the expert parameters are taken over the random task matrices $\{X_t\}$, conditional on the frozen statistics used for allocation. Assumption 3 ensures that this condition preserves the assumed independence and isotropy of $\{X_t\}$. A frozen representation is invariant to adapter optimization, but its statistics can still depend on the samples from which they were computed. Consequently, performing allocation before optimization does not by itself imply independence between the allocation decision and the samples used for adaptation. Using independent samples for task summarization and adaptation provides one way to satisfy this assumption; when the same samples are used for both purposes, an additional conditional argument is required.

Expert k is created at its first assigned task $\tau_k(1)$ with parameter $\theta_k^{(0)}$. We interpret $\theta_k^{(\tau_k(1)-1)} = \theta_k^{(0)}$ in Eq. 7. For a newly created expert, its initialization may depend on experts learned from earlier tasks, as in the averaging initialization used by PACE, but not on X_t from its current or later assigned tasks. For the first expert, a fixed initialization such as the zero vector may be used.

Expert update under interpolation. For task t, we consider the loss $\frac{1}{2}\|X_t^\top\theta - y_t\|_2^2$. Under Assumption 1, X_t has rank N. Every gradient lies in $\mathrm{range}(X_t)$, so gradient descent leaves the component of the initial state in $\ker(X_t^\top)$ unchanged. With $0 < \eta_t < \frac{2}{\lambda_{\max}(X_t^\top X_t)}$, the component in $\mathrm{range}(X_t)$ converges to the one that interpolates y_t. If θ_{pre} is the expert state before learning task t, the resulting parameter is:

$$\theta_{\mathrm{pre}} + X_t(X_t^\top X_t)^{-1}\big(y_t - X_t^\top\theta_{\mathrm{pre}}\big). \tag{16}$$

Any other interpolating solution differs from this parameter by a vector in $\ker(X_t^\top)$, which is orthogonal to the update in $\mathrm{range}(X_t)$. Hence, Eq. 16 is the interpolant closest to θ_{pre} and gives Eq. 7. This analysis concerns the update in Eq. 7, rather than the nonlinear optimization of the full MLLM.

Excess risk and initialization. For an independent test vector $x \sim \mathcal{N}(0, I_p)$,

$$\mathbb{E}_x\big(x^\top\theta - x^\top\theta_t^\star\big)^2 = (\theta - \theta_t^\star)^\top I_p(\theta - \theta_t^\star) = E_t(\theta). \tag{17}$$

The initialization error $\xi_{k,q}^2$ in Eq. 8 can depend on the states of experts learned from earlier tasks. It is therefore not assumed to be the same under different allocations. Accordingly, the comparison between expert sharing and isolation following Theorem 2, derived in Sec. B.5, explicitly uses matched initialization errors.

B.2 Additional Lemmas for expert updates

Let $P_t = X_t(X_t^\top X_t)^{-1}X_t^\top$, the orthogonal projector onto $\mathrm{range}(X_t)$.

Lemma 1. *Under Assumptions 1–2, for the expert $k = z_t$ and every $\theta \in \mathbb{R}^p$,*

$$\begin{aligned}\theta_k^{(t)} - \theta &= (I_p - P_t)(\theta_k^{(t-1)} - \theta) + P_t(\theta_t^\star - \theta),\\ \|\theta_k^{(t)} - \theta\|_2^2 &= \|(I_p - P_t)(\theta_k^{(t-1)} - \theta)\|_2^2 + \|P_t(\theta_t^\star - \theta)\|_2^2.\end{aligned} \tag{18}$$

Lemma 1 follows from the projection view of sequential interpolation in overparameterized linear regression (Evron et al., 2022). This lemma characterizes how a single task update decomposes the expert error into the component retained from its previous state and the component introduced by the current task.

Proof. Fix task t and let $k = z_t$. Under Assumption 1, the N Gaussian columns of X_t are linearly independent with probability one because $N < p$. Hence, $X_t^\top X_t$ is invertible with probability one, and P_t is well defined.

First, P_t is symmetric because $X_t^\top X_t$ and its inverse are symmetric. Moreover,

$$\begin{aligned}P_t^2 &= X_t(X_t^\top X_t)^{-1}X_t^\top X_t(X_t^\top X_t)^{-1}X_t^\top\\ &= X_t(X_t^\top X_t)^{-1}(X_t^\top X_t)(X_t^\top X_t)^{-1}X_t^\top = P_t.\end{aligned} \tag{19}$$

Thus, P_t is a symmetric idempotent matrix. Its range is contained in $\mathrm{range}(X_t)$, while for every vector $a \in \mathbb{R}^N$,

$$P_tX_ta = X_t(X_t^\top X_t)^{-1}X_t^\top X_ta = X_ta. \tag{20}$$

Therefore, P_t is the orthogonal projector onto $\mathrm{range}(X_t)$, and $I_p - P_t$ is the orthogonal projector onto its orthogonal complement. In particular,

$$P_t(I_p - P_t) = (I_p - P_t)P_t = 0. \tag{21}$$

By Assumption 2, $y_t = X_t^\top\theta_t^\star$. Substituting this identity into Eq. 7 gives

$$\theta_k^{(t)} = \theta_k^{(t-1)} + X_t(X_t^\top X_t)^{-1}X_t^\top\big(\theta_t^\star - \theta_k^{(t-1)}\big) = (I_p - P_t)\theta_k^{(t-1)} + P_t\theta_t^\star. \tag{22}$$

For any $\theta \in \mathbb{R}^p$, subtracting $\theta = (I_p - P_t)\theta + P_t\theta$ from both sides yields

$$\theta_k^{(t)} - \theta = (I_p - P_t)\big(\theta_k^{(t-1)} - \theta\big) + P_t\big(\theta_t^\star - \theta\big), \tag{23}$$

which is the first identity in Eq. 18. The two terms on the right-hand side are orthogonal because

$$\Big\langle(I_p - P_t)\big(\theta_k^{(t-1)} - \theta\big), P_t\big(\theta_t^\star - \theta\big)\Big\rangle = \big(\theta_k^{(t-1)} - \theta\big)^\top(I_p - P_t)P_t\big(\theta_t^\star - \theta\big) = 0. \tag{24}$$

Applying the Pythagorean identity therefore gives the second equality in Eq. 18. The argument is deterministic conditional on X_t and the preceding expert state; no independence condition is needed for this one-step decomposition. □

Lemma 2. *Under Assumption 1, $\mathbb{E}[P_t] = (N/p)I_p$. For any square-integrable random vector v independent of X_t,*

$$\mathbb{E}\|P_tv\|_2^2 = (1 - \varrho)\mathbb{E}\|v\|_2^2, \qquad \mathbb{E}\|(I_p - P_t)v\|_2^2 = \varrho\mathbb{E}\|v\|_2^2. \tag{25}$$

This result follows from the rotational invariance of isotropic Gaussian matrices and the standard expected-norm property of projection onto a uniformly random subspace (Papaspiliopoulos, 2020). Lemma 2 quantifies the expected effect of the projection in Lemma 1, producing the contraction factor ϱ that governs how much of the previous error is retained after each update.

Proof. For every orthogonal matrix $Q \in \mathbb{R}^{p\times p}$, the columns of QX_t remain independent $\mathcal{N}(0, I_p)$ vectors. Consequently, QX_t and X_t have the same distribution. The projector constructed from QX_t is

$$(QX_t)\big((QX_t)^\top QX_t\big)^{-1}(QX_t)^\top = QX_t(X_t^\top X_t)^{-1}X_t^\top Q^\top = QP_tQ^\top. \tag{26}$$

Hence, $\mathbb{E}[P_t]$ is invariant under every orthogonal transformation Q, which implies that $\mathbb{E}[P_t] = cI_p$ for some $c \in \mathbb{R}$. Taking the trace gives

$$\begin{aligned}\operatorname{tr}(\mathbb{E}[P_t]) &= \mathbb{E}\left[\operatorname{tr}\left(X_t(X_t^\top X_t)^{-1}X_t^\top\right)\right] \\ &= \mathbb{E}\left[\operatorname{tr}\left((X_t^\top X_t)^{-1}X_t^\top X_t\right)\right] = \operatorname{tr}(I_N) = N.\end{aligned} \tag{27}$$

Since $\operatorname{tr}(cI_p) = cp$, we obtain $c = N/p$. Therefore, $\mathbb{E}[P_t] = (N/p)I_p = (1-\varrho)I_p$.

Now let v satisfy the conditions in the lemma. Because P_t is a function of X_t, its independence from v gives $\mathbb{E}[P_t \mid v] = \mathbb{E}[P_t]$. Also, $P_t^\top P_t = P_t$. Conditioning on v and then taking expectations, we obtain

$$\mathbb{E}\|P_tv\|_2^2 = \mathbb{E}\big[v^\top P_t v\big] = \mathbb{E}\big[v^\top \mathbb{E}[P_t \mid v]v\big] = \frac{N}{p}\,\mathbb{E}\|v\|_2^2 = (1-\varrho)\mathbb{E}\|v\|_2^2. \tag{28}$$

Similarly, $(I_p - P_t)^\top(I_p - P_t) = I_p - P_t$, and hence

$$\begin{aligned}\mathbb{E}\|(I_p - P_t)v\|_2^2 &= \mathbb{E}\big[v^\top(I_p - P_t)v\big] = \mathbb{E}\big[v^\top(I_p - \mathbb{E}[P_t \mid v])v\big] \\ &= \left(1 - \frac{N}{p}\right)\mathbb{E}\|v\|_2^2 = \varrho\mathbb{E}\|v\|_2^2.\end{aligned}$$

Both expectations are finite because orthogonal projections do not increase the Euclidean norm and $\mathbb{E}\|v\|_2^2 < \infty$. This proves Eq. 25. □

Lemma 3. *Under Assumptions 1–3, define $e_{k,q}(0) = \xi_{k,q}^2$. For any expert k with assigned task set $\mathcal{G}_k$, any task q, and $1 \le \ell \le n_k$,*

$$e_{k,q}(\ell) := \mathbb{E}\|\theta_k^{(\tau_k(\ell))} - \theta_q^\star\|_2^2 = \varrho^\ell \xi_{k,q}^2 + (1-\varrho)\sum_{i=1}^{\ell}\varrho^{\ell-i}\|\theta^\star_{\tau_k(i)} - \theta_q^\star\|_2^2, \tag{29}$$

where $\tau_k(n_k)$ denotes the last task assigned to expert k. Expert k is unchanged between its assigned tasks and after $\tau_k(n_k)$.

Applying Lemmas 1 and 2 recursively to the tasks assigned to the same expert gives the error expression in Eq. 29. This follows the projection-based analysis of sequential linear regression (Evron et al., 2022; Ding et al., 2024), adapted here to expert-specific task subsequences. Lemma 3 extends the one-step result to successive tasks assigned to the same expert, providing the error recursion used to derive the forgetting and generalization results in Theorems 1 and 2.

Proof. Fix an expert k and a task q. Before task $\tau_k(\ell)$, expert k depends only on its initialization and previously assigned task matrices. By Assumptions 1 and 3, this state is independent of $X_{\tau_k(\ell)}$. Moreover, the required expected squared errors are finite.

Applying Lemmas 1 and 2 gives:

$$\mathbb{E}\Big[\,\|\theta_k^{(\tau_k(\ell))} - \theta_q^\star\|_2^2 \,\Big|\, \theta_k^{(\tau_k(\ell)-1)}\Big] = \varrho\|\theta_k^{(\tau_k(\ell)-1)} - \theta_q^\star\|_2^2 + (1-\varrho)\|\theta^\star_{\tau_k(\ell)} - \theta_q^\star\|_2^2. \tag{30}$$

Taking expectations and using $\theta_k^{(\tau_k(\ell)-1)} = \theta_k^{(\tau_k(\ell-1))}$ $(\ell \ge 2)$, gives

$$e_{k,q}(\ell) = \varrho e_{k,q}(\ell-1) + (1-\varrho)\|\theta^\star_{\tau_k(\ell)} - \theta_q^\star\|_2^2, \qquad 1 \le \ell \le n_k. \tag{31}$$

For $\ell = 1$, $e_{k,q}(1) = \varrho\xi_{k,q}^2 + (1-\varrho)\|\theta^\star_{\tau_k(1)} - \theta^\star_q\|_2^2$, which agrees with Eq. 29. Suppose that Eq. 29 holds for $\ell - 1$, substitution into the recursion gives:

$$\begin{aligned} e_{k,q}(\ell) &= \varrho\left[\varrho^{\ell-1}\xi_{k,q}^2 + (1-\varrho)\sum_{i=1}^{\ell-1}\varrho^{\ell-1-i}\|\theta^\star_{\tau_k(i)} - \theta^\star_q\|_2^2\right] \\ &\quad + (1-\varrho)\|\theta^\star_{\tau_k(\ell)} - \theta^\star_q\|_2^2 \\ &= \varrho^\ell\xi_{k,q}^2 + (1-\varrho)\sum_{i=1}^{\ell}\varrho^{\ell-i}\|\theta^\star_{\tau_k(i)} - \theta^\star_q\|_2^2. \end{aligned} \tag{32}$$

Since only assigned tasks update expert k, $\theta_k^{(T)} = \theta_k^{(\tau_k(n_k))}$, and the expert is unchanged between consecutive assigned tasks. □

B.3 Proof of Theorem 1

Proof. Based on Lemma 3, we have:

$$\mathbb{E}[F_T] = \frac{1}{T-1}\sum_{t=1}^{T}\left[\mathbb{E}\left[E_t(\theta_{z_t}^{(T)})\right] - \mathbb{E}\left[E_t(\theta_{z_t}^{(t)})\right]\right] = \frac{1}{T-1}\sum_{k=1}^{K}\sum_{j=1}^{n_k}\left[e_{k,\tau_k(j)}(n_k) - e_{k,\tau_k(j)}(j)\right].$$

Fix k and j and apply Eq. 29 at n_k and at j gives:

$$\begin{aligned} e_{k,\tau_k(j)}(n_k) - e_{k,\tau_k(j)}(j) \;=\; (\varrho^{n_k} - \varrho^j)\xi_{k,\tau_k(j)}^2 \;+\; (1-\varrho)\sum_{i=1}^{n_k}\varrho^{n_k-i}\|\theta^\star_{\tau_k(i)} - \theta^\star_{\tau_k(j)}\|_2^2 \\ - (1-\varrho)\sum_{i=1}^{j}\varrho^{j-i}\|\theta^\star_{\tau_k(i)} - \theta^\star_{\tau_k(j)}\|_2^2. \end{aligned}$$

The term $i = j$ is zero in both sums. Therefore, separating the indices $i < j$ and $i > j$ gives

$$\begin{aligned} e_{k,\tau_k(j)}(n_k) - e_{k,\tau_k(j)}(j) \;&=(\varrho^{n_k} - \varrho^j)\xi_{k,\tau_k(j)}^2 \;+\; (1-\varrho)\sum_{i=1}^{j-1}(\varrho^{n_k-i} - \varrho^{j-i})\|\theta^\star_{\tau_k(i)} - \theta^\star_{\tau_k(j)}\|_2^2 \\ &+\; (1-\varrho)\sum_{i=j+1}^{n_k}\varrho^{n_k-i}\|\theta^\star_{\tau_k(i)} - \theta^\star_{\tau_k(j)}\|_2^2. \end{aligned} \tag{33}$$

For a fixed pair $1 \le i < j \le n_k$, its squared target distance occurs twice. The error for task $\tau_k(j)$ contributes the coefficient $(1-\varrho)(\varrho^{n_k-i} - \varrho^{j-i})$, while the error for task $\tau_k(i)$ contributes $(1-\varrho)\varrho^{n_k-j}$. The squared distances in these two terms coincide by symmetry of the Euclidean norm. Their combined coefficient is exactly: $(1-\varrho)\left(\varrho^{n_k-i} + \varrho^{n_k-j} - \varrho^{j-i}\right) = c_{ij}^{(n_k)}$. Consequently, we have:

$$\sum_{j=1}^{n_k}\left[e_{k,\tau_k(j)}(n_k) - e_{k,\tau_k(j)}(j)\right] \;=\; \sum_{j=1}^{n_k}(\varrho^{n_k} - \varrho^j)\xi_{k,\tau_k(j)}^2 \;+\; \sum_{1\le i<j\le n_k} c_{ij}^{(n_k)}\|\theta^\star_{\tau_k(i)} - \theta^\star_{\tau_k(j)}\|_2^2. \tag{34}$$

Substituting Eq. 34 into the preceding expression for $\mathbb{E}[F_T]$ gives:

$$\mathbb{E}[F_T] = \frac{1}{T-1}\sum_{k=1}^{K}\left[\sum_{j=1}^{n_k}(\varrho^{n_k} - \varrho^j)\xi_{k,\tau_k(j)}^2 + \sum_{1\le i<j\le n_k} c_{ij}^{(n_k)}\|\theta^\star_{\tau_k(i)} - \theta^\star_{\tau_k(j)}\|_2^2\right], \tag{35}$$

which is Eq. 10. □

B.4 Proof of Corollary 1

Proof. The bound follows from Eq. 10 in Theorem 1. Fix an expert k. Since $0 < \varrho < 1$ and $j \le n_k$, every coefficient $\varrho^{n_k} - \varrho^j$ in Eq. 10 is nonpositive, while $\xi_{k,\tau_k(j)}^2 \ge 0$. Dropping the initialization

terms therefore gives

$$\mathbb{E}[F_T] \le \frac{1}{T-1}\sum_{k=1}^{K}\sum_{1\le i<j\le n_k} c_{ij}^{(n_k)}\|\theta^\star_{\tau_k(i)}-\theta^\star_{\tau_k(j)}\|_2^2. \tag{36}$$

For $1 \le i < j \le n_k$, we have $n_k - i \ge j - i$, so $\varrho^{n_k-i} \le \varrho^{j-i}$ and hence $c_{ij}^{(n_k)} \le (1-\varrho)\varrho^{n_k-j}$. Both targets in every pair belong to $\mathcal{G}_k$, so each squared distance is at most Δ_k^2. As the squared distances are nonnegative,

$$\sum_{1\le i<j\le n_k} c_{ij}^{(n_k)}\|\theta^\star_{\tau_k(i)}-\theta^\star_{\tau_k(j)}\|_2^2 \le (1-\varrho)\Delta_k^2\sum_{j=2}^{n_k}(j-1)\varrho^{n_k-j} = (1-\varrho)\Delta_k^2\sum_{m=0}^{n_k-2}(n_k-1-m)\varrho^m, \tag{37}$$

where the equality substitutes $m = n_k - j$; both sums are empty when $n_k = 1$. Writing $n_k - 1 - m = \sum_{r=m+1}^{n_k-1} 1$ and exchanging the order of summation,

$$\begin{aligned}(1-\varrho)\sum_{m=0}^{n_k-2}(n_k-1-m)\varrho^m &= (1-\varrho)\sum_{r=1}^{n_k-1}\sum_{m=0}^{r-1}\varrho^m \\ &= \sum_{r=1}^{n_k-1}(1-\varrho^r) = n_k - \sum_{r=0}^{n_k-1}\varrho^r \\ &= n_k - \frac{1-\varrho^{n_k}}{1-\varrho} = c_\varrho(n_k).\end{aligned} \tag{38}$$

Substituting Eqs. 37 and 38 into Eq. 36 proves the first inequality in Eq. 11.

It remains to bound $c_\varrho(n)$. From its definition and the geometric-series identity,

$$c_\varrho(n) = n - \sum_{m=0}^{n-1}\varrho^m = \sum_{m=0}^{n-1}(1-\varrho^m). \tag{39}$$

The term $m = 0$ is zero and every other term is at most one, so $c_\varrho(n) \le n - 1$. Also, for $m \ge 1$,

$$1-\varrho^m = (1-\varrho)\sum_{r=0}^{m-1}\varrho^r \le (1-\varrho)m = \frac{N}{p}\,m. \tag{40}$$

Consequently,

$$c_\varrho(n) \le \frac{N}{p}\sum_{m=1}^{n-1} m = \frac{N}{p}\frac{n(n-1)}{2}. \tag{41}$$

Combining the two estimates gives

$$\boxed{c_\varrho(n) \le \min\left\{n-1, \frac{N}{p}\frac{n(n-1)}{2}\right\}} \tag{42}$$

which completes the proof of Eq. 11.

□

Special cases. If $K = T$, every expert is assigned one task and receives no subsequent updates, so $F_T = 0$. If $\Delta_k = 0$ for every expert, all tasks assigned to the same expert have the same target, so we have

$$\mathbb{E}[F_T] = \frac{1}{T-1}\sum_{k=1}^{K}\sum_{j=1}^{n_k}(\varrho^{n_k}-\varrho^j)\xi^2_{k,\tau_k(j)} \le 0, \tag{43}$$

since $\varrho^{n_k} \le \varrho^j$ for $j \le n_k$. Finally, Eq. 7 shows that updating one expert leaves the others unchanged.

B.5 Proof of Theorem 2

Proof. Fix an expert k and a task $q \in \mathcal{G}_k$. Since no task after $\tau_k(n_k)$ is assigned to expert k, so Eq. 7 implies $\theta_k^{(T)} = \theta_k^{(\tau_k(n_k))}$. Applying Lemma 3 with $\ell = n_k$ therefore gives the exact expected final error on task q:

$$\begin{aligned}\mathbb{E}\left[E_q\big(\theta_k^{(T)}\big)\right] &= \mathbb{E}\left[\left\|\theta_k^{(T)} - \theta_q^\star\right\|_2^2\right] = \mathbb{E}\left[\left\|\theta_k^{(\tau_k(n_k))} - \theta_q^\star\right\|_2^2\right] \\ &= e_{k,q}(n_k).\end{aligned} \tag{44}$$

From Lemma 3, for $1 \le \ell \le n_k$, $e_{k,q}(\ell) = \varrho e_{k,q}(\ell-1) + (1-\varrho)\left\|\theta^\star_{\tau_k(\ell)} - \theta_q^\star\right\|_2^2$. For $\ell \ge 2$, applying this recursion repeatedly gives:

$$\begin{aligned}
e_{k,q}(\ell) &= \varrho e_{k,q}(\ell-1) + (1-\varrho)\left\|\theta^\star_{\tau_k(\ell)} - \theta_q^\star\right\|_2^2 \\
&= \varrho\left[\varrho e_{k,q}(\ell-2) + (1-\varrho)\left\|\theta^\star_{\tau_k(\ell-1)} - \theta_q^\star\right\|_2^2\right] + (1-\varrho)\left\|\theta^\star_{\tau_k(\ell)} - \theta_q^\star\right\|_2^2 \\
&= \varrho^2 e_{k,q}(\ell-2) + (1-\varrho)\left[\varrho\left\|\theta^\star_{\tau_k(\ell-1)} - \theta_q^\star\right\|_2^2 + \left\|\theta^\star_{\tau_k(\ell)} - \theta_q^\star\right\|_2^2\right] \\
&= \varrho^2\left[\varrho e_{k,q}(\ell-3) + (1-\varrho)\left\|\theta^\star_{\tau_k(\ell-2)} - \theta_q^\star\right\|_2^2\right] \\
&\quad + (1-\varrho)\left[\varrho\left\|\theta^\star_{\tau_k(\ell-1)} - \theta_q^\star\right\|_2^2 + \left\|\theta^\star_{\tau_k(\ell)} - \theta_q^\star\right\|_2^2\right] \\
&= \varrho^3 e_{k,q}(\ell-3) + (1-\varrho)\left[\varrho^2\left\|\theta^\star_{\tau_k(\ell-2)} - \theta_q^\star\right\|_2^2 + \varrho\left\|\theta^\star_{\tau_k(\ell-1)} - \theta_q^\star\right\|_2^2 + \left\|\theta^\star_{\tau_k(\ell)} - \theta_q^\star\right\|_2^2\right] \\
&\ \ \vdots \\
&= \varrho^\ell e_{k,q}(0) + (1-\varrho)\sum_{i=1}^{\ell}\varrho^{\ell-i}\left\|\theta^\star_{\tau_k(i)} - \theta_q^\star\right\|_2^2 \\
&= \varrho^\ell \xi_{k,q}^2 + (1-\varrho)\sum_{i=1}^{\ell}\varrho^{\ell-i}\left\|\theta^\star_{\tau_k(i)} - \theta_q^\star\right\|_2^2,
\end{aligned} \tag{45}$$

where the last equality uses $e_{k,q}(0) = \xi_{k,q}^2$. For $\ell = 1$, the same expression follows directly from the one-step recursion. Setting $\ell = n_k$ therefore yields:

$$\mathbb{E}\left[E_q\big(\theta_k^{(T)}\big)\right] = \varrho^{n_k}\xi_{k,q}^2 + (1-\varrho)\sum_{i=1}^{n_k}\varrho^{n_k-i}\left\|\theta^\star_{\tau_k(i)} - \theta_q^\star\right\|_2^2. \tag{46}$$

Using linearity of expectation first, and then using the fact that $\mathcal{G}_1, \ldots, \mathcal{G}_K$ partition the T tasks, we obtain:

$$\begin{aligned}
\mathbb{E}[G_T] &= \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} E_t\big(\theta_{z_t}^{(T)}\big)\right] \\
&= \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[E_t\big(\theta_{z_t}^{(T)}\big)\right] \\
&= \frac{1}{T}\sum_{k=1}^{K}\sum_{q\in\mathcal{G}_k}\mathbb{E}\left[E_q\big(\theta_k^{(T)}\big)\right] \\
&= \frac{1}{T}\sum_{k=1}^{K}\sum_{q\in\mathcal{G}_k}\left[\varrho^{n_k}\xi_{k,q}^2 + (1-\varrho)\sum_{i=1}^{n_k}\varrho^{n_k-i}\left\|\theta^\star_{\tau_k(i)} - \theta_q^\star\right\|_2^2\right].
\end{aligned} \tag{47}$$

This proves the equality in Eq. 12.

For every $q \in \mathcal{G}_k$ and $1 \le i \le n_k$, both q and $\tau_k(i)$ belong to $\mathcal{G}_k$. Therefore, by the definition of Δ_k in Eq. 8, $\left\|\theta^\star_{\tau_k(i)} - \theta^\star_q\right\|_2^2 \le \Delta_k^2$. Substituting this inequality into Eq. 47 and carrying out the sums explicitly gives:

$$\begin{aligned}
\mathbb{E}[G_T] &\le \frac{1}{T}\sum_{k=1}^{K}\sum_{q\in\mathcal{G}_k}\left[\varrho^{n_k}\xi_{k,q}^2 + (1-\varrho)\Delta_k^2\sum_{i=1}^{n_k}\varrho^{n_k-i}\right] \\
&= \frac{1}{T}\sum_{k=1}^{K}\left[\varrho^{n_k}\sum_{q\in\mathcal{G}_k}\xi_{k,q}^2 + n_k(1-\varrho)\Delta_k^2\sum_{i=1}^{n_k}\varrho^{n_k-i}\right] \\
&= \frac{1}{T}\sum_{k=1}^{K}\left[n_k\varrho^{n_k}\bar{\xi}_k^2 + n_k(1-\varrho)\Delta_k^2\big(1+\varrho+\cdots+\varrho^{n_k-1}\big)\right] \\
&= \frac{1}{T}\sum_{k=1}^{K}\left[n_k\varrho^{n_k}\bar{\xi}_k^2 + n_k(1-\varrho)\Delta_k^2\frac{1-\varrho^{n_k}}{1-\varrho}\right] \\
&= \frac{1}{T}\sum_{k=1}^{K} n_k\left[\varrho^{n_k}\bar{\xi}_k^2 + (1-\varrho^{n_k})\Delta_k^2\right].
\end{aligned} \tag{48}$$

Here we used $\sum_{q\in\mathcal{G}_k}\xi_{k,q}^2 = n_k\bar{\xi}_k^2$ and the geometric identity $\sum_{i=1}^{n_k}\varrho^{n_k-i} = (1-\varrho^{n_k})/(1-\varrho)$. This proves the upper bound in Eq. 12. □

Comparison with isolated experts. Consider the same tasks in $\mathcal{G}_k$, but assign a separate expert to each task and match its expected initialization error on task q to $\xi_{k,q}^2$. An isolated expert receives only one update toward its own target. Applying Eq. 29 with one assigned task gives:

$$\text{expected final error on task } q = \varrho\xi_{k,q}^2 + (1-\varrho)\left\|\theta_q^\star - \theta_q^\star\right\|_2^2 = \varrho\xi_{k,q}^2. \tag{49}$$

Thus, the exact expected contribution of the n_k isolated experts to G_T is:

$$\frac{1}{T}\sum_{q\in\mathcal{G}_k}\varrho\xi_{k,q}^2 = \frac{\varrho}{T}\sum_{q\in\mathcal{G}_k}\xi_{k,q}^2 = \frac{n_k}{T}\varrho\bar{\xi}_k^2. \tag{50}$$

A sufficient condition for expert sharing to reduce the expected contribution of group $\mathcal{G}_k$ to G_T is that the upper bound under sharing be smaller than the exact contribution under isolation. Thus,

$$\begin{aligned}
&\frac{n_k}{T}\left[\varrho^{n_k}\bar{\xi}_k^2 + (1-\varrho^{n_k})\Delta_k^2\right] < \frac{n_k}{T}\varrho\bar{\xi}_k^2 \\
\iff\quad &\varrho^{n_k}\bar{\xi}_k^2 + (1-\varrho^{n_k})\Delta_k^2 < \varrho\bar{\xi}_k^2 \\
\iff\quad &(1-\varrho^{n_k})\Delta_k^2 < (\varrho-\varrho^{n_k})\bar{\xi}_k^2 \\
\iff\quad &\Delta_k^2 < \frac{\varrho-\varrho^{n_k}}{1-\varrho^{n_k}}\bar{\xi}_k^2.
\end{aligned} \tag{51}$$

Here, the first equivalence follows from $n_k/T > 0$, and the last follows from $1-\varrho^{n_k} > 0$ since $0 < \varrho < 1$. Because the shared-expert contribution is controlled by an upper bound, the resulting condition is sufficient, but not necessary, for expert sharing to improve generalization.

B.6 Frozen statistics and complementary response information

The assumptions for estimating task statistics concern the frozen feature representation and are separate from the assumptions used in the linear-regression analysis of Theorems 1–2.

Assumption 4. For each task s, the $M_s \ge 2$ feature vectors $\{u_i^{(s)}\}_{i=1}^{M_s}$ used to compute $\beta_s = (\bar{u}_s, v_s, M_s)$ are i.i.d. under $\mathcal{P}_s$, with M_s fixed. The representation function φ in Eq. 3 remains frozen during continual adaptation. Each feature component has Euclidean norm at most one, and v_s is the unbiased sample variance of $\{u_i^{(s)}\}_{i=1}^{M_s}$, computed coordinate-wise.

The norm bound also permits a missing feature component to be represented by zero. Independence between the summarized samples of different tasks is not required for the results below.

Lemma 4. *Under Assumption 4, let $\nu_s = \mathbb{E}_{\mathcal{P}_s}[u]$, $\sigma^2_{s,j} = \mathrm{Var}_{\mathcal{P}_s}(u_j)$ denote the population feature mean and coordinate-wise variance of task s, respectively, and $c_\delta = 1 + \sqrt{2\log(1/\delta)}$. For $\delta \in (0,1)$, with probability at least $1-\delta$, simultaneously for all $s \le T$,*

$$\|v_s - \sigma_s^2\|_\infty \le 3\epsilon_s + \frac{1}{M_s - 1}, \tag{52}$$

where $\epsilon_s = \sqrt{\frac{2\log(4dT/\delta)}{M_s}}$. For a fixed task s, each of the following bounds holds with probability at least $1-\delta$:

$$\|\bar{u}_s^x - \nu_s^x\|_2 \le \frac{\sqrt{2}\, c_\delta}{\sqrt{M_s}}. \tag{53}$$

Adapter optimization leaves the feature of every fixed example unchanged, and subsequent optimization does not alter retained summaries.

Lemma 4 establishes that the frozen empirical statistics retained by PACE consistently approximate the corresponding population mean and variance, providing the basis for relating observed task similarity to the task and adapter distances considered in Appendix B.7.

Proof. Fix a task s and a coordinate j, we apply Hoeffding's inequality Hoeffding (1963) to the independent variables $u^{(s)}_{1,j}, \dots, u^{(s)}_{M_s,j}$ gives:

$$\Pr(|\bar{u}_{s,j} - \nu_{s,j}| > \epsilon_s) \le 2\exp\left(-\frac{M_s\epsilon_s^2}{2}\right) = \frac{\delta}{2dT}. \tag{54}$$

Since the squared coordinates $(u^{(s)}_{i,j})^2$ lie in $[0,1]$, we have:

$$\Pr\left(\left|\frac{1}{M_s}\sum_{i=1}^{M_s}(u^{(s)}_{i,j})^2 - \mathbb{E}_{\mathcal{P}_s}[u_j^2]\right| > \epsilon_s\right) \le 2\exp(-2M_s\epsilon_s^2) = 2\left(\frac{4dT}{\delta}\right)^{-4} \le \frac{\delta}{2dT}. \tag{55}$$

Taking a union bound over both sets of dT deviations shows that, with probability at least $1-\delta$, both bounds hold for all $s \le T$ and $1 \le j \le d$. Hence, $\|\bar{u}_s - \nu_s\|_\infty \le \epsilon_s$ for all s.

From Assumption 4:

$$v_{s,j} = \frac{1}{M_s - 1}\sum_{i=1}^{M_s}(u^{(s)}_{i,j} - \bar{u}_{s,j})^2 = \frac{M_s}{M_s - 1}\underbrace{\left[\frac{1}{M_s}\sum_{i=1}^{M_s}(u^{(s)}_{i,j})^2 - \bar{u}^2_{s,j}\right]}_{(A)}. \tag{56}$$

The term (A) is non-negative and at most one. Since $\sigma^2_{s,j} = \mathbb{E}_{\mathcal{P}_s}[u_j^2] - \nu^2_{s,j}$, we have:

$$\begin{aligned} |v_{s,j} - \sigma^2_{s,j}| &\le \left|\frac{1}{M_s}\sum_{i=1}^{M_s}(u^{(s)}_{i,j})^2 \;-\; \mathbb{E}_{\mathcal{P}_s}[u_j^2]\right| \;+\; |\bar{u}^2_{s,j} - \nu^2_{s,j}| \;+\; \frac{1}{M_s - 1} \\ &\le \epsilon_s \;+\; |\bar{u}_{s,j} - \nu_{s,j}|\,|\bar{u}_{s,j} + \nu_{s,j}| \;+\; \frac{1}{M_s - 1} \\ &\le \epsilon_s + 2\epsilon_s + \frac{1}{M_s - 1} \;=\; 3\epsilon_s \;+\; \frac{1}{M_s - 1}, \end{aligned} \tag{57}$$

where $|\bar{u}_{s,j}| \le 1$ and $|\nu_{s,j}| \le 1$. Taking the maximum over coordinates proves Eq. 52.

We next prove the fixed-task Euclidean bounds. Starting with:

$$\begin{aligned} \mathbb{E}\|\bar{u}_s - \nu_s\|_2^2 &= \frac{1}{M_s^2}\sum_{i=1}^{M_s}\mathbb{E}\|u_i^{(s)} - \nu_s\|_2^2 \\ &= \frac{1}{M_s}\mathbb{E}\|u - \nu_s\|_2^2 \le \frac{1}{M_s}\mathbb{E}\|u\|_2^2 \le \frac{4}{M_s}. \end{aligned} \tag{58}$$

By Cauchy–Schwarz, $\mathbb{E}\|\bar{u}_s - \nu_s\|_2 \le 2/\sqrt{M_s}$. Replacing one observation in the sample changes $\bar{u}_s$ by at most $4/M_s$ in Euclidean norm, and therefore changes $\|\bar{u}_s - \nu_s\|_2$ by at most $4/M_s$. Consequently, for every $a > 0$, McDiarmid's inequality (McDiarmid et al., 1989) gives,

$$\Pr\left(\|\bar{u}_s - \nu_s\|_2 > \mathbb{E}\|\bar{u}_s - \nu_s\|_2 + a\right) \le \exp\left(-\frac{M_s a^2}{8}\right). \tag{59}$$

With probability at least $1-\delta$, taking $a = 2\sqrt{2\log(1/\delta)/M_s}$ shows that,

$$\|\bar{u}_s - \nu_s\|_2 \le \frac{2}{\sqrt{M_s}} + \frac{2\sqrt{2\log(1/\delta)}}{\sqrt{M_s}} = \frac{2c_\delta}{\sqrt{M_s}}. \tag{60}$$

Since u_x concatenates the normalized visual and prompt representations, $\|u_x\|_2 \le \sqrt{2}$: $\mathbb{E}|\bar{u}_s^x - \nu_s^x|_2 \le \sqrt{\frac{2}{M_s}}$. Moreover, replacing one observation changes $\bar{u}_s^x$ by at most $2\sqrt{2}/M_s$, and thus changes $\|\bar{u}_s^x - \nu_s^x\|_2$ by at most the same amount. Therefore, McDiarmid's inequality gives:

$$\Pr\left(\|\bar{u}_s^x - \nu_s^x\|_2 > \mathbb{E}\|\bar{u}_s^x - \nu_s^x\|_2 + a\right) \le \exp\left(-\frac{M_s a^2}{4}\right). \tag{61}$$

Choosing $a = \sqrt{2}\sqrt{2\log(1/\delta)/M_s}$ proves:

$$\|\bar{u}_s^x - \nu_s^x\|_2 \le \frac{\sqrt{2}\, c_\delta}{\sqrt{M_s}} \tag{62}$$

with probability at least $1-\delta$. □

Proposition 1. *Fix the representation in Eq. 3. Write $u_y = u_y^{\text{mean}} \oplus u_y^{\text{first}}$, $\nu_s = (\nu_s^x, \nu_s^y)$, and $D_{sq} = \|\nu_s - \nu_q\|_2$, with $D_{sq,x}$ and $D_{sq,y}$ the corresponding input-side and response-side distances. Then*

$$\begin{aligned} D_{sq}^2 &= D_{sq,x}^2 + \|\nu_s^{y,\text{mean}} - \nu_q^{y,\text{mean}}\|_2^2 + \|\nu_s^{y,\text{first}} - \nu_q^{y,\text{first}}\|_2^2 \\ &= D_{sq,x}^2 + D_{sq,y}^2, \qquad D_{sq} \ge \max\{D_{sq,x}, D_{sq,y}\}. \end{aligned} \tag{63}$$

If $\mathcal{P}_s^x = \mathcal{P}_q^x$, then, for every fixed history, any allocation rule observing the arriving task only through an i.i.d. input sample of a common size and independent internal randomness has the same decision law under tasks s and q. Consequently, if such a rule returns a fresh sample of task s to its previous group with probability at least $1-\delta'$, it assigns task q to that group with probability at least $1-\delta'$ as well.

Proof. By Eq. 3, the population mean inherits the same concatenated structure:

$$\nu_s = \nu_s^x \oplus \nu_s^{y,\text{mean}} \oplus \nu_s^{y,\text{first}}, \qquad \nu_s^y = \nu_s^{y,\text{mean}} \oplus \nu_s^{y,\text{first}}. \tag{64}$$

Hence, the difference between the population means of tasks s and q is:

$$\begin{aligned} D_{sq}^2 &= \|\nu_s - \nu_q\|_2^2 \\ &= \|\nu_s^x - \nu_q^x\|_2^2 + \|\nu_s^{y,\text{mean}} - \nu_q^{y,\text{mean}}\|_2^2 + \|\nu_s^{y,\text{first}} - \nu_q^{y,\text{first}}\|_2^2 \\ &= D_{sq,x}^2 + D_{sq,y}^2. \end{aligned} \tag{65}$$

Since both terms are non-negative, $D_{sq} \ge \max\{D_{sq,x}, D_{sq,y}\}$, proving Eq. 63.

We next consider an allocation rule that observes only input features. For the input-only allocation claim, fix all information available from previous tasks and let $U = (u_{x,1}, \ldots, u_{x,m})$ denote the observed input-feature sample. If $\mathcal{P}_s^x = \mathcal{P}_q^x$, then U has the same joint distribution under tasks s and q. Since the rule's internal randomness is independent and identically distributed in both cases, its output has the same conditional distribution. Therefore, if a fresh sample from task s is assigned to its previous group with probability at least $1-\delta'$, then a fresh sample from task q is assigned to that group with the same probability, and hence with probability at least $1-\delta'$. □

Proposition 2. *Under Assumption 4, for any fixed tasks s and q, with probability at least $1-2\delta$,*

$$\|\bar{u}_s - \bar{u}_q\|_2 \ge D_{sq} - 2c_\delta\left(M_s^{-1/2} + M_q^{-1/2}\right). \tag{66}$$

In particular, if $\mathcal{P}_s^x = \mathcal{P}_q^x$ and $D_{sq,y} > 0$, then $D_{sq} = D_{sq,y}$, and, with probability at least $1 - 2\delta$, the empirical task means are distinct whenever

$$\min\{M_s, M_q\} > \frac{16c_\delta^2}{D_{sq,y}^2}. \tag{67}$$

Proposition 2 shows that population-level separation provided by response information is preserved in the empirical task means when sufficiently many samples are used.

Proof. Applying the bound in Eq. 53 separately to tasks s and q. Each bound holds with probability at least $1 - \delta$. Therefore, by the union bound, with probability at least $1 - 2\delta$,

$$\|\bar{u}_s - \nu_s\|_2 \le \frac{2c_\delta}{\sqrt{M_s}}, \qquad \|\bar{u}_q - \nu_q\|_2 \le \frac{2c_\delta}{\sqrt{M_q}}. \tag{68}$$

When both bounds hold, $\bar{u}_s - \bar{u}_q = (\nu_s - \nu_q) + (\bar{u}_s - \nu_s) - (\bar{u}_q - \nu_q)$. Hence, by the triangle inequality,

$$\begin{aligned} D_{sq} = \|\nu_s - \nu_q\|_2 &\le \|\bar{u}_s - \bar{u}_q\|_2 + \|\bar{u}_s - \nu_s\|_2 + \|\bar{u}_q - \nu_q\|_2 \\ \|\bar{u}_s - \bar{u}_q\|_2 &\ge D_{sq} - \|\bar{u}_s - \nu_s\|_2 - \|\bar{u}_q - \nu_q\|_2 \\ &\ge D_{sq} - 2c_\delta \left(M_s^{-1/2} + M_q^{-1/2}\right), \end{aligned} \tag{69}$$

which is Eq. 66.

Now suppose $\mathcal{P}_s^x = \mathcal{P}_q^x$. Then $\nu_s^x = \nu_q^x$, so $D_{sq,x} = 0$ and, by Proposition 1, $D_{sq} = D_{sq,y}$. Let $M_{\min} = \min\{M_s, M_q\}$. Since $M_s, M_q \ge M_{\min}$,

$$M_s^{-1/2} + M_q^{-1/2} \le \frac{2}{\sqrt{M_{\min}}}. \tag{70}$$

Thus Eq. 66 further gives:

$$\|\bar{u}_s - \bar{u}_q\|_2 \ge D_{sq,y} - \underbrace{\frac{4c_\delta}{\sqrt{M_{\min}}}}_{(*)}. \tag{71}$$

The term (*) is positive whenever $\frac{4c_\delta}{\sqrt{M_{\min}}} < D_{sq,y}$, which is equivalent to $M_{\min} > \frac{16c_\delta^2}{D_{sq,y}^2}$. Therefore, $\|\bar{u}_s - \bar{u}_q\|_2 > 0$, and $\bar{u}_s \ne \bar{u}_q$. □

B.7 Relation between task statistics and adapter distances

Assumption 5. For the tasks under consideration, there exists $L > 0$ such that

$$\|\theta_s^\star - \theta_q^\star\|_2 \le L\|\nu_s - \nu_q\|_2 \qquad \text{for all } s, q. \tag{72}$$

This assumption relates distances between the frozen task representations used by PACE to distances between the corresponding optimal adapters. For finitely many tasks with distinct ν_s, some L always satisfies it, so its content is whether L is small enough that grouping tasks with nearby ν_s keeps Δ_k small. We do not verify this, and the counterfactual audit in Tab. **??** is evidence against it on CoIN: task pairs that are closer in the frozen representation lose more accuracy when they share an expert. The proposition below is therefore a conditional statement. Theorems 1 and 2 do not use Assumption 5.

Proposition 3. *For any $\delta \in (0, 1)$, under Assumptions 4 and 5, with probability at least $1 - \delta$, the following holds simultaneously for every group $\mathcal{G}_k$: if $\{r_s : s \in \mathcal{G}_k\}$ lies in a Euclidean ball of radius η_k, then*

$$\Delta_k \le 2L \left(\eta_k + \frac{2c_{\delta/T}}{\sqrt{M_{k,\min}}}\right), \qquad M_{k,\min} = \min_{s \in \mathcal{G}_k} M_s. \tag{73}$$

Proof. For each fixed task s, apply the first Euclidean concentration bound in Eq. 53 with δ replaced by δ/T. Since $r_s = \bar{u}_s$,

$$\Pr\left(\|r_s - \nu_s\|_2 \le \frac{2c_{\delta/T}}{\sqrt{M_s}}\right) \ge 1 - \frac{\delta}{T}. \tag{74}$$

By the union bound over the T tasks,

$$\Pr\left(\|r_s - \nu_s\|_2 \le \frac{2c_{\delta/T}}{\sqrt{M_s}} \text{ for every } s \le T\right) \ge 1 - \delta. \tag{75}$$

Under these inequalities, fix any group $\mathcal{G}_k$. By the hypothesis that $\{r_s : s \in \mathcal{G}_k\}$ lies in a Euclidean ball of radius η_k, there exists a center from which every r_s in the group has distance at most η_k. Therefore, for every $s, q \in \mathcal{G}_k$, the triangle inequality gives

$$\|r_s - r_q\|_2 \le 2\eta_k. \tag{76}$$

Using r_s and r_q as intermediate points between the population means, we obtain:

$$\begin{aligned}\|\nu_s - \nu_q\|_2 &\le \|\nu_s - r_s\|_2 + \|r_s - r_q\|_2 + \|r_q - \nu_q\|_2 \\ &\le \frac{2c_{\delta/T}}{\sqrt{M_s}} + 2\eta_k + \frac{2c_{\delta/T}}{\sqrt{M_q}}.\end{aligned} \tag{77}$$

Since $M_s, M_q \ge M_{k,\min}$ for all $s, q \in \mathcal{G}_k$, it follows that:

$$\|\nu_s - \nu_q\|_2 \le 2\eta_k + \frac{4c_{\delta/T}}{\sqrt{M_{k,\min}}} = 2\left(\eta_k + \frac{2c_{\delta/T}}{\sqrt{M_{k,\min}}}\right).$$

Therefore, for every $s, q \in \mathcal{G}_k$, Assumption 5 gives:

$$\|\theta_s^\star - \theta_q^\star\|_2 \le 2L\left(\eta_k + \frac{2c_{\delta/T}}{\sqrt{M_{k,\min}}}\right). \tag{78}$$

Taking the maximum over $s, q \in \mathcal{G}_k$ yields:

$$\Delta_k \le 2L\left(\eta_k + \frac{2c_{\delta/T}}{\sqrt{M_{k,\min}}}\right). \tag{79}$$

Because the concentration inequalities hold simultaneously for all tasks, the conclusion applies simultaneously to every group, including groups determined from the observed summaries, with probability at least $1 - \delta$. □

C Implementation Details

C.1 Benchmarks and training protocol

We evaluate on three MCIT benchmarks, including CoIN-CL (Chen et al., 2024), UCIT (Guo et al., 2025a), and TriGap (Xie et al., 2026), and on the domain continual learning (DCL) benchmark from MLLM-CL (Zhao et al., 2026b). All CoIN-CL baselines are reproduced with Qwen2.5-VL-7B-Instruct under a shared training protocol. All UCIT baselines are reproduced with LLaVA-1.5-7B under a shared training protocol. TriGap and DCL baseline results are adopted directly from SAME and MLLM-CL, respectively, both using LLaVA-1.5-7B. By default, we use the same task orders as described in these benchmarks.

All experiments use AdamW with cosine learning-rate decay, a warmup ratio of 0.03, zero weight decay, LoRA dropout of 0.05, and DeepSpeed ZeRO-2. For CoIN-CL, we set the maximum sequence length to 4096 and constrain the image resolution to between 200,704 and 1,003,520 total pixels. For LLaVA-based experiments, we use a maximum sequence length of 2048, CLIP ViT-L/14 at 336 pixels, and the Vicuna conversation template. Tab. 5 summarizes the remaining benchmark-specific settings.

Table 5: PACE configurations. Slots are preallocated capacity; K counts experts used after the final task. Total rank is divided equally among slots. "All" targets the decoder's q, k, v, o, gate, up, and down projections; "FFN" targets gate, up, and down.

Setting	CoIN-CL	UCIT	TriGap	DCL
Backbone (7B)	Qwen2.5-VL	LLaVA-1.5	LLaVA-1.5	LLaVA-1.5
tasks / experts (K)	8 / 6	6 / 5	10 / 10	5 / 4
Total rank / rank per expert	192 / 24	96 / 16	80 / 8	160 / 32
LoRA α_L	256	192	160	320
Injection scope	All	All	FFN	All
Adapter learning rate	2×10^{-4}	2×10^{-4}	2×10^{-4}	10^{-4}
Epochs per task	1	1	1	1 / 3 / 1 / 2 / 1
Effective batch size	112	128	24	64
d / d_x	14,336 / 7,168	16,384 / 8,192	16,384 / 8,192	16,384 / 8,192
$\log\alpha = \lambda_0 d$	15,960.1	18,240.1	18,240.1	18,240.1

C.2 PACE IMPLEMENTATION

Task statistics and allocation. For each task, we draw 256 distinct training examples uniformly at random without replacement and also use them for adaptation. On CoIN, these examples constitute only 0.15%–2.01% of each task's training set (12,726 to 165,348 examples), so sampling without replacement closely approximates i.i.d. sampling from the task distribution. From these examples, we compute 4 frozen feature components in Eq. 3: mean projected image tokens, mean prompt-token embeddings, mean response-token embeddings, and the first response-token embedding. Each component is independently $L2$-normalized. Prompt lines occurring in more than 20% of a file are removed, with a fallback that preserves a nonempty prompt. At inference, the same unlabeled preprocessing is applied using only the visual and prompt components. The bank stores $\beta_t = (\bar{u}_t, v_t, M_t)$ and the expert assignment z_t. Note that no training examples are retained.

We set $\kappa_0 = 0.01$, $a_0 = 2$, and the variance floor to 10^{-7}. At boundary t, $G_{0,t}$ uses the mean and unbiased coordinate-wise variance of previous task means. When only one previous task is available, its within-task variance is used instead. Each task mean contributes one NIG observation. Allocation follows Eq. 5, including the CRP occupancy term $\log n_k$, with $\log\alpha = \lambda_0 d$ fixed before the first task (Appendix A.2). Task 1 directly initializes expert 1.

Adaptation and inference. The vision encoder and base language-model weights remain frozen. Only the selected LoRA expert receives gradients and is forced for all tokens during training and generation. Expert 1 uses randomly initialized A and zero-initialized B. A newly activated expert initializes its A and B factors by separately averaging those of the active experts, whereas expert reuse updates the assigned expert in place. LoRA scaling is $\alpha_L/r_{\text{total}}$.

At inference, Eq. 6 scores task prototypes using a shared diagonal covariance computed from the mean within-task variance of the stored input features. The selected task is mapped to its stored expert assignment, which is then applied with weight one. Thus, inference does not require a trainable router or test-answer information.

C.3 BASELINE IMPLEMENTATIONS

For CoIN-CL and UCIT, reproduced baselines use the same backbone, task order, training splits, optimizer schedule, and total LoRA rank. CoIN-CL uses rank 192 with $\alpha_L = 256$, while UCIT uses rank 96 with $\alpha_L = 192$. Single-adapter methods use the full rank, while expert-based methods divide it across 8 and 6 experts, respectively. All methods apply LoRA to the seven decoder projections in Tab. 5. For O-LoRA and SAME, this uses a broader set of target layers than in their original implementations. UCIT additionally trains the shared projector. Tab. 6 summarizes the method-specific settings.

Zero-shot uses the unadapted backbone, while joint training performs instruction tuning using the union of data from all tasks simultaneously. On UCIT, joint training uses one LoRA adapter with total rank 96, trained for one epoch on the combined data from all 6 tasks. For TriGap, we

Table 6: Baseline settings on CoIN-CL and UCIT. Values separated by a slash denote CoIN-CL / UCIT.

Method	Implementation and settings
FT-LoRA	Sequential fine-tuning of one LoRA adapter.
EWC	Diagonal Fisher penalty of 3×10^5 / 10^4, estimated from 10% / 1% of training examples.
O-LoRA	One subspace per task with previous subspaces frozen, orthogonality coefficient 0.5, and eight rank-24 / six rank-16 subspaces.
MoELoRA	Dense softmax routing over all experts without auxiliary load balancing.
CL-MoE	Dense token routing with momentum merging after each task, using coefficient 0.7 over all eight / six experts.
HiDe-LLaVA	Top-layer routing with frozen CLIP image/text prototypes, temperature 0.1, equal modality weights, and lower-layer adapter fusion.
SMoLoRA	Two expert groups (4+4 / 3+3), top-1 selection per group, and 768-dimensional instruction features using frozen CLIP text features instead of Sentence-BERT.
DISCO	DKO and SSA with threshold 0.9, producing five groups on each benchmark.
SEFE	RegLoRA coefficient 2500, 2% key-element fraction, and fresh adapters trained over merged predecessors. CoIN-ASD augmentation is omitted.
SAME	Energy threshold 0.9, damping 0.9, freezing threshold 0.1, window 3, up to 64 spectral components, covariance refresh every 20 steps, and top-2 inference.

adopt SAME's reproduced results for FT-LoRA, Replay-LoRA, MoE-LoRA, HiDe-LLaVA, ModalPrompt, CL-MoE, and SAME. These methods use one epoch per task, a learning rate of 2×10^{-4}, rank-8 FFN adapters, and an effective batch size of 24 on a single NVIDIA H100.

For DCL, we retain the published variants with and without replay. MR-LoRA uses isolated domain adapters and a router trained with 20 stored examples per task. PACE follows the MLLM-CL setting with rank 32 per expert, learning rate 10^{-4}, batch size 64, and domain epochs 1/3/1/2/1. Other adopted baselines retain their published ranks and batch sizes with learning rate 2×10^{-5}.

C.4 Evaluation and metrics

Following the original setup from CoIN-CL (Guo et al., 2025a), all methods on that benchmark use the same fixed 25% sample of each configured test split with seed 42. UCIT evaluates all 3000 examples per task, while TriGap and DCL use their full test splits.

CoIN-CL uses task-specific metrics: option accuracy for ScienceQA, VQA soft accuracy for TextVQA, normalized class matching for ImageNet, string matching for GQA, exact match for VizWiz, VQAv2, and OCR-VQA, and IoU > 0.5 for Grounding. UCIT uses normalized exact match for its four QA/classification tasks. For captioning, it averages BLEU-1–4, METEOR, ROUGE-L, and CIDEr. TriGap follows Prism's exact-match and ANLS implementations, including unthresholded ANLS for DocVQA and InfographicVQA. DCL uses its released domain-specific scorers: exact match for finance, answer containment for RS, Med, and AD, and question-dependent scoring for science.

Let $R_{t,s}$ be the score on task s after training task t. All task scores use a 0–100 scale. We compute:

$$\mathrm{AA} = \frac{1}{T}\sum_{s=1}^{T} R_{T,s}, \qquad \mathrm{BWT} = \frac{1}{T-1}\sum_{s=1}^{T-1}(R_{T,s} - R_{s,s}),$$

$$\mathrm{FGT} = \frac{1}{T-1}\sum_{s=1}^{T-1}\left(\max_{s\le j\le T} R_{j,s} - R_{T,s}\right),$$

On DCL, mean final accuracy (MFN) equals AA, mean finetuning accuracy is $\mathrm{MFT} = T^{-1}\sum_{t=1}^{T} R_{t,t}$, and mean average accuracy is $\mathrm{MAA} = T^{-1}\sum_{t=1}^{T} t^{-1}\sum_{s=1}^{t} R_{t,s}$. Published baseline aggregates are retained as reported. MFT is undefined for zero-shot; the Oracle's independently trained experts do not form a continual trajectory.

D Additional Experiments

D.1 Results on TriGap Benchmark

Tab. 7 shows that PACE achieves the best overall performance on TriGap, outperforming the state-of-the-art baseline, SAME, in average score. Notably, some continual-learning baselines even fall below the zero-shot lower bound on individual tasks, e.g., FT-LoRA and Replay-LoRA on PM-CVQA, and FT-LoRA on Roadside, indicating that sequential adaptation can degrade pretrained capabilities. In contrast, PACE consistently outperforms zero-shot across all tasks while remaining competitive on tasks where other methods perform best, demonstrating robust adaptation across the diverse TriGap tasks.

Table 7: **Comparison on the TriGap benchmark.** All baseline and zero-shot results are adopted from SAME using LLaVA-1.5-7B. We retain the original ChemVQA scoring, which joins annotations by non-unique question IDs, to match the published baselines. The best and second-best results are highlighted in **bold** and underline, respectively.

Method	PMCVQA	DocVQA	ChartQA	IconQA	InfoVQA	ArxivQA	Roadside	ChemVQA	FloodNet	CLEVR	Average
Zero-shot	35.40	12.68	9.36	19.27	5.06	53.77	7.40	5.30	47.41	20.37	21.60
FT-LoRA	34.20	23.32	9.84	37.07	23.53	83.83	7.00	12.70	80.31	60.27	37.21
Replay-LoRA	33.70	33.95	14.00	46.67	28.97	75.57	9.40	15.90	73.81	58.80	39.08
MoE-LoRA	39.03	37.49	12.44	43.43	35.17	90.90	7.93	20.70	**90.41**	**67.00**	44.45
HiDe-LLaVA	37.00	33.20	10.52	41.97	24.09	79.20	7.73	11.17	57.39	23.00	32.53
ModalPrompt	38.23	38.23	11.92	44.73	37.37	84.47	10.13	12.43	71.52	52.50	40.15
CL-MoE	40.53	36.79	13.72	52.70	32.27	**93.00**	7.77	18.33	80.09	65.90	44.11
SAME	41.60	43.87	17.56	64.03	39.57	90.46	10.83	21.77	81.09	54.50	46.53
PACE (Ours)	**44.37**	**44.75**	**19.92**	**68.70**	**40.24**	89.43	**12.93**	**27.13**	80.80	62.40	**49.07**

D.2 Results on MLLM-CL Benchmark

In this experiment, we adopt the LLaVA-1.5 baseline results from MLLM-CL, including its replay variants. Tab. 8 shows that PACE achieves the strongest overall performance among replay-free methods on MLLM-CL, obtaining the best MFN, MFT, MAA, and BWT. In particular, its near-zero BWT indicates that PACE largely preserves previously learned domains, whereas other replay-free baselines suffer substantially greater forgetting. PACE also closely approaches the per-domain performance of the Oracle, despite using only four experts, no replay samples, and no task identity at inference. Moreover, PACE remains competitive with MR-LoRA, which stores 20 examples per task and uses five experts, while requiring substantially fewer populated LoRA parameters. These results demonstrate that PACE can achieve strong domain adaptation and retention without replay or inference-time task labels.

D.3 Additional Analysis

Response Behavior and Forgetting. To better understand the source of forgetting beyond aggregate performance, we examine whether continual adaptation changes task-specific response behavior, particularly whether the model answers a question or declines to answer. We further investigate whether such behavioral changes account for the loss of previously correct answers. Fig. 5(a) shows that learning VizWiz can introduce non-answer responses on earlier tasks for many baselines, reaching 16% on TextVQA for HiDe-LLaVA and 33% on ImageNet for SAME. In contrast, PACE limits these responses to 2% on TextVQA and 0% on the other evaluated tasks, indicating limited transfer of this behavior across tasks. Fig. 5(b) further shows that FT-LoRA's non-answer rate on TextVQA rises sharply after learning VizWiz and later falls to nearly zero, whereas PACE remains stable throughout training. Fig. 5(c) shows the opposite effect on VizWiz: subsequent tasks substantially suppress the non-answer behavior learned by FT-LoRA and the learned-router MoE, while PACE preserves it. Finally, Fig. 5(d) connects these behavioral changes to forgetting: switches between answering and declining to answer account for most forgotten VizWiz answers in several baselines, whereas PACE loses no previously correct answers in either evaluated transition. Overall, PACE better preserves both task-specific response behavior and answer correctness throughout continual adaptation.

Assignment Score and Cross-Task Interference.

Table 8: **Comparison on MLLM-CL DCL with LLaVA-1.5-7B.** Baseline results are adopted from MLLM-CL. Oracle trains one independent adapter per domain and uses the task ID at inference. Parameter counts include populated LoRA weights, including MR-LoRA's router adapter, and exclude unused slots, projectors, and non-LoRA routing weights. "w/ Replay" uses 20 examples per task; MR-LoRA also stores 20 per task for router training. "–" denotes an inapplicable metric. The best and second-best results are highlighted in **bold** and underline, respectively, excluding Zero-shot and Oracle.

Method	No. experts	Expert rank	Populated LoRA params	No. replay samples/task	RS	Med	AD	Sci	Fin	MFN	MFT	MAA	BWT
Zero-shot	–	–	–	0	32.29	28.28	15.59	35.55	62.56	34.85	–	–	–
Oracle	5	32	399.8M	0	81.06	65.83	54.17	56.86	91.14	–	69.81	–	–
FT-LoRA	1	32	80.0M	0	69.65	41.59	25.43	40.88	87.45	53.00	64.98	61.13	-14.97
FT-LoRA w/ Replay	1	32	80.0M	20	76.54	50.27	43.01	43.32	89.85	60.60	66.32	64.72	-7.15
MR-LoRA	5	32	479.7M	20	**80.87**	65.32	**54.12**	**56.71**	91.12	**69.63**	69.64	**71.06**	**-0.01**
SEFE	1	32	80.0M	0	77.26	50.37	37.21	40.87	86.82	58.51	65.01	63.63	-8.13
SEFE w/ Replay	1	32	80.0M	20	78.43	52.85	46.21	47.76	89.33	62.92	66.89	66.51	-4.97
DISCO	5	32	399.8M	0	76.03	45.20	43.79	42.33	88.95	59.26	64.43	63.35	-6.46
DISCO w/ Replay	5	32	399.8M	20	77.78	46.25	50.45	49.51	89.71	62.74	65.27	64.92	-3.17
CL-MoE	5	32	399.8M	0	71.34	46.84	26.33	41.17	88.74	54.88	66.06	61.79	-13.96
CL-MoE w/ Replay	5	32	399.8M	20	76.58	52.31	39.65	45.64	90.21	60.88	66.65	64.95	-7.22
MoELoRA	5	7	87.4M	0	77.54	41.85	27.62	40.13	86.75	54.78	64.94	61.76	-12.70
MoELoRA w/ Replay	5	7	87.4M	20	77.63	49.54	39.08	41.04	89.21	59.30	66.24	64.81	-8.68
HiDe-LLaVA	5	32	399.8M	0	74.31	48.95	33.21	38.54	81.55	55.31	60.77	60.68	-6.82
HiDe-LLaVA w/ Replay	5	32	399.8M	20	74.80	42.29	34.03	38.01	79.22	53.67	60.83	61.81	-8.95
O-LoRA	5	32	399.8M	0	74.64	44.42	30.02	41.47	87.15	55.54	65.16	62.12	-12.03
O-LoRA w/ Replay	5	32	399.8M	20	76.94	41.17	34.18	39.61	83.22	55.02	60.49	60.73	-6.83
PACE (Ours)	**4**	32	**319.8M**	**0**	79.68	**65.80**	53.86	56.36	**91.44**	69.43	**69.77**	70.78	-0.43

Fig. 6 examines how allocation signals relate to cross-task interference. At each task boundary, we counterfactually train the arriving task for 100 optimizer steps on every existing expert and re-evaluate the earlier tasks assigned to that expert, yielding 28 measurements from 21 updates. The arriving-task loss before adaptation is associated with the resulting accuracy change, whereas the PACE assignment score does not directly predict the magnitude of interference. Notably, VQAv2 and GQA cause the two largest drops when trained on ImageNet's expert (-54.3 and -52.5 points), yet PACE assigns neither task to that expert. These results suggest that PACE's probabilistic allocation score determines expert reuse without serving as a direct estimator of optimization-induced cross-task interference.

Effect of Assignment Rules.

Fig. 7 shows that the assignment criterion strongly affects how tasks are grouped, even when a similar number of experts is used. PACE produces 6 experts with an ARI of 1.00, whereas cosine- and Euclidean-prototype rules yield substantially different groupings (ARI from -0.09 to 0.46) with six or seven experts. This indicates that controlling only the number of experts is insufficient; which tasks are assigned to share an expert is critical, highlighting the importance of PACE's probabilistic assignment rule.

We further provide additional analysis in Appendix D.3 to examine how continual learning changes task-specific answering behavior and whether such changes contribute to forgetting.

Expert Allocation Behavior. Fig. 8 illustrates how PACE converts task statistics into expert assignment decisions. TextVQA reuses the ScienceQA expert with

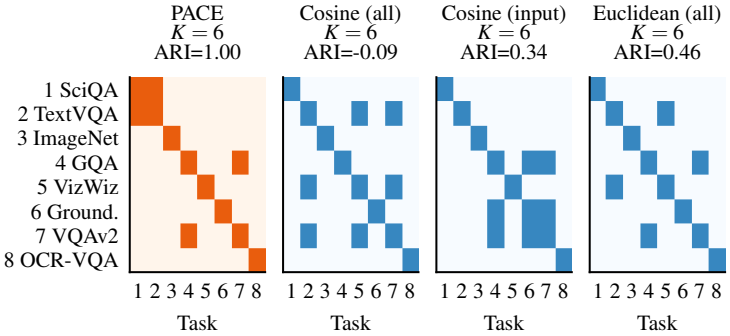

Figure 7: **Effect of different assignment rules on task-to-expert grouping on CoIN.** ARI measures agreement with the assignments produced by PACE.

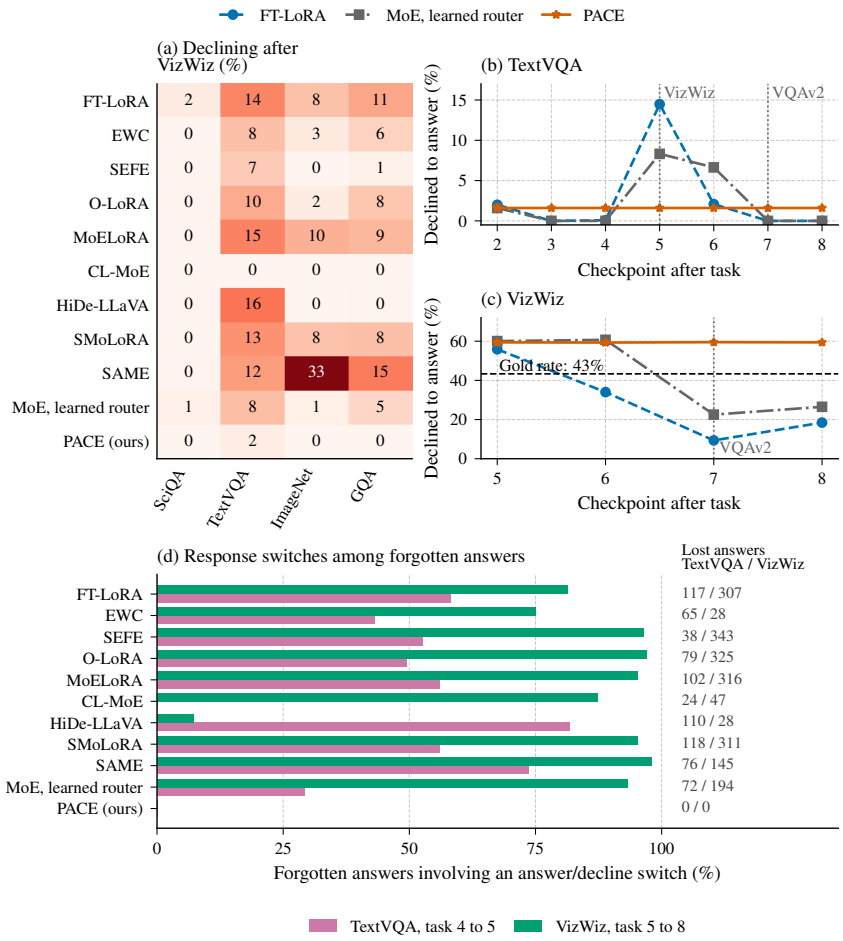

Figure 5: **Response behavior during continual learning on CoIN.** (a) Frequency of declining to answer on earlier tasks after learning VizWiz. (b–c) Changes in this behavior on TextVQA and VizWiz across training checkpoints. (d) Percentage of forgotten answers involving a switch between answering and declining to answer; annotations indicate the total number of previously correct answers lost.

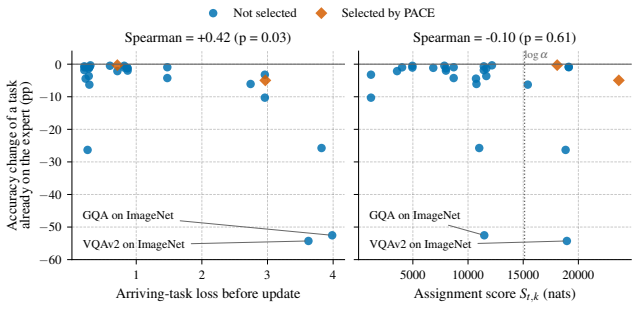

Figure 6: **Assignment signals and cross-task interference on CoIN.**

a margin of +3.0k, while VQAv2 joins the GQA expert with the largest margin of +8.5k. The other tasks fall below the assignment threshold and create new experts, resulting in 6 experts for eight tasks. This allocation shows that PACE shares parameters only between tasks with sufficiently similar statistics.

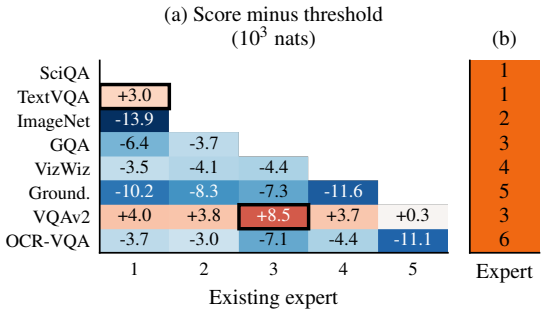

Figure 8: **PACE expert-allocation decisions on CoIN.** Left: assignment scores relative to the expansion threshold as each task arrives. Right: the resulting task-to-expert assignment.

Table 9: **Adapter cost.** Parameter cost of Dense LoRA, MoELoRA, and PACE.

Method	Experts	Expert rank	Number of parameters (M)			
			Updated per task	Populated after 8 tasks	Applied per token	Total allocated
Dense LoRA	1	192	484.4	484.4	484.4	484.4
MoELoRA	8	24	484.4	484.4	484.4	484.4
PACE	6	24	**60.6**	**363.3**	**60.6**	484.4

Table 10: **Inference latency.** Runtime comparison between MoELoRA and PACE.

Method	Selection stage	Selection + generation	Selection + generation (16 tokens)
MoELoRA	–	1722.5	4849.8
PACE	923.7	2120.8	**2692.2**

Computational Cost and Inference Efficiency. Tabs. 9 and 10 show that PACE substantially reduces the cost of expert adaptation and generation. As shown in Tab. 9, PACE updates and applies only 60.6M LoRA parameters per task/token, compared with 484.4M for Dense LoRA and MoELoRA, while populating 363.3M LoRA parameters after 8 tasks. Tab. 10 shows that this efficiency comes with a one-time expert-selection overhead (923.7 ms): PACE is slightly slower for short generation (2120.8 vs. 1722.5 ms), but becomes substantially faster for 16-token generation (2692.2 vs. 4849.8 ms), reducing total latency by 44.5%. This shows that the selection overhead becomes less significant as generation length increases, while single-expert execution lowers per-token computation.

Plasticity and Stability Analysis. We further report learning accuracy, $\mathrm{LA} = \frac{1}{T}\sum_{t=1}^{T} R_{t,t}$, which measures plasticity by evaluating each task immediately after it is learned. Since $\mathrm{AA} = \mathrm{LA} + \frac{T-1}{T}\mathrm{BWT}$, the gap to joint training, $\mathrm{AA}^\star - \mathrm{AA}$, decomposes exactly into $\mathrm{AA}^\star - \mathrm{LA}$ (Chaudhry et al., 2018) and the forgetting term $-\frac{T-1}{T}\mathrm{BWT}$.

Tab. 11 separates performance into plasticity, measured by learning accuracy (LA), and stability, measured by BWT and FGT. On CoIN, PACE maintains strong plasticity (79.50 LA) while achieving the best retention among continual-learning methods (-0.61 BWT and 0.71 FGT), resulting in 78.97 AA, only 1.01 points below joint training. In contrast, methods with higher LA, such as FT-LoRA, MoELoRA, and SMoLoRA, suffer substantially more forgetting. On UCIT, PACE achieves the highest LA (68.95) and lowest FGT (1.43), yielding the highest final AA (68.19) among continual-learning methods. These results show that PACE preserves new-task learning while substantially reducing forgetting, providing a favorable plasticity–stability balance across both benchmarks.

E Limitations

PACE relies on a frozen multimodal representation to summarize tasks. The quality of the resulting assignments depends on how well that representation reflects which tasks can share an adapter, and the audit in Appendix D.3 shows that on CoIN, proximity in the frozen representation does not predict the interference caused by sharing. Future work could develop more scalable mechanisms for retrieving and comparing stored task statistics to maintain efficient allocation and inference as the task stream grows.

Finally, our theoretical analysis adopts an overparameterized linear-regression model to isolate the effects of expert sharing on forgetting and generalization. This abstraction enables closed-form characterization of the sharing trade-off, but does not model the full nonlinear optimization dynamics of multimodal large language models. Developing theory that more closely reflects large-scale multimodal adaptation is an important direction for future work.

Table 11: **Plasticity and stability on CoIN and UCIT.** LA is the score on each task right after it is learned; BWT and FGT measure retention; $\mathrm{AA} = \mathrm{LA} + \frac{T-1}{T}\mathrm{BWT}$. The best and second-best results are highlighted in **bold** and underline, respectively.

	CoIN				UCIT			
Method	**LA** ↑	**BWT** ↑	**FGT** ↓	**AA** ↑	**LA** ↑	**BWT** ↑	**FGT** ↓	**AA** ↑
Zero-shot	–	–	–	65.15	–	–	–	31.91
Joint training	–	–	–	79.98	–	–	–	71.20
FT-LoRA	**81.19**	-12.96	12.96	69.85	67.69	-25.82	26.39	46.17
EWC	76.64	-10.66	11.23	67.31	66.77	-5.85	9.11	61.89
O-LoRA	80.03	-6.37	6.81	74.46	65.17	-6.53	7.35	59.73
MoELoRA	<u>80.72</u>	-17.06	17.06	65.79	66.76	-10.95	13.94	57.64
CL-MoE	71.39	-4.60	5.19	67.37	59.30	<u>-0.27</u>	<u>4.04</u>	59.07
HiDe-LLaVA	73.49	<u>-3.84</u>	4.46	70.14	62.85	-0.55	4.22	62.39
SMoLoRA	80.69	-14.75	14.75	67.79	66.66	-8.70	11.73	59.41
DISCO	79.27	-12.31	12.38	68.50	66.42	**0.20**	4.95	<u>66.58</u>
SEFE	79.84	-3.98	<u>4.19</u>	<u>76.36</u>	<u>67.89</u>	-7.41	7.41	61.72
SAME	77.28	-8.88	9.24	69.51	49.18	-6.23	7.16	43.99
PACE (Ours)	79.50	**-0.61**	**0.71**	**78.97**	**68.95**	-0.92	**1.43**	**68.19**