PREFMOE: MITIGATING PREFERENCE COLLAPSE IN PERSONALIZED MULTIMODAL LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Personalized multimodal large language models (MLLMs) aim to generate user-specific responses, but existing methods mainly rely on profile-level information and overlook diverse user preferences. We identify group preference collapse, where multi-user personalized MLLMs become insensitive to individual preferences and drift toward dominant population-level choices due to suppressed preference signals and unreliable preference use during generation. We propose PrefMoE, a preference-centric framework that separates stable profile information from preference-related representations. PrefMoE decomposes preferences into shared prototypes and personalized residuals, preserves individualized residuals with imbalance-aware learning, counterfactual pseudo-user augmentation, and residual decorrelation, and routes profile and preference factors through separate LoRA adaptation paths. Experiments across multiple MLLM backbones show that PrefMoE improves preference-sensitive personalization while substantially reducing preference collapse.

1 INTRODUCTION

Multimodal Large Language Models (MLLMs) (Alayrac et al., 2022; Li et al., 2023; Dai et al., 2023; Zhu et al., 2024) have shown strong performance across a wide range of multimodal tasks, including visual question answering (Antol et al., 2015; Fang et al., 2025; Wang et al., 2025), visual grounding (Tang et al., 2026), and multimodal reasoning (Huang et al., 2025). Building on this progress, personalized MLLMs have recently attracted increasing attention (Alaluf et al., 2024; Cohen et al., 2022; Yeh et al., 2023; Nguyen et al., 2025), aiming to generate responses tailored to a target user by conditioning on user-related information (Seifi et al., 2026). Existing efforts mainly incorporate user-related profile signals (Alaluf et al., 2024; Pi et al., 2025; Pham et al., 2025; Seifi et al., 2026), such as portraits, demographic cues, or textual descriptions, to ground responses in user identity and attributes (Pi et al., 2025). *However, these methods primarily model who the user is, while paying much less attention to what the user prefers.*

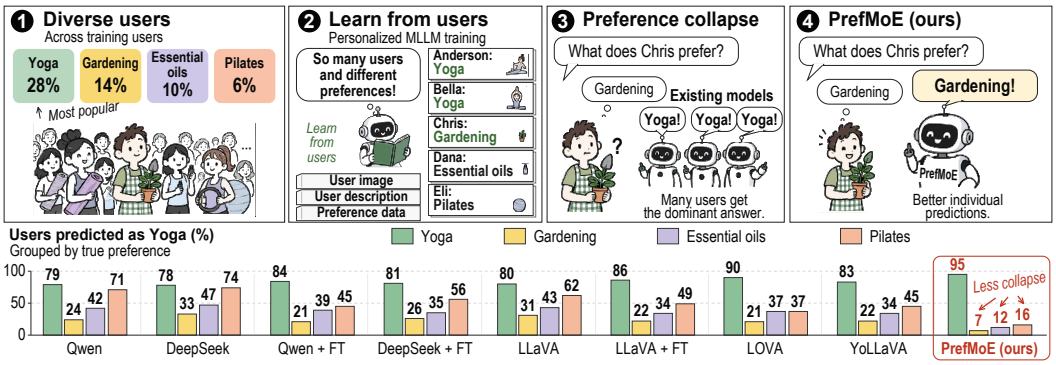

Figure 1: Illustration of *group preference collapse*. Although users have diverse preferences, existing personalized MLLMs tend to predict the dominant preference across users ("Yoga") rather than reflecting the target user's specific preference. Bars indicate the percentage of users from each ground-truth preference group whose Top-1 preference is the dominant preference ("Yoga").

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.

Preference personalization differs from profile-based personalization because preferences describe context-dependent user choices rather than relatively stable identity cues (Rendle et al., 2009). Users with similar profiles may still prefer different clothing styles, travel plans, or food options under the same visual query. This distinction motivates dedicated mechanisms that go beyond profile modeling to preserve individual preference differences and selectively use the preference cues relevant to the current image-question pair. Despite its practical importance, preference-level personalization remains largely underexplored.

Under joint multi-user learning, personalized MLLMs may exhibit a failure mode that we term *group preference collapse*, where predictions favor population-dominant choices over individual user preferences. As illustrated in Fig. 1, this behavior persists even when user preferences are explicitly provided as textual conditions and the models are fine-tuned. We attribute group preference collapse to two key failures. First, individualized preference signals can be suppressed during user representation learning, as sparse and low-frequency preferences are easily absorbed by dominant population-level patterns. Second, preference use during generation can be unreliable. Even when preference information is provided, the model may fail to select the preference cue relevant to the current image-question pair and instead follow generic or population-dominant reasoning patterns. Together, these failures make responses drift toward broadly plausible group-level preferences rather than fine-grained user-specific choices.

To address these failures, we propose *PrefMoE*, a preference-centric personalization framework for MLLMs. PrefMoE first learns a structured user state that separates stable profile cues from preference-related information. It decomposes preference representations into shared population-level prototypes and personalized residuals, allowing dominant group tendencies and user-specific deviations to be modeled separately. To prevent minority preferences from being absorbed by shared patterns, we preserve individualized residuals through imbalance-aware contrastive learning, counterfactual preference augmentation, and decorrelation. In addition, PrefMoE introduces a factorized hierarchical MoE router to ensure that learned preferences are used during generation: preference residuals activate preference-specific LoRA experts, profile cues activate a profile-aware adaptation branch, and a query-dependent router adaptively fuses these updates for the current visual question. In this way, PrefMoE not only preserves fine-grained preference diversity in user representations, but also turns structured user states into factorized computation paths for preference-sensitive response generation. Our work makes three main contributions:

(1) We identify group preference collapse as a critical failure mode in multi-user personalized MLLMs, where dominant population-level preferences suppress user-specific preference signals even when explicit user information is available.

(2) We propose PrefMoE, a preference-centric framework that combines residual-based preference factor learning with factorized hierarchical routing to preserve individualized preference offsets and activate profile- and preference-aware adaptation paths during generation.

(3) Experiments across multiple MLLM backbones show that PrefMoE improves preference accuracy and reduces preference collapse. On LLaVA-1.5-7B under the 0-turn setting, PrefMoE improves preference accuracy from 44.13% to 67.33% and reduces collapse from 34.25% to 12.33% compared with full fine-tuning.

2 Related Work

MLLM Personalization. MLLM personalization associates user-specific concepts and information with visual inputs through learnable embeddings, concept tokens, prompts, and in-context conditioning (Nguyen et al., 2024; Pi et al., 2025). Recent work expands concept learning and representations (An et al., 2025), personal memory retrieval (Xu et al., 2025), and streaming video understanding (Zheng et al., 2026). M2A also maintains and updates multimodal memories, including user preferences, during long-term interactions (Feng et al., 2026). These approaches support concept acquisition, memory access, and temporal interaction. We focus on preference diversity under joint multi-user learning: preserving low-frequency user-specific preferences and preventing responses from drifting toward population-dominant choices.

Preference Learning. Preference learning models preferred outputs from comparisons, rankings, or choice-based feedback (Bradley & Terry, 1952). Human feedback also underlies LLM alignment methods including RLHF (Ouyang et al., 2022), DPO (Rafailov et al., 2023), contrastive preference learning (Hejna et al., 2024), and KTO (Ethayarajh et al., 2024). Learning shared or aggregated

preferences (Ge et al., 2024) differs from jointly modeling users with heterogeneous preferences. Under multi-user learning, dominant preference signals can suppress individual or minority choices. Our work investigates this failure in personalized MLLMs and combines residual preference learning with factorized routing to preserve user-specific preferences and use them during generation. An extended discussion is provided in Appendix A.

3 MLLM Personalization and Group Preference Collapse

We study joint multi-user personalization of MLLMs, where a shared model learns from the available information of multiple users and generates personalized responses for these users at inference time. For each user u_i, we denote the available personalization information as $\mathcal{X}_i = \{I_i, D_i, \mathcal{P}_i\}$, where I_i and D_i denote user-related image information and textual profile description, and $\mathcal{P}_i$ denotes the user preference set over different preference facets. Here, preference facets refer to semantic dimensions of user preferences, such as fashion, lifestyle, travel, shopping, and entertainment. The profile information I_i and D_i captures relatively stable identity-related cues, while $\mathcal{P}_i$ captures user-specific tendencies across these facets and answers what the user prefers.

We take visual question answering as an example downstream task. Given an input image V_n, a question Q_n, for the target user u_i, the goal is to learn a personalized MLLM that is able to predict the user-specific answer $A_{n,i}$:

$$\mathcal{L}_{n,i}^{\mathrm{vqa}} = \mathrm{CE}\big(\mathrm{MLLM}(V_n, Q_n; \mathcal{X}_i), A_{n,i}\big), \tag{1}$$

where CE denotes the answer cross-entropy prediction loss. This formulation defines the desired user-conditioned prediction, but does not prescribe how the user condition should be implemented.

A straightforward implementation is to encode the user's profile and preference information as tokens and concatenate them with the visual and question tokens. However, explicitly providing user information does not guarantee that individual preferences are preserved and used appropriately. *Two challenges* arise under joint multi-user learning. First, during representation learning, low-frequency preference signals may be overshadowed by dominant population patterns, making individual preferences difficult to distinguish. Second, during generation, the model may rely on visual priors or common answer patterns rather than use the preference cues relevant to the current image-question pair. These challenges can lead to responses that favor population-dominant choices over the target user's preferences. We term this systematic tendency *group preference collapse*. Operationally, a collapse error falls outside the target user's acceptable preferences for the current image-question pair and shifts toward a more popular preference within the relevant facet. Incorrect predictions without this population-directed shift are ordinary errors, while verified acceptable alternatives are not counted as errors.

4 Method

PrefMoE addresses the two challenges discussed in the previous section through preference representation learning and query-dependent adaptation. First, it separates shared preference structure from personalized residuals and regularizes these residuals to preserve individual preference differences. Second, it uses hierarchical routing to combine profile- and preference-conditioned LoRA updates according to the current image-question pair, supporting the selective use of the learned user factors during generation.

4.1 Factorized User Representation and Profile Learning

Personalized responses may depend on different aspects of user information. Profile-related questions require identity and attribute cues, whereas preference-related questions require the user's choices within the relevant semantic facets. We therefore organize user information into separate profile and preference factors, allowing these sources to be learned and used selectively. Given the personalization information $\mathcal{X}_i = \{I_i, D_i, \mathcal{P}_i\}$ of user u_i, we construct the factorized state $\{\mathbf{z}_i^{\mathrm{img}}, \mathbf{z}_i^{\mathrm{des}}, \{\mathbf{z}_{i,f}^{\mathrm{pref}}\}_{f=1}^{F}\}$. The profile factors $\mathbf{z}_i^{\mathrm{img}}$ and $\mathbf{z}_i^{\mathrm{des}}$ capture complementary information from the user's images and textual description. The preference factors $\mathbf{z}_{i,f}^{\mathrm{pref}}$ represent the user's preferences across F semantic facets, such as lifestyle and fashion. This factorization makes the different sources of user information separately accessible to the subsequent learning objectives and hierarchical router.

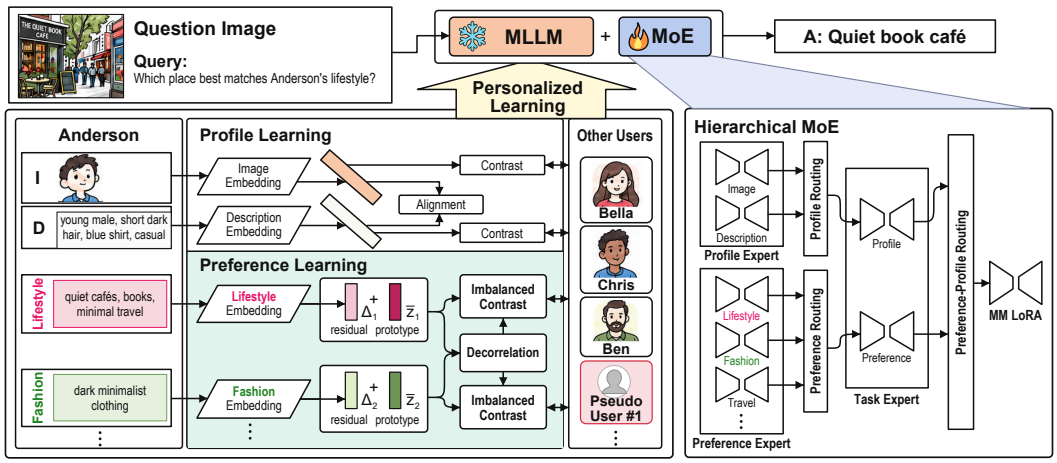

Figure 2: *Overview of PrefMoE*. PrefMoE separates profile and preference representations, models preferences with shared prototypes and personalized residuals, and regularizes the residuals with contrastive learning and decorrelation. A hierarchical MoE router then activates profile- and preference-aware LoRA experts for query-dependent personalized reasoning.

We first learn the profile factors from the visual and textual inputs:

$$\mathbf{z}_i^{\text{img}} = g_{\text{img}}(I_i), \quad \mathbf{z}_i^{\text{des}} = g_{\text{des}}(D_i), \tag{2}$$

where g_{img} and g_{des} are the visual and textual profile encoders. Because the two inputs describe the same user, we encourage cross-modal alignment while preserving distinctions between users. The profile learning objective is

$$\mathcal{L}_{\text{prof}} = \mathcal{L}_{\text{align}}^{\text{prof}} + \mathcal{L}_{\text{contra}}^{\text{prof}}. \tag{3}$$

Here, $\mathcal{L}_{\text{align}}^{\text{prof}}$ aligns the visual and textual profile factors of the same user, while $\mathcal{L}_{\text{contra}}^{\text{prof}}$ encourages discrimination between users. These profile factors provide identity-related information for personalization, while the preference factors are learned through the following residual-based formulation.

4.2 Preserving Individual Preference Residuals

Under joint multi-user learning, frequent preference patterns may overshadow less common user-specific signals. To preserve individual preference differences, we decompose each facet-specific preference factor into a shared prototype and a personalized residual:

$$\mathbf{z}_{i,f}^{\text{pref}} = \bar{\mathbf{z}}_f + \Delta_{i,f}, \qquad f = 1, \ldots, F, \tag{4}$$

where $\bar{\mathbf{z}}_f$ is shared across users within facet f, and $\Delta_{i,f}$ represents the preference offset of user u_i from this shared reference. We obtain each personalized residual $\Delta_{i,f}$ through a learnable embedding layer and optimize it during training. For each facet f, the shared prototype $\bar{\mathbf{z}}_f$ is computed by averaging the corresponding user representations across training users. This decomposition provides separate components for shared preference structure and individual deviations. However, the decomposition alone does not ensure that the residuals retain informative preference differences. We therefore combine imbalance-aware residual contrast, pseudo-user augmentation, and residual decorrelation to regularize preference learning.

Imbalance-aware Residual Contrast. Low-frequency preferences may be underrepresented during training. To preserve their influence on query-specific responses, we apply an imbalance-aware contrastive objective to residual-conditioned activations rather than directly to the residual embeddings. For each image-question pair (V_n, Q_n), let $\mathbf{h}_n = \text{Backbone}(V_n, Q_n)$ denote the backbone hidden representation. For user u_i and facet f, we compute $\mathbf{a}_{i,f}^n = E_f(\mathbf{h}_n; \Delta_{i,f})$, where E_f produces an activation conditioned on the current visual-question context and the personalized residual.

We construct contrastive supervision using facet-level preference annotations. Let $\mathcal{B}$ denote the candidate users in the mini-batch, and let $\mathcal{N}_{i,f} \subseteq \mathcal{B}$ contain users whose annotations match those of u_i on facet f. For the same image-question pair, we compute the normalized similarity mass assigned

to matching preferences:

$$p_{i,f}^n = \frac{\sum_{u_j \in \mathcal{N}_{i,f}} \exp\left(\cos(\mathbf{a}_{i,f}^n, \mathbf{a}_{j,f}^n)/\tau\right)}{\sum_{u_j \in \mathcal{B}} \exp\left(\cos(\mathbf{a}_{i,f}^n, \mathbf{a}_{j,f}^n)/\tau\right)}. \tag{5}$$

$\cos(\cdot,\cdot)$ denotes cosine similarity and τ is the temperature. Maximizing this quantity encourages similar activations for matching facet-level preferences relative to non-matching preferences. Inspired by focal loss (Lin et al., 2017) and class-imbalanced reweighting (Cui et al., 2019), we define:

$$\mathcal{L}_{\text{res}} = -\frac{1}{BF} \sum\nolimits_{i=1}^{B} \sum\nolimits_{f=1}^{F} (1-\rho_{i,f})^{\eta} (1-p_{i,f}^n) \log p_{i,f}^n, \tag{6}$$

where $B = |\mathcal{B}|$, $\rho_{i,f} \in [0,1]$ is the normalized frequency of the preference group associated with u_i under facet f, and $\eta \geq 0$ controls the frequency weighting. The factor $(1-\rho_{i,f})^{\eta}$ gives relatively greater weight to less frequent preference groups, while $(1-p_{i,f}^n)$ emphasizes cases where matching preferences receive low similarity mass.

Counterfactual Pseudo-user Augmentation. Observed users provide a limited set of preference combinations within each facet. To expand these combinations, we construct pseudo users by recombining preference items from different real users. Specifically, for each of the fixed F facets, we randomly sample multiple preference items from other users and combine them into the pseudo user's preferences for that facet. The resulting preference profile $\tilde{\mathcal{P}}$ contains new combinations of existing items without introducing new facets or preference values. We include these pseudo users as additional candidates in the residual contrastive objective, with preference matches determined from their recombined facet contents. Pseudo users participate only in residual contrastive learning and require no additional answer-level labels.

Preference Residual Decorrelation. The residual contrastive objective encourages preference-specific activations but does not explicitly discourage overlapping residual directions across facets. To reduce this redundancy, we introduce soft decorrelation penalties between different facet residuals and between each residual and its corresponding shared prototype. Let $\tilde{\Delta}_{i,f} = \Delta_{i,f}/\|\Delta_{i,f}\|_2$ and $\tilde{\mathbf{z}}_f = \bar{\mathbf{z}}_f/\|\bar{\mathbf{z}}_f\|_2$ denote the normalized residual and prototype, respectively. We define

$$\mathcal{L}_{\text{dec}}(u_i) = \frac{1}{\binom{F}{2}} \sum\nolimits_{1 \leq f < g \leq F} \left(\tilde{\Delta}_{i,f}^{\top} \tilde{\Delta}_{i,g}\right)^2 + \frac{1}{F} \sum\nolimits_{f=1}^{F} \left(\tilde{\Delta}_{i,f}^{\top} \tilde{\mathbf{z}}_f\right)^2. \tag{7}$$

The first term encourages distinct residual directions across preference facets. The second discourages overlap with the shared prototype direction, encouraging residuals to encode complementary information. Both terms act as soft regularizers rather than enforcing independence between preference factors. We average $\mathcal{L}_{\text{dec}}(u_i)$ over real users in the mini-batch to obtain $\mathcal{L}_{\text{dec}}$. The overall preference learning objective combines query-conditioned residual contrast with residual decorrelation:

$$\mathcal{L}_{\text{pref}} = \mathcal{L}_{\text{res}} + \mathcal{L}_{\text{dec}}. \tag{8}$$

4.3 Query-Dependent Preference Utilization

Preserving individual preferences does not ensure that the model uses the cues relevant to a given image-question pair. We therefore introduce hierarchical routing over LoRA experts to support query-dependent use of the factorized user state. The router first combines experts within the profile and preference branches, then fuses the two branch updates. Given an image-question pair (V, Q), we obtain query features $\mathbf{q} = [\phi_v(V); \phi_q(Q)]$ for routing. For clarity, we describe the adaptation at a single adapter insertion site, where $\mathbf{h}$ denotes the backbone hidden representation. Each expert is implemented as a LoRA and produces an additive low-rank update to $\mathbf{h}$. The target user's profile factors $\mathbf{z}_i^{\text{img}}, \mathbf{z}_i^{\text{des}}$ and preference factors $\{\mathbf{z}_{i,f}^{\text{pref}}\}_{f=1}^{F}$ provide the user conditions for adaptation.

The preference branch contains F facet-specific LoRA experts $\{E_f^{\text{pref}}\}_{f=1}^{F}$, while the profile branch contains two experts $\{E_{\text{img}}^{\text{prof}}, E_{\text{des}}^{\text{prof}}\}$ associated with the visual and textual profile factors. The routers assign mixture weights using the query and the corresponding user factors:

$$\boldsymbol{\alpha}^{\text{pref}} = \sigma\left(\left\{\mathbf{w}_{\text{pref},f}^{\top}[\mathbf{q}; \mathbf{z}_{i,f}^{\text{pref}}]\right\}_{f=1}^{F}\right), \quad \boldsymbol{\alpha}^{\text{prof}} = \sigma\left(\left\{\mathbf{w}_{\text{prof},k}^{\top}[\mathbf{q}; \mathbf{z}_{i,k}^{\text{prof}}]\right\}_{k \in \{\text{img},\text{des}\}}\right), \tag{9}$$

where σ denotes Softmax over the experts within each branch, $\mathbf{z}^{\text{prof}}_{i,\text{img}} = \mathbf{z}^{\text{img}}_i$, and $\mathbf{z}^{\text{prof}}_{i,\text{des}} = \mathbf{z}^{\text{des}}_i$. The scoring vectors $\mathbf{w}_{\text{pref},f}$ and $\mathbf{w}_{\text{prof},k}$ are expert-specific, allowing the query to affect the relative expert weights. The preference router weights the facet-specific experts, while the profile router weights the visual and textual profile experts. The factor-aware MoE updates are computed as

$$\mathbf{h}^{\text{pref}} = \sum\nolimits_{f=1}^{F} \alpha^{\text{pref}}_f E^{\text{pref}}_f(\mathbf{h}; \mathbf{z}^{\text{pref}}_{i,f}), \quad \mathbf{h}^{\text{prof}} = \sum\nolimits_{k\in\{\text{img,des}\}} \alpha^{\text{prof}}_k E^{\text{prof}}_k(\mathbf{h}; \mathbf{z}^{\text{prof}}_{i,k}). \tag{10}$$

Here, each expert returns only the adaptation update, excluding the original backbone representation.

A second router determines the relative contributions of the profile and preference branches using the query and their updates:

$$\boldsymbol{\gamma} = \sigma\left(\mathbf{W}_{\text{br}}[\mathbf{q}; \mathbf{h}^{\text{prof}}; \mathbf{h}^{\text{pref}}]\right). \tag{11}$$

Softmax is applied over the two branches, and the adapted representation is

$$\tilde{\mathbf{h}} = \mathbf{h} + \gamma_{\text{prof}}\mathbf{h}^{\text{prof}} + \gamma_{\text{pref}}\mathbf{h}^{\text{pref}}. \tag{12}$$

This hierarchy supports query-dependent weighting of individual user factors within each branch and adaptive fusion of profile and preference information.

Training objective and inference. The overall training objective combines three loss functions:

$$\mathcal{L} = \mathcal{L}_{\text{vqa}} + \mathcal{L}_{\text{prof}} + \mathcal{L}_{\text{pref}}. \tag{13}$$

At inference time, the model retrieves the learned profile and preference factors associated with the target user identifier. For each image-question pair, the hierarchical routers compute the mixture weights and combine the corresponding LoRA updates to generate a personalized response.

5 Experiment

5.1 Experimental Setup

Dataset and implementation details. We evaluate on MMPB-Clean, a cleaned split of MMPB (Kim et al., 2025) covering 50 human users, 61 non-human concepts, and five preference facets. We retain the original user-concept annotations and labels for recognition and multiple-choice questions, reconstructing preference yes/no labels when needed to reduce question-format shortcuts. All methods share the same split and are evaluated without dialogue history (0-turn) or with a ten-turn history (10-turn). Construction details and statistics are provided in Appendix B. We implement PrefMoE in PyTorch and train it on four NVIDIA A100 GPUs using AdamW (Loshchilov & Hutter, 2019). Training configurations are detailed in Appendix C.

Metrics. We report exact-match accuracy on profile and preference questions, together with overall accuracy. To distinguish population-directed errors from ordinary errors, we additionally measure preference collapse:

$$\text{Collapse} = \frac{1}{|\mathcal{O}|}\sum\nolimits_{i\in\mathcal{O}} \hat{y}_i\, \mathbb{I}[\text{pop}(p_i) > \text{pop}(p^\star_i)]\,. \tag{14}$$

where $\mathcal{O}$ contains binary queries about items outside the target user's annotated acceptable preference set, and $\hat{y}_i = 1$ indicates that the model accepts the queried item p_i as preferred. A false acceptance counts only if p_i is more popular than a fixed acceptable reference $p^\star_i$ for the same user and facet, with popularity measured by annotated user counts. Lower is better. Reference selection, popularity computation, and answer normalization are detailed in Appendix D.

Compared methods. We compare with representative general-purpose MLLMs across different model scales, inclduing LLaVA series, DeepSeek VL2 and Qwen2.5-VL. We also include recent personalized or multimodal baselines, including Yo'LLaVA (Nguyen et al., 2024), LLaVA-NeXT (Liu et al., 2024), LOVA3 (Zhao et al., 2024), and TG-LLaVA (Yan et al., 2025). For general-purpose MLLMs, we report no-training (NT) and full fine-tuning (FFT) results. For personalized multimodal methods and our method, we report PEFT-based personalization results. At inference time, NT and FFT baselines receive explicit user profile and preference descriptions as contextual prompts, while PrefMoE only receives the user identifier and retrieves the learned structured user state from memory.

5.2 Major Comparisons and Ablation Studies

Comparisons on MMPB-Clean. Table 1 reports the main results under 0-turn and 10-turn settings. Raw MLLMs exhibit high preference collapse, showing that general multimodal capability alone is

Table 1: Major comparisons with SOTAs under 0-turn and 10-turn settings.

Method	Type	0-turn				10-turn			
		Overall↑	Preference↑	Profile↑	Collapse↓	Overall↑	Preference↑	Profile↑	Collapse↓
LLaVA-1.5-7B	*NT*	0.3601	0.3387	0.3716	0.6228	0.3243	0.3067	0.3337	0.6164
LLaVA-1.5-13B	*NT*	0.4023	0.3653	0.4222	0.6027	0.3720	0.3333	0.3928	0.6119
LLaVA-OV-72B	*NT*	0.5100	0.4800	0.5262	0.5297	0.4620	0.4333	0.4774	0.5525
DeepSeek-VL2-Tiny	*NT*	0.3683	0.3613	0.3720	0.7169	0.3497	0.3440	0.3527	0.6895
DeepSeek-VL2-Small	*NT*	0.3866	0.3853	0.3871	0.6895	0.3668	0.3667	0.3669	0.6712
DeepSeek-VL2	*NT*	0.4620	0.4533	0.4667	0.5890	0.4103	0.4067	0.4122	0.5982
Qwen2.5-VL-7B	*NT*	0.4401	0.4000	0.4616	0.6484	0.4007	0.3433	0.4315	0.6530
Qwen2.5-VL-32B	*NT*	0.4914	0.4333	0.5226	0.5708	0.4410	0.4027	0.4616	0.5982
Qwen2.5-VL-72B	*NT*	0.5301	0.4600	0.5677	0.5023	0.5016	0.4453	0.5319	0.5525
LLaVA-1.5-7B	*FFT*	0.5072	0.4413	0.5427	0.3425	0.5026	0.4413	0.5355	0.3470
LLaVA-1.5-13B	*FFT*	0.5301	0.4600	0.5677	0.3105	0.5100	0.4400	0.5477	0.3196
LLaVA-OV-72B	*FFT*	0.6200	0.5947	0.6337	0.3151	0.5613	0.5320	0.5771	0.3379
DeepSeek-VL2-Tiny	*FFT*	0.4709	0.4213	0.4975	0.5845	0.4681	0.4187	0.4946	0.5936
DeepSeek-VL2-Small	*FFT*	0.4862	0.3840	0.5412	0.6530	0.4834	0.3747	0.5419	0.6530
DeepSeek-VL2	*FFT*	0.5814	0.5093	0.6201	0.4521	0.5748	0.4987	0.6158	0.4566
Qwen2.5-VL-7B	*FFT*	0.4452	0.3013	0.5226	0.6530	0.4807	0.3707	0.5398	0.7854
Qwen2.5-VL-32B	*FFT*	0.5718	0.5480	0.5849	0.4795	0.4914	0.4333	0.5226	0.5114
Qwen2.5-VL-72B	*FFT*	0.6014	0.5947	0.6057	0.4018	0.5217	0.5053	0.5305	0.3927
Yo'LLaVA	*PEFT*	0.5040	0.4880	0.5125	0.2075	0.4840	0.4680	0.4925	0.2146
LLaVA-NeXT-34B	*PEFT*	0.5599	0.6200	0.5276	0.2466	0.5299	0.5900	0.4976	0.2054
LOVA3	*PEFT*	0.5329	0.5680	0.5140	0.4292	0.4979	0.5330	0.4790	0.4247
TG-LLaVA	*PEFT*	0.5413	0.5800	0.5204	0.2329	0.5013	0.5400	0.4804	0.1506
PrefMoE (LLaVA-1.5-7B)	*PEFT*	0.6751	0.6733	0.6760	0.1233	0.5986	0.5840	0.6065	0.1553
PrefMoE (LLaVA-1.5-13B)	*PEFT*	0.7012	0.6800	0.7125	0.1416	0.6601	0.6267	0.6781	0.1370
PrefMoE (LLaVA-OV-72B)	*PEFT*	0.7893	0.7613	0.8043	0.1142	0.6914	0.6627	0.7068	**0.1187**
PrefMoE (DeepSeek-VL2-Tiny)	*PEFT*	0.6601	0.6467	0.6674	0.2511	0.6033	0.5933	0.6086	0.2740
PrefMoE (DeepSeek-VL2-Small)	*PEFT*	0.7305	0.5813	0.8108	0.2740	0.6382	0.5200	0.7018	0.2694
PrefMoE (DeepSeek-VL2)	*PEFT*	0.7991	0.7520	**0.8244**	0.1324	0.7012	**0.6800**	0.7125	0.1826
PrefMoE (Qwen2.5-VL-7B)	*PEFT*	0.7613	0.6880	0.8007	0.1781	0.6503	0.6253	0.6638	0.1826
PrefMoE (Qwen2.5-VL-32B)	*PEFT*	0.7902	0.7640	0.8043	0.1416	0.6900	0.6653	0.7032	0.2100
PrefMoE (Qwen2.5-VL-72B)	*PEFT*	**0.8117**	**0.7893**	0.8237	**0.1096**	**0.7301**	0.5813	**0.8100**	0.1279

insufficient for user-specific preference modeling. Full fine-tuning improves accuracy but still fails to reliably preserve preference boundaries. By contrast, our method consistently improves overall and preference accuracy while reducing collapse across backbones and model scales. On LLaVA-1.5-7B, it improves 0-turn overall accuracy from 0.5072 to 0.6751 and reduces collapse from 0.3425 to 0.1233 over FFT. The gains further extend to stronger backbones, with different PrefMoE variants achieving the best 0-turn overall accuracy, the best 10-turn preference accuracy, and the lowest collapse rates. These results suggest that preference-sensitive personalization benefits from factorized adaptation rather than relying solely on scaling or standard fine-tuning.

Ablation studies. Table 2 shows that each component contributes to our method. The basic user embedding still suffers from severe collapse, indicating that storing user information alone is insufficient. Profile factor learning brings the largest early improvement, while imbalance-aware residual preservation further improves preference accuracy by protecting low-density preference offsets. Counterfactual augmentation and preference decorrelation mainly reduce collapse, suggesting that they help break group-level shortcuts and separate preference facets. The hierarchical MoE router achieves the best results, confirming the need to couple factorized user states with factorized computation paths.

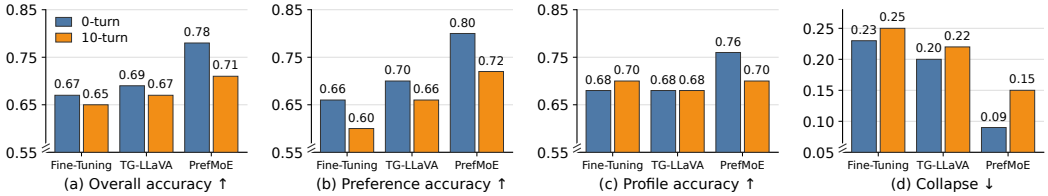

Figure 3: Cross-domain results on food and fashion using LLaVA-1.5-7B.

Validation on additional domains. Beyond MMPB-Clean, we further construct a validation set using images from Food-101 (Bossard et al., 2014) and Fashion Product Images (Small) (Aggarwal, 2019). It contains 20 users, 10 food and fashion preference categories, 500 training questions, and 100 test questions, equally divided between profile and preference questions. The personalized questions and labels are constructed for this evaluation and independently verified by human annotators for consistency with the images and user preferences. Fig. 3 shows that PrefMoE achieves the highest preference accuracy and lowest collapse among the compared methods in both dialogue settings. These results provide complementary evidence beyond MMPB-Clean, although the small, constructed set limits broader real-world conclusions.

Table 2: Component-wise ablation study of the proposed method. E, P, I, C, D, and M denote the basic user embedding module, profile factor learning, imbalance-aware residual preservation, counterfactual user augmentation, preference decorrelation, and hierarchical MoE router, respectively.

						0-turn				10-turn			
E	**P**	**I**	**C**	**D**	**M**	Overall↑	Preference↑	Profile↑	Collapse↓	Overall↑	Preference↑	Profile↑	Collapse↓
✓						0.5562	0.4120	0.6337	0.6027	0.4928	0.3584	0.5651	0.6347
✓	✓					0.6000	0.5700	0.6159	0.3333	0.5266	0.4902	0.5461	0.3653
✓	✓	✓				0.6279	0.6133	0.6358	0.3288	0.5506	0.5275	0.5631	0.3607
✓	✓	✓	✓			0.6303	0.6253	0.6330	0.2283	0.5571	0.5393	0.5667	0.2603
✓	✓	✓	✓	✓		0.6382	0.6467	0.6337	0.1457	0.5688	0.5622	0.5723	0.1781
✓	✓	✓	✓	✓	✓	**0.6751**	**0.6733**	**0.6760**	**0.1233**	**0.5986**	**0.5840**	**0.6065**	**0.1553**

5.3 More Analysis

Number of pseudo users. Fig. 4 shows that performance improves as the number of pseudo users increases from 10 to 50, while both Preference and Collapse become saturated afterward. This suggests that 50 pseudo users are sufficient to capture representative personalized patterns in our setting, and we therefore use this value by default.

w/o vs. w/ facet labels. MMPB annotations organize preferences into five facets: entertainment, travel, lifestyle, shopping, and fashion. To test whether our method relies on explicit facet-name shortcuts, we remove these facet labels and merge all preference descriptions into a single unstructured text paragraph. In this setting, facet structure is not directly provided and must be inferred from the preference content, mainly through the decorrelation-based factor learning. As shown in Fig. 5, our method remains robust without facet labels and still outperforms FFT baselines across different backbones. This suggests that the gains do not mainly come from surface-level facet names, but from learning structured target-specific preference relations from the underlying descriptions.

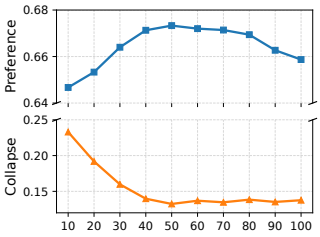

Figure 4: No. of pseudo users.

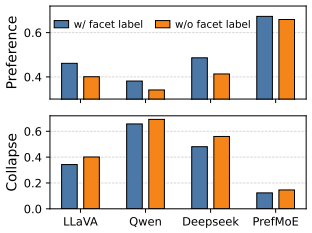

Figure 5: W/o facet labels.

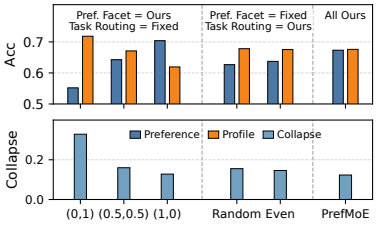

Figure 6: Routing analysis.

Adaptive routing vs. fixed routing. Fig. 6 analyzes the routing behavior. We first fix the preference–profile branch weights while keeping our preference facet router unchanged. The results show a clear trade-off: profile-only routing achieves strong profile accuracy but weak preference accuracy and severe collapse, whereas preference-only routing improves preference accuracy and reduces collapse but sacrifices profile accuracy and overall performance. Although balanced fixed routing mitigates this trade-off, it still underperforms our adaptive branch router. We also replace the adaptive preference facet router with random or uniform facet selection while keeping the adaptive branch router. Both variants degrade performance, showing that query-dependent facet selection is necessary. Overall, PrefMoE achieves the best overall accuracy and the lowest collapse in the 0-turn setting, suggesting that effective personalized reasoning requires both adaptive profile–preference fusion and adaptive preference facet routing.

t-SNE visualizations. We use t-SNE to qualitatively examine whether the learned preference representations preserve structured preference boundaries at both facet and population levels. Fig. 7 visualizes representations from the five semantic facets, including entertainment, travel, lifestyle, shopping, and fashion. Compared with standard fine-tuning, our method produces more separated facet-level distributions, suggesting that preference decorrelation helps different facets capture complementary user-specific offsets rather than collapsing into redundant directions. Fig. 8 further examines the top-1 to top-4 preference populations within each facet. Standard fine-tuning tends to mix these populations into shared regions, indicating that frequent and semantically related preferences are not well distinguished. In contrast, our method preserves clearer local population structures within the same broad facet. These results suggest that our method mitigates preference collapse not only by separating different preference facets, but also by maintaining fine-grained boundaries among preference populations inside each facet.

Qualitative Case Study. Fig. 9 presents representative profile and preference cases under 0-turn and

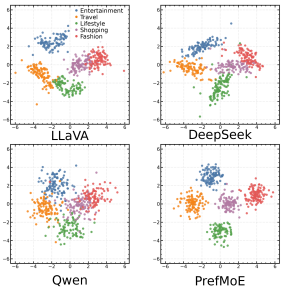

Figure 7: t-SNE of 5 different preference facets.

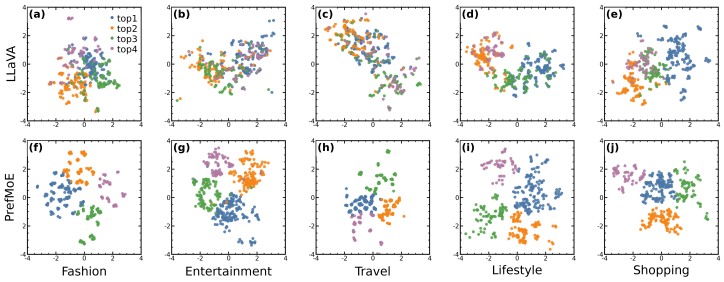

Figure 8: t-SNE visualization of top-1 to top-4 preference groups within five preference facets.

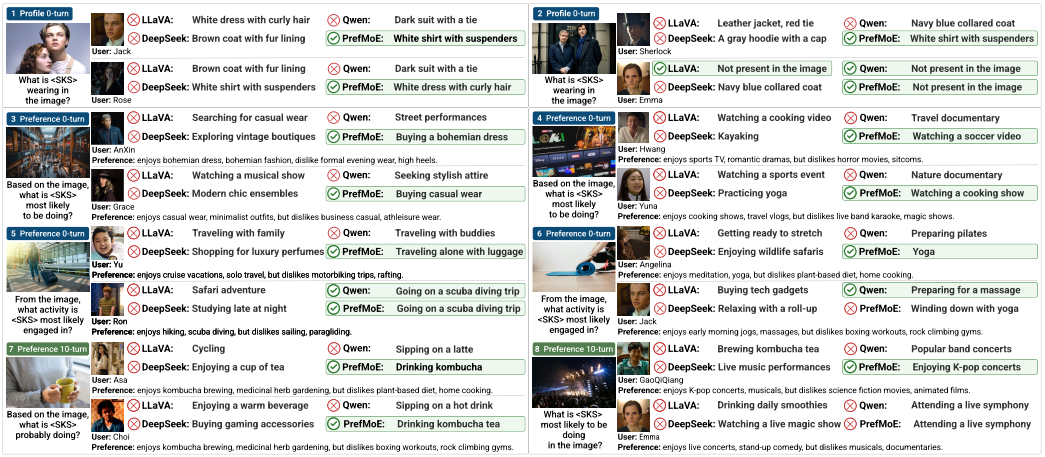

Figure 9: Qualitative examples. <SKS> denotes the target personalized identity or concept. Green boxes indicate correct predictions, while red marks indicate incorrect predictions.

10-turn settings. For profile questions, baseline MLLMs often rely on salient visual cues and produce target-agnostic descriptions, while PrefMoE better conditions on the queried user. For preference questions, PrefMoE more often predicts answers consistent with individual preferences, such as bohemian fashion, solo travel, kombucha, and K-pop concerts. The failure cases also reveal remaining challenges. In Case 6, PrefMoE identifies a relevant preference but fails to precisely ground the visual activity, while Case 8 shows confusion among fine-grained entertainment categories. These results highlight both the effectiveness and limitations of preference-grounded personalized reasoning.

6 Conclusion

In this paper, we identified group preference collapse as a critical failure mode in multi-user personalized MLLMs, where responses drift toward dominant population-level preferences rather than reflecting individualized user preferences. We analyzed this problem from both representation and reasoning perspectives, showing that sparse preference signals can be suppressed in user representations and unreliably used during generation. To address this issue, we proposed PrefMoE, a preference-centric personalization framework that separates stable profile information from preference representations and explicitly preserves individualized preference residuals. A hierarchical routing mechanism further directs profile and preference factors into separate LoRA adaptation paths for query-dependent personalized reasoning. Experiments across multiple MLLM backbones, dialogue settings, and additional visual domains show that PrefMoE consistently improves preference-sensitive personalization and reduces preference collapse.

Despite these improvements, PrefMoE still relies on explicit and well-specified preference descriptions. Personalized reasoning may remain difficult when preferences are underspecified, weakly grounded in the visual scene, or not clearly relevant to the query. The additional validation set is also small and controlled, with a bounded preference vocabulary. Although PrefMoE maintains higher preference accuracy under the evaluated sequential-update setting, our current results do not establish improved resistance to forgetting, and longer-term continual adaptation remains an open challenge.

AI USE STATEMENT

Generative AI tools were used to help reduce grammatical errors and to assist with LaTeX formatting, reference formatting, migration to the ICLR template, and organization and editing of author-provided supplementary results. The authors are responsible for the final content of this work.

ETHICS STATEMENT

This work uses existing multimodal datasets and constructed personalization annotations, with human verification of evaluation samples. Personalized models may retain sensitive user information and reinforce population-level biases. Appendix E reports bias diagnostics and discusses privacy risks and evaluation limitations. Applications involving real users should protect personal information, obtain appropriate consent, and assess fairness beyond the evaluated settings.

REPRODUCIBILITY STATEMENT

We provide the source code in the supplementary materials. Implementation details, model configurations, and training hyperparameters are documented in Appendix C. Dataset construction and evaluation metrics are described in Appendices B and D, respectively.

REFERENCES

Param Aggarwal. Fashion Product Images (Small). Kaggle, 2019. URL https://www.kaggle.com/datasets/paramaggarwal/fashion-product-images-small.

Yuval Alaluf, Elad Richardson, Sergey Tulyakov, Kfir Aberman, and Daniel Cohen-Or. MyVLM: Personalizing VLMs for user-specific queries. In *ECCV*, pp. 73–91, 2024.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: A visual language model for few-shot learning. In *NeurIPS*, volume 35, pp. 23716–23736, 2022. doi: 10.52202/068431-1723.

Ruichuan An, Sihan Yang, Renrui Zhang, Ming Lu, Tianyi Jiang, Kai Zeng, Yulin Luo, Jiajun Cao, Hao Liang, Ying Chen, Qi She, Shanghang Zhang, and Wentao Zhang. MC-LLaVA: Multi-concept personalized vision-language model. *arXiv preprint arXiv:2411.11706*, 2024.

Ruichuan An, Sihan Yang, Renrui Zhang, Zijun Shen, Ming Lu, Gaole Dai, Hao Liang, Ziyu Guo, Shilin Yan, Yulin Luo, Bocheng Zou, Chaoqun Yang, and Wentao Zhang. UniCTokens: Boosting personalized understanding and generation via unified concept tokens. In *NeurIPS*, volume 38, pp. 144638–144664, 2025. doi: 10.52202/085713-4840.

Ruichuan An, Kai Zeng, Ming Lu, Sihan Yang, Renrui Zhang, Huitong Ji, Hao Liang, and Wentao Zhang. Concept-as-Tree: A controllable synthetic data framework makes stronger personalized VLMs. In *ECCV*, 2026. URL https://eccv.ecva.net/virtual/2026/poster/4499.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *ICCV*, pp. 2425–2433, December 2015.

Huiyu Bai, Runze Wang, Zhuoyun Du, Yiyang Zhao, Fengji Zhang, Haoyu Chen, Xiaoyong Zhu, Bo Zheng, and Xuejiao Zhao. Online-PVLM: Advancing personalized VLMs with online concept learning. *arXiv preprint arXiv:2511.20056*, 2025.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez,

Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *ECCV*, pp. 446–461, 2014. doi: 10.1007/978-3-319-10599-4_29. URL https://data.vision.ee.ethz.ch/cvl/datasets_extra/food-101/.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3–4):324–345, 1952. doi: 10.1093/biomet/39.3-4.324.

Ta-Chung Chi, Po-Chun Chen, Shang-Yu Su, and Yun-Nung Chen. Speaker role contextual modeling for language understanding and dialogue policy learning. In *IJCNLP*, pp. 163–168, 2017.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *NeurIPS*, volume 30, 2017.

Niv Cohen, Rinon Gal, Eli A. Meirom, Gal Chechik, and Yuval Atzmon. "this is my unicorn, Fluffy": Personalizing frozen vision-language representations. In *ECCV*, pp. 558–577, 2022. doi: 10.1007/978-3-031-20044-1_32.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, pp. 9268–9277, 2019.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven C. H. Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *NeurIPS*, volume 36, pp. 49250–49267, 2023. doi: 10.52202/075280-2142.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *ICML*, volume 235, pp. 12634–12651, 2024.

Wenlong Fang, Qiaofeng Wu, Jing Chen, and Yun Xue. Notes-guided MLLM reasoning: Enhancing MLLM with knowledge and visual notes for visual question answering. In *CVPR*, pp. 19597–19607, 2025.

Junyu Feng, Binxiao Xu, Jiayi Chen, Mengyu Dai, Cenyang Wu, Haodong Li, Bohan Zeng, Yunliu Xie, Hao Liang, Ming Lu, and Wentao Zhang. M2A: Multimodal memory agent with dual-layer hybrid memory for long-term personalized interactions. *arXiv preprint arXiv:2602.07624*, 2026.

Luise Ge, Daniel Halpern, Evi Micha, Ariel D. Procaccia, Itai Shapira, Yevgeniy Vorobeychik, and Junlin Wu. Axioms for AI alignment from human feedback. In *NeurIPS*, volume 37, pp. 80439–80465, 2024. doi: 10.52202/079017-2557.

Joey Hejna, Rafael Rafailov, Harshit Sikchi, Chelsea Finn, Scott Niekum, W. Bradley Knox, and Dorsa Sadigh. Contrastive preference learning: Learning from human feedback without reinforcement learning. In *ICLR*, pp. 18770–18798, 2024.

Qihan Huang, Weilong Dai, Jinlong Liu, Wanggui He, Hao Jiang, Mingli Song, Jingyuan Chen, Chang Yao, and Jie Song. Boosting MLLM reasoning with text-debiased Hint-GRPO. In *ICCV*, pp. 4848–4857, 2025.

Jaeik Kim, Woojin Kim, Woohyeon Park, and Jaeyoung Do. MMPB: It's time for multi-modal personalization. In *NeurIPS Datasets and Benchmarks Track*, volume 38, pp. 29306–29358, 2025. doi: 10.52202/085713-0872.

Jiwei Li, Michel Galley, Chris Brockett, Georgios P. Spithourakis, Jianfeng Gao, and Bill Dolan. A persona-based neural conversation model. In *ACL*, pp. 994–1003, 2016. doi: 10.18653/v1/P16-1094.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, volume 202, pp. 19730–19742, 2023.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pp. 2980–2988, 2017.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. Technical blog, 2024. URL https://llava-vl.github.io/blog/2024-01-30-llava-next/.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Thao Nguyen, Haotian Liu, Yuheng Li, Mu Cai, Utkarsh Ojha, and Yong Jae Lee. Yo'LLaVA: Your personalized language and vision assistant. In *NeurIPS*, volume 37, pp. 40913–40951, 2024. doi: 10.52202/079017-1294.

Thao Nguyen, Krishna Kumar Singh, Jing Shi, Trung Bui, Yong Jae Lee, and Yuheng Li. Yo'Chameleon: Personalized vision and language generation. In *CVPR*, pp. 14438–14448, 2025.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *NeurIPS*, volume 35, pp. 27730–27744, 2022. doi: 10.52202/068431-2011.

Chau Pham, Hoang Phan, David Doermann, and Yunjie Tian. PLVM: A tuning-free approach for personalized large vision-language model. In *CVPR Workshops*, pp. 3671–3680, 2025.

Renjie Pi, Jianshu Zhang, Tianyang Han, Jipeng Zhang, Rui Pan, and Tong Zhang. Personalized visual instruction tuning. In *ICLR*, pp. 43247–43266, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, volume 36, pp. 53728–53741, 2023. doi: 10.52202/075280-2338.

Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*, pp. 452–461, 2009.

Soroush Seifi, Vaggelis Dorovatas, Matteo Cassinelli, Fabien Despinoy, Daniel Olmeda Reino, and Rahaf Aljundi. Personalization toolkit: Training free personalization of large vision language models. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=5mbn3B0O29.

Wei Tang, Yanpeng Sun, Qinying Gu, and Zechao Li. Visual position prompt for MLLM based visual grounding. *IEEE Transactions on Multimedia*, 28:3739–3754, 2026. doi: 10.1109/tmm.2026.3654372.

Zining Wang, Tongkun Guan, Pei Fu, Chen Duan, Qianyi Jiang, Zhentao Guo, Shan Guo, Junfeng Luo, Wei Shen, and Xiaokang Yang. Marten: Visual question answering with mask generation for multi-modal document understanding. In *CVPR*, pp. 14460–14471, 2025.

Binxiao Xu, Junyu Feng, Shaolin Lu, Yulin Luo, Shilin Yan, Hao Liang, Ming Lu, and Wentao Zhang. Jarvis: Towards personalized AI assistant via personal KV-cache retrieval. *arXiv preprint arXiv:2510.22765*, 2025.

Dawei Yan, Pengcheng Li, Yang Li, Hao Chen, Qingguo Chen, Weihua Luo, Wei Dong, Qingsen Yan, Haokui Zhang, and Chunhua Shen. TG-LLaVA: Text guided LLaVA via learnable latent embeddings. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(9):9076–9084, 2025.

Sihan Yang, Huitong Ji, Shaolin Lu, Jiayi Chen, Binxiao Xu, Ming Lu, Yuanxing Zhang, Wenhui Dong, and Wentao Zhang. Small-large collaboration: Training-efficient concept personalization for large VLM using a meta personalized small VLM. *arXiv preprint arXiv:2508.07260*, 2025.

Chun-Hsiao Yeh, Bryan Russell, Josef Sivic, Fabian Caba Heilbron, and Simon Jenni. Meta-personalizing vision-language models to find named instances in video. In *CVPR*, pp. 19123–19132, 2023.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. Personalizing dialogue agents: I have a dog, do you have pets too? In *ACL*, pp. 2204–2213, 2018. doi: 10.18653/v1/P18-1205.

Henry Hengyuan Zhao, Pan Zhou, Difei Gao, Zechen Bai, and Mike Zheng Shou. LOVA3: Learning to visual question answering, asking and assessment. In *NeurIPS*, volume 37, pp. 115146–115175, 2024. doi: 10.52202/079017-3656.

Yuanhong Zheng, Ruichuan An, Xiaopeng Lin, Yuxing Liu, Sihan Yang, Huanyu Zhang, Haodong Li, Qintong Zhang, Renrui Zhang, Guopeng Li, Yifan Zhang, Yuheng Li, and Wentao Zhang. PEARL: Personalized streaming video understanding model. *arXiv preprint arXiv:2603.20422*, 2026.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *ICLR*, pp. 18378–18394, 2024.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

Appendix

A Extended Related Work

MLLM Personalization. Early personalization studies in language models mainly adapted generated text to personas, speaker roles, or dialogue histories, focusing on stylistic and contextual consistency in text-only interactions (Li et al., 2016; Zhang et al., 2018; Chi et al., 2017). With the rise of MLLMs, personalization has been extended to multimodal inputs, where most methods focus on associating user-specific or instance-specific concepts with visual entities. For example, existing approaches bind personalized concepts to visual representations through learnable embeddings, special concept tokens, or soft prompts (Nguyen et al., 2024; An et al., 2024). Recent methods further improve scalability and flexibility through multimodal in-context conditioning, reference-based identification, and online concept learning (Pi et al., 2025; Pham et al., 2025; Bai et al., 2025). Recent work broadens personalization in several complementary directions. SLC combines a personalized small VLM with a larger model (Yang et al., 2025). UniCTokens unifies concept representations for understanding and generation (An et al., 2025). Concept-as-Tree constructs synthetic concept-learning data (An et al., 2026). Retrieval and memory approaches include Jarvis, which retrieves personal visual and textual KV-caches (Xu et al., 2025), and M2A, which maintains and updates multimodal memories, including user preferences, over long-term interactions (Feng et al., 2026). PEARL studies personalized streaming video understanding (Zheng et al., 2026). These directions address concept acquisition, memory access, and temporal interaction. Our focus is preference diversity under joint multi-user learning: preserving low-frequency user-specific preferences and preventing predictions from drifting toward more popular choices.

Preference Learning. Preference learning aims to infer preferred outputs from pairwise comparisons (Bradley & Terry, 1952), rankings, or choice-based feedback and has been widely used for recommendations (Rendle et al., 2009) and decision making (Christiano et al., 2017). Beyond classical ranking settings, preference feedback has also been used to train agents (Christiano et al., 2017) and generative models from human judgments (Ziegler et al., 2019). It has become a central tool for aligning large language models, from RLHF (Ouyang et al., 2022; Bai et al., 2022) to direct or RL-free optimization methods such as DPO (Rafailov et al., 2023), contrastive preference learning (Hejna et al., 2024), and KTO (Ethayarajh et al., 2024). Most existing studies focus on exploiting preference signals to optimize model outputs toward shared or aggregated human preferences (Ouyang et al., 2022; Bai et al., 2022; Rafailov et al., 2023; Ge et al., 2024). However, this objective differs from personalized preference modeling, where multiple users with heterogeneous preferences must be modeled jointly. In such settings, aggregating preference signals can suppress minority or user-specific preferences and make different users receive increasingly similar responses. Our work studies this failure mode in personalized MLLMs and explicitly models preference diversity to reduce group preference collapse.

B Dataset Construction

B.1 Why MMPB-Clean is Needed

We adopt MMPB (Kim et al., 2025) as the base benchmark because it provides a relatively systematic evaluation setting for multimodal personalization. MMPB contains 10,017 image-query pairs and 111 personalizable concepts spanning humans, animals, objects, and characters. Its human category further includes preference-grounded questions, making it suitable for studying both profile-centric recognition and preference-sensitive reasoning. However, the original benchmark is not specifically designed to evaluate user-conditioned personalization under shortcut-controlled settings.

To make the evaluation better reflect whether a model retrieves and applies the personalized information associated with the queried target, we derive a cleaned evaluation protocol from MMPB, termed **MMPB-Clean**. MMPB-Clean does not introduce external images, users, concepts, or preference annotations. Instead, it reuses the original MMPB resources and reorganizes them into a shortcut-controlled protocol. Specifically, we preserve the original personalized concepts, user profiles, preference annotations, images, and natural-language questions whenever possible, while adjusting the train/evaluation split and reconstructing preference yes/no labels only when necessary. The resulting number of QA pairs may differ from the original MMPB because additional preference negative pairs are constructed from existing MMPB users and preference annotations, rather than from newly collected data.

MMPB-Clean reduces two potential shortcut sources. For recognition questions, we reduce image-level shortcuts by grouping samples derived from the same underlying image and constructing the evaluation set mainly from name-sensitive groups, where changing the queried user or concept name can change the answer. Nearly uniform groups, where most queried targets lead to the same answer, are primarily kept for training support and are prevented from dominating evaluation. For preference yes/no questions, we reduce question-format shortcuts by reformulating the task as image-semantics to user-preference matching. Instead of relying on the surface polarity of the question, such as explicit negation, the reconstructed labels indicate whether the semantic meaning of the image group matches the queried user's corresponding preference field. All methods are trained and evaluated using the same reconstructed split, labels, answer normalization, and evaluation protocol. Therefore, MMPB-Clean provides a fair and reproducible protocol for evaluating user-conditioned personalized memory, with greater emphasis on name-sensitive recognition and preference-boundary reasoning.

B.2 Construction Protocol

MMPB-Clean is constructed from the original MMPB annotations. The reconstruction consists of three parts, corresponding to profile-centric recognition questions, preference yes/no questions, and multiple-choice questions.

For profile-centric recognition questions, we preserve the original task semantics and answer labels. We first group samples by their underlying image. For each image group, we examine whether changing the queried user or concept name can change the answer. Image groups satisfying this condition are treated as name-sensitive groups. The recognition evaluation set is mainly constructed from these name-sensitive groups, while nearly uniform groups are primarily kept for training support. We remove internally inconsistent image groups. Across the dataset, the training and evaluation splits share 387 images. These include multi-person images used in recognition questions with different queried identities. For example, a training question may ask whether person A is present, while an evaluation question asks whether person B is present in the same image. Thus, the recognition split permits image reuse across queries about different identities and is not strictly image-disjoint.

For preference yes/no questions, we reconstruct the evaluation labels as image-semantics to user-preference matching. Each user profile contains preference fields over several facets, including entertainment, fashion, lifestyle, shopping, and travel. For each originally positive image group, we identify the stable preference elements shared by the associated users and treat them as the semantic meaning of the image group. A query is labeled positive if this semantic meaning appears in the queried user's corresponding preference field, and negative otherwise. We keep the original positive pairs and construct additional negative pairs by selecting users whose corresponding preference fields do not contain the image-related elements. Originally negative groups are kept unchanged.

For multiple-choice questions, we keep the original questions and labels unchanged. We only adjust the split by user or concept identity, so that each target contributes a limited number of multiple-choice questions to evaluation while the remaining samples are used for training. This preserves the original multiple-choice task format while reducing repeated target-specific patterns across training and evaluation.

B.3 Human Validation of Reconstructed Labels

The preference yes/no reconstruction uses human-designed deterministic rules rather than an LLM or a learned classifier. Within an image group, the retained preference elements are the intersection of the associated users' preference annotations. Unrelated user–image pairs satisfy a zero-overlap constraint in the relevant preference field. The relatedness decision and question polarity determine the yes/no label, and question polarities are checked programmatically. We manually validate 200 reconstructed samples covering 50 users, collecting 2,000 human judgments. The reconstructed labels agree with these judgments in 99.40% of cases, with an average confidence of 4.97 on a five-point scale (Table 3). This audit supports the reliability of the sampled labels, but it does not establish that every reconstructed label is correct.

Table 3: Human audit of reconstructed MMPB-Clean labels. Judgments denotes individual annotation decisions, not the number of annotators.

Users	Samples	Judgments	Agreement ↑	Confidence (1–5)
50	200	2,000	99.40%	4.97

B.4 Fairness and Reproducibility

MMPB-Clean is used only to define the data split and evaluation labels. All methods are trained and evaluated on exactly the same training and evaluation sets. We do not apply method-specific filtering, relabeling, or sample selection. The reconstructed preference labels are independent of our model design and do not rely on factorized user representations, residual preference learning, or MoE routing. Thus, MMPB-Clean provides a shared evaluation protocol rather than an architecture-specific benchmark. To ensure fair comparison, all methods use the same visual inputs, natural-language questions, target user or concept identifiers, candidate answer space, answer normalization procedure, and evaluation metrics. For questions inherited from the original benchmark, we keep the original wording and labels. For reconstructed preference yes/no questions, the image-semantics to user-preference matching rule is applied before model training and is shared by all methods. To support reproducibility, we will release the reconstructed split files, image-group identifiers, preference-element mappings, answer-label files, and evaluation scripts. These files are sufficient to reproduce MMPB-Clean from the original MMPB annotations and to evaluate future methods under the same protocol.

B.5 Dataset Statistics

Table 4: Detailed statistics of MMPB-Clean. The same evaluation split is used for both 0-turn and 10-turn evaluation settings.

Statistic	Train	Evaluation	Total
Dataset scale			
QA pairs	10371	2145	12516
Images	8272	2131	10016
Human users	50	50	50
Non-human concepts	61	61	61
Preference QA by facet			
All preference QA	6750	750	7500
Entertainment	1341	159	1500
Fashion	1349	151	1500
Lifestyle	1357	143	1500
Shopping	1356	144	1500
Travel	1347	153	1500
Recognition QA by concept type			
All recognition QA	3621	1395	5016
Animal	600	100	700
Character	318	482	800
Human identity	2250	250	2500
Object	453	563	1016

Table 4 summarizes the statistics of MMPB-Clean. The dataset contains 12,516 QA pairs in total, with 10,371 examples for training and 2,145 examples for evaluation. The evaluation split preserves all 50 human preference users and all 61 non-human recognition concepts, ensuring that both personalized preference reasoning and visual concept recognition are assessed under the same user/concept coverage as training.

Compared with the original MMPB, MMPB-Clean increases the number of QA pairs from 10,017 to 12,516. The increase comes from preference QA, which is expanded from 5,001 to 7,500 pairs by constructing additional negative user–image pairs using existing MMPB users and preference annotations. Recognition QA keeps the original 5,016 pairs, with only the train/evaluation split reorganized. Thus, the changed statistics come from protocol reconstruction rather than external data collection.

For preference-oriented QA, MMPB-Clean contains 7,500 examples evenly distributed across five preference facets: entertainment, fashion, lifestyle, shopping, and travel. Each facet contains 1,500 QA pairs in total, while the train/evaluation split remains approximately balanced across facets. For recognition-oriented QA, the dataset contains 5,016 examples covering animal, character, human

identity, and object recognition. In this table, question subtypes are aggregated to provide a compact view of the overall data composition.

C Implementation Details

We instantiate our framework on multiple MLLM backbones, including LLaVA-1.5, LLaVA-OneVision, DeepSeek-VL2, and Qwen2.5-VL. Unless otherwise specified, we use LLaVA-1.5-7B as the default backbone for ablation studies and detailed analysis. For all backbone variants, we keep the same training objective and personalized modules, while adapting the LoRA insertion modules and batch configuration to the corresponding model architecture and memory constraints. We apply rank-64 LoRA to seven projection modules. We insert these adapters into the top 8 decoder layers, with two insertion sites per selected layer: one after the self-attention output and one after the MLP output. This results in 16 personalized residual-adapter insertion sites. For other backbones, we use the same top-layer insertion strategy and adapt the exact number of insertion sites according to the depth of the language decoder. All experiments are implemented in PyTorch. We use AdamW as the optimizer with a learning rate of 2×10^{-4} and a cosine learning-rate scheduler. The default batch size is 4 for LLaVA-1.5-7B. Training is performed with bfloat16 precision and TF32 enabled, and gradient checkpointing is used to reduce memory consumption. Unless otherwise noted, experiments are conducted on NVIDIA A100-40G GPUs.

D Evaluation Metrics

We evaluate model performance using two metrics: answer accuracy and population-directed preference collapse.

Exact-match Answer Accuracy. We evaluate answer correctness with strict exact-match accuracy after normalizing both predictions and ground-truth answers into a canonical label space. For multiple-choice questions, we normalize answers to `a,b,c,d`. For binary questions, we normalize answers to `yes,no`. The evaluator first extracts an explicit option or yes/no answer from the generated response. If no explicit option is found for a multiple-choice question, it further matches the response against the candidate option texts. Predictions that cannot be mapped to a valid canonical label are counted as incorrect. Given an evaluation set $\mathcal{S}$, let g_i and $\hat{g}_i$ denote the normalized ground-truth answer and prediction for sample i, respectively.

We define the per-sample correctness indicator as

$$\mathrm{hit}_i = \mathbb{I}\left[\hat{g}_i = g_i \wedge \hat{g}_i \neq \emptyset\right].$$

The exact-match accuracy on $\mathcal{S}$ is then computed by instance-level micro-averaging:

$$\mathrm{Acc}(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \mathrm{hit}_i.$$

We report accuracy on three evaluation sets:

$$\mathrm{Overall} = \mathrm{Acc}(\mathcal{S}_{\mathrm{all}}), \qquad \mathrm{Preference} = \mathrm{Acc}(\mathcal{S}_{\mathrm{pref}}), \qquad \mathrm{Profile} = \mathrm{Acc}(\mathcal{S}_{\mathrm{prof}}).$$

Here, $\mathcal{S}_{\mathrm{all}}$ is the full evaluation set, $\mathcal{S}_{\mathrm{pref}}$ contains preference-related questions, and $\mathcal{S}_{\mathrm{prof}}$ contains profile-recognition questions that require user-specific profile information. Since Overall is micro-averaged over all samples, it is weighted by the subset sizes and is not necessarily the arithmetic mean of Preference and Profile.

Preference Collapse. Accuracy alone does not reveal whether a model preserves user-specific preference boundaries. A model may answer many samples correctly while still over-expanding a user's preference range by attributing more popular, non-preferred items to that user. We therefore report **Collapse**, which measures erroneous preference attributions outside the target user's acceptable preference set that shift toward more popular preferences.

For each binary preference sample i, let u_i denote the target user and S_i denote the true preference user set of the queried item, i.e., the users for whom the item should be considered preference-matching. Since preference questions may use different surface formulations, we first normalize yes/no answers into a unified preference-semantic space. Let $y_i^* \in \{0, 1\}$ be the normalized ground-truth label and $\hat{y}_i \in \{0, 1\}$ be the normalized model prediction, where 1 means preferred and 0 means not preferred.

Let p_i denote the preference represented by the queried item, and let $p_i^\star$ denote the sample's designated original ground-truth preference within the same facet. We define preference popularity, $\mathrm{pop}(p)$, as the number of distinct users whose annotations include preference p within the relevant facet. When multiple preferences are acceptable, the popularity comparison still uses the designated original reference $p_i^\star$.

We define the boundary-external set as

$$\mathcal{O} = \{\, i \mid u_i \notin S_i,\ y_i^* = 0 \,\}.$$

This set contains samples where the queried item should remain outside the target user's preference boundary. The Collapse score is computed as

$$\begin{aligned}\text{Collapse} &= \frac{1}{|\mathcal{O}|} \sum_{i \in \mathcal{O}} \hat{y}_i \, \mathbb{I}[\mathrm{pop}(p_i) > \mathrm{pop}(p_i^\star)] \\ &= \mathbb{P}(\hat{y} = 1,\ \mathrm{pop}(p) > \mathrm{pop}(p^\star) \mid u \notin S,\ y^* = 0)\,.\end{aligned}$$

Here, $\mathbb{P}$ denotes the empirical probability over the evaluated binary preference samples. An erroneous attribution counts as collapse only when the attributed preference is more popular than the designated reference preference. Errors toward equally or less popular preferences are not counted as collapse, and annotated acceptable alternatives are excluded.

A higher Collapse score means that the model more often attributes more popular, non-preferred items to the target user, while a lower value indicates fewer such population-directed violations of user-specific preference boundaries. Unlike Overall, Preference, and Profile, Collapse is not an accuracy score. Lower values are better.

E Bias Diagnostics and Scope of the Evaluation

E.1 Preference-Frequency Bias and Gender-Stratified Results

Our analysis targets preference-frequency bias under joint multi-user learning: errors that favor common preferences at the expense of a target user's less common choices. Reduced collapse does not by itself establish demographic fairness. Table 5 reports the available accuracy and error-composition diagnostics by preference frequency. For low-frequency preferences, accuracy rises from 28% with full fine-tuning to 53% with PrefMoE, while the proportion of errors classified as collapse falls from 84% to 32%. The error-composition columns describe the share of collapse among incorrect predictions within a group. They are not the Collapse scores reported in the main comparison table.

Table 5: Available preference-frequency diagnostics. All entries are percentages. A dash denotes a value not reported in this diagnostic.

Group	Accuracy ↑		Collapse share among errors ↓	
	Fine-Tuning	PrefMoE	Fine-Tuning	PrefMoE
High	50	73	–	20
Mid	–	65	–	27
Low	28	53	84	32

Using the available gender annotations, we report a diagnostic for 28 male and 22 female users in Table 6. PrefMoE improves accuracy and reduces reported collapse in both groups. The observed accuracy gap is 0.0082 for full fine-tuning and 0.0061 for PrefMoE. With only 50 users, two annotated groups, and no explicit demographic fairness objective, this comparison provides limited descriptive evidence rather than a comprehensive fairness assessment.

E.2 Limitations and Remaining Risks

Dataset and preference scope. MMPB-Clean uses predefined facets and a bounded preference vocabulary. The human-verified cross-domain set, multi-label questions, and explicit weather contexts extend the evaluation in controlled ways, but do not replace a large, naturally collected longitudinal preference dataset. Open-ended preferences, implicit evidence, and context-dependent changes in the same user's choices remain incompletely evaluated.

Table 6: Gender-stratified diagnostic using available user annotations (LLaVA-1.5-7B).

Group	Method	Accuracy ↑	Collapse ↓
Male ($n = 28$)	Fine-Tuning	0.5108	0.3391
	PrefMoE	0.6724	0.1258
Female ($n = 22$)	Fine-Tuning	0.5026	0.3468
	PrefMoE	0.6785	0.1201

Continual adaptation. The sequential experiments reveal degradation despite PrefMoE's stronger final performance. The method does not resolve long-term forgetting, and the evaluated update horizon is insufficient to establish long-term stability. Stale or incorrectly updated user states may continue to produce mismatched answers.

Privacy and fairness. The framework learns and retains user-specific representations, which can encode sensitive information even when the stored state is not a raw personal description. It provides no formal privacy guarantee and does not optimize a comprehensive demographic fairness objective. Remaining risks include demographic shortcuts from portraits or profiles, exposure of personal preferences, and feedback loops that reinforce earlier inferred preferences. The reported reduction in preference-frequency collapse should be interpreted within this narrower scope.

F More Experimental Results

F.1 Collapse Breakdown by Preference Popularity

Table 7 reports the original popularity-stratified results across LLaVA-family backbones. The groups are defined by how many users share the queried preference: at most four users, five to eight users, or at least nine users. PrefMoE has lower reported collapse than the corresponding full fine-tuning baseline in each group. Appendix G.1 complements this breakdown with ground-truth-to-prediction frequency transitions and distinguishes frequency-group agreement from preference correctness.

Table 7: Collapse breakdown for LLaVA-family models across user frequency groups.

Method	**Setting**	≤ 4	5–8	≥ 9	**Overall**
LLaVA-1.5-7B	*NT*	0.6020	0.6103	0.6760	0.6228
LLaVA-1.5-13B	*NT*	0.5750	0.5958	0.6490	0.6027
LLaVA-OV-72B	*NT*	0.5140	0.5333	0.5370	0.5297
LLaVA-1.5-7B	*FFT*	0.3180	0.3381	0.3790	0.3425
LLaVA-1.5-13B	*FFT*	0.2920	0.3056	0.3420	0.3105
LLaVA-OV-72B	*FFT*	0.2970	0.3056	0.3580	0.3151
PrefMoE (LLaVA-1.5-7B)	*PEFT*	0.1050	0.1175	0.1570	0.1233
PrefMoE (LLaVA-1.5-13B)	*PEFT*	0.1390	0.1330	0.1660	0.1416
PrefMoE (LLaVA-OV-72B)	*PEFT*	**0.0970**	**0.1079**	**0.1480**	**0.1142**

F.2 Detailed Comparisons for Different Preference Facets

Tables 8 and 9 provide a facet-wise analysis of 0-turn preference questions across entertainment, fashion, lifestyle, shopping, and travel. PrefMoE consistently improves preference accuracy and reduces collapse across LLaVA-family backbones, showing that its gains are not only reflected in aggregate metrics but also hold across different preference types. For example, on LLaVA-OV-72B, PrefMoE improves the overall preference accuracy from 0.5947 to 0.7613 and reduces the overall collapse rate from 0.3151 to 0.1142 compared with FFT.

The results also reveal clear facet-level difficulty differences. Shopping achieves particularly high preference accuracy under PrefMoE, suggesting that its cues are often more visually grounded and easier to associate with the image. In contrast, lifestyle remains more challenging, with lower accuracy and relatively higher collapse across backbones. This suggests that lifestyle preferences require more abstract and context-dependent personalization rather than direct visual matching. Overall, scaling and full fine-tuning improve general performance, but they do not sufficiently preserve fine-grained preference boundaries. PrefMoE provides more consistent preference-sensitive reasoning across facets.

Table 8: Facet-wise preference collapse on the LLaVA family.

Method	Setting	Entertainment	Fashion	Lifestyle	Shopping	Travel	Overall
LLaVA-1.5-7B	*NT*	0.6250	0.6458	0.6222	0.6000	0.6364	0.6228
LLaVA-1.5-13B	*NT*	0.5833	0.6250	0.6000	0.5778	0.6061	0.6027
LLaVA-OV-72B	*NT*	0.5208	0.5625	0.4889	0.5333	0.5455	0.5297
LLaVA-1.5-7B	*FFT*	0.3542	0.3750	0.3333	0.2889	0.3636	0.3425
LLaVA-1.5-13B	*FFT*	0.3125	0.2917	0.2889	0.2667	0.3939	0.3105
LLaVA-OV-72B	*FFT*	0.3333	0.3542	0.3111	0.2444	0.3030	0.3151
PrefMoE (LLaVA-1.5-7B)	*PEFT*	0.1458	0.0417	0.1778	0.0889	0.1212	0.1233
PrefMoE (LLaVA-1.5-13B)	*PEFT*	0.1667	**0.1042**	0.2000	0.1111	0.0909	0.1416
PrefMoE (LLaVA-OV-72B)	*PEFT*	**0.1250**	0.0625	**0.1556**	**0.0667**	**0.0303**	**0.1142**

Table 9: Facet-wise 0-turn preference accuracy on the LLaVA family.

Method	Type	Entertainment	Travel	Lifestyle	Shopping	Fashion	Ovr.Preference
LLaVA-1.5-7B	*NT*	0.3310	0.3440	0.3370	0.3420	0.3399	0.3387
LLaVA-1.5-13B	*NT*	0.3590	0.3670	0.3610	0.3710	0.3688	0.3653
LLaVA-OV-72B	*NT*	0.4740	0.4860	0.4770	0.4820	0.4812	0.4800
LLaVA-1.5-7B	*FFT*	0.4370	0.4560	0.3870	0.4620	0.4626	0.4413
LLaVA-1.5-13B	*FFT*	0.4520	0.4710	0.4050	0.4860	0.4846	0.4600
LLaVA-OV-72B	*FFT*	0.5850	0.6130	0.5310	0.6310	0.6121	0.5947
PrefMoE (LLaVA-1.5-7B)	*PEFT*	0.6120	0.6522	0.5860	0.8420	0.6810	0.6733
PrefMoE (LLaVA-1.5-13B)	*PEFT*	0.6230	0.6557	0.5920	0.8310	**0.7040**	0.6800
PrefMoE (LLaVA-OV-72B)	*PEFT*	**0.7800**	**0.8594**	**0.6040**	**0.8580**	0.6990	**0.7613**

F.3 Computational Cost

Table 10: Computational cost comparison. GPU-hours are computed as the number of GPUs multiplied by wall-clock time.

Method	Train Cost	Eval Cost	Peak Memory
LLaVA-1.5-7B	–	0.5–1 GPU-hours	15–20 GB/GPU
Yo'LLaVA-7B	5–10 GPU-hours	1–2 GPU-hours	28–34 GB/GPU
LOVA3-7B	6–12 GPU-hours	1–2 GPU-hours	28–34 GB/GPU
TG-LLaVA-7B	6–12 GPU-hours	1–2 GPU-hours	28–34 GB/GPU
PrefMoE (LLaVA-1.5-7B)	8–16 GPU-hours	1–2 GPU-hours	28–34 GB/GPU
PrefMoE (LLaVA-1.5-13B)	24–32 GPU-hours	2–3 GPU-hours	34–39 GB/GPU
PrefMoE (LLaVA-OV-72B)	450–600 GPU-hours	40–80 GPU-hours	38–40 GB/GPU

Table 10 reports the computational cost of different methods. The no-training LLaVA-1.5-7B baseline only requires inference, resulting in the lowest cost and memory usage. Existing personalized VQA methods introduce additional training overhead, but their costs remain in a similar range under 7B backbones. Compared with these methods, PrefMoE has a slightly higher training cost on LLaVA-1.5-7B due to its preference-aware modules and auxiliary objectives, while maintaining comparable evaluation cost and peak memory. When scaling to larger backbones, the cost increases mainly with model size: PrefMoE on LLaVA-1.5-13B requires more training resources, and the LLaVA-OV-72B variant is substantially more expensive due to the larger backbone. Overall, PrefMoE introduces moderate overhead at the 7B scale, while the major cost increase comes from backbone scaling rather than the personalization mechanism itself.

G Additional Data and Evaluation Validation

All experiments in this section use LLaVA-1.5-7B. The 0-turn and 10-turn settings refer to dialogue history, as in the main experiments. They are distinct from the sequential model updates studied in Appendix H.

G.1 Popularity-Stratified Prediction Transitions

We group preferences by their frequency across users, following the popularity thresholds used in the main appendix: High denotes at least nine users, Mid denotes five to eight users, and Low denotes at most four users. Figure 10 reports row-normalized transitions from the ground-truth preference

group to the predicted preference group. These coarse groups describe prediction direction: the diagonal indicates agreement in frequency group and does not necessarily indicate a correct preference prediction. Consequently, the coarse transition matrix and the preference-level collapse score should not be treated as interchangeable measures.

PrefMoE reduces the Mid-to-High transition from 0.293 for TG-LLaVA to 0.095, and the Low-to-High transition from 0.280 to 0.081. TG-LLaVA and LOVA3 have similar overall answer accuracies (0.5413 and 0.5329), but different transition patterns and reported collapse scores (0.2329 and 0.4292). This comparison illustrates why answer accuracy alone does not describe the direction of preference errors.

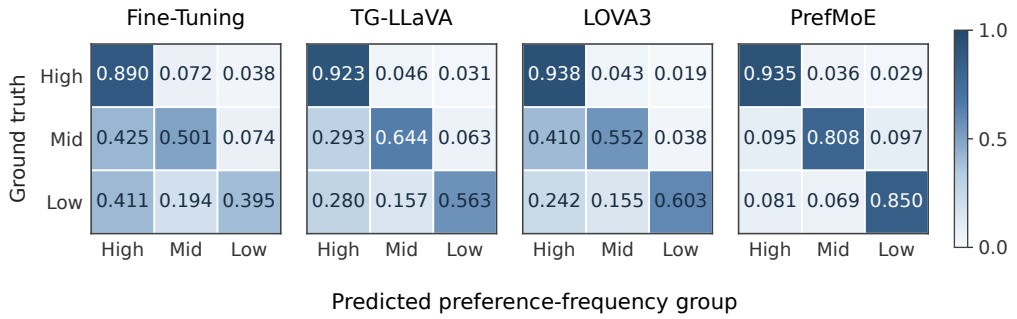

Figure 10: Row-normalized transitions from ground-truth to predicted preference-frequency groups. High, Mid, and Low denote at least nine, five to eight, and at most four users, respectively. All panels share the same color scale. Each row sums to one; the diagonal denotes a matching frequency group, not exact-answer accuracy.

G.2 Multiple Acceptable Preferences

We construct 40 questions with two human-verified acceptable preferences per question. A prediction is correct if it matches either acceptable preference. All acceptable alternatives are excluded from collapse errors before examining the direction of the remaining errors. Table 11 shows that PrefMoE retains the highest accuracy and lowest reported collapse among the compared methods in both dialogue settings. The result supports the same relative ordering when more than one answer is acceptable, within this small diagnostic set.

Table 11: Multi-label diagnostic with 40 questions and two acceptable preferences per question.

Method	0-turn		10-turn	
	Accuracy ↑	Collapse ↓	Accuracy ↑	Collapse ↓
Fine-Tuning	0.400	0.425	0.325	0.475
TG-LLaVA	0.475	0.325	0.425	0.375
PrefMoE	0.575	0.175	0.500	0.250

G.3 Human Audit of Collapse Errors

We manually audit 20 stratified prediction errors, including both metric-positive collapse cases and metric-negative ordinary errors. Annotators examine whether a prediction follows the population Top-1 preference or instead constitutes a plausible or non-dominant error. The automatic classification agrees with the human judgments in all 20 audited cases. This finding supports the interpretation of the selected cases. The small, stratified Top-1 audit does not establish a population-wide agreement rate or validate every possible form of preference shift.

G.4 Explicit Context Conditioning

We additionally evaluate preference queries with sunny, rainy, and cloudy contexts. Table 12 reports the 0-turn results. PrefMoE obtains the highest answer accuracies and lowest reported collapse among the compared methods. This controlled experiment tests explicit contextual conditioning, but it does not establish that the model can infer arbitrary changes in a user's preferences from naturally evolving context.

Table 12: Preference queries conditioned on explicit weather contexts (0-turn).

Method	Overall ↑	Preference ↑	Profile ↑	Collapse ↓
Fine-Tuning	0.4921	0.4236	0.5289	0.3714
Yo'LLaVA	0.4794	0.4618	0.4889	0.2348
TG-LLaVA	0.5155	0.5524	0.4957	0.2586
LOVA3	0.5079	0.5417	0.4898	0.4568
PrefMoE	0.6542	0.6487	0.6571	0.1519

H Sequential Personalization Updates

We study changes in personalized information separately from the dialogue-history setting. All results in this section use LLaVA-1.5-7B.

H.1 Two-Round Preference and Profile Updates

We simulate two sequential update rounds, each modifying the preference information or the profile information of 30% of users. The preference-change and profile-change experiments are evaluated separately. In the profile-update experiment, PrefMoE re-encodes the updated textual profile and updates the corresponding user-specific factors, while leaving the backbone and unaffected users' representations unchanged. Figures 11 and 12 compare the initial and final scores, with connecting segments showing the changes for each method.

PrefMoE retains the highest final accuracies and lowest reported collapse in both experiments. However, its overall accuracy decreases by 0.0437 with preference changes and by 0.0634 with profile changes, compared with decreases of 0.0410 and 0.0539 for TG-LLaVA. Its stronger final performance therefore should not be interpreted as evidence that it degrades less during updating.

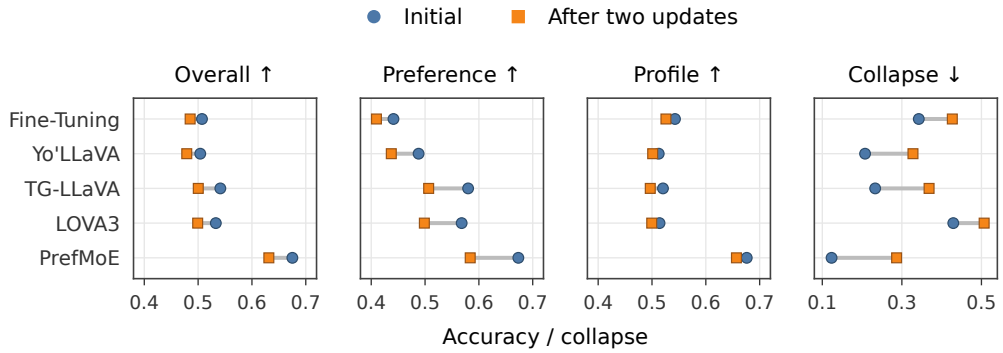

Figure 11: Scores before and after two sequential rounds of preference changes. Circles show initial scores and squares show final scores; each segment connects the same method before and after updating. Overall, Preference, and Profile denote accuracies (higher-is-better); Collapse is lower-is-better.

H.2 Ten Sequential Preference Updates

We divide the users into ten groups and construct ten sequential update tasks. At task t, only group t undergoes a preference change, while other groups' preference states remain unchanged in that task. Each update uses the newly changed user data without revisiting data from previous updates. We evaluate preference accuracy using a consistent evaluation protocol across stages and report each group's trajectory from its update onward. Unreported entries before a group's update are omitted from the plot.

Table 13 summarizes the initial and final preference accuracies by taking an unweighted mean across the ten reported groups. Figure 13 shows every reported group trajectory alongside its initial score. The summary drop is the mean initial preference accuracy minus the mean accuracy after update ten. Positive values indicate a decline.

After ten preference updates, PrefMoE still achieves higher overall accuracy than TG-LLaVA (0.6065 vs. 0.4956), despite a larger absolute decline (0.0686 vs. 0.0457). This difference may partly reflect their unequal starting performance: TG-LLaVA begins with substantially lower accuracy (0.5413

vs. 0.6751), which may contribute to its smaller absolute decline over the limited update horizon. Nevertheless, its smaller decline does not translate into better final performance, and PrefMoE maintains its advantage at every reported group–update pair.

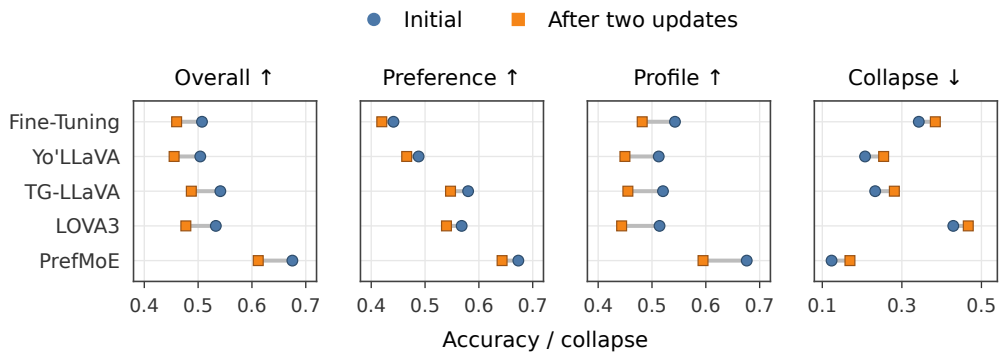

Figure 12: Scores before and after two sequential rounds of profile changes, using the same layout and axis ranges as Figure 11. Circles show initial scores and squares show final scores. Accuracies are higher-is-better and collapse is lower-is-better.

Table 13: Ten-update accuracy, averaged equally across the ten user groups. These summary values are computed from the group trajectories below.

Method	Initial	After update 10	Initial-to-final drop
TG-LLaVA	0.5413	0.4956	0.0457
PrefMoE	0.6751	0.6065	0.0686

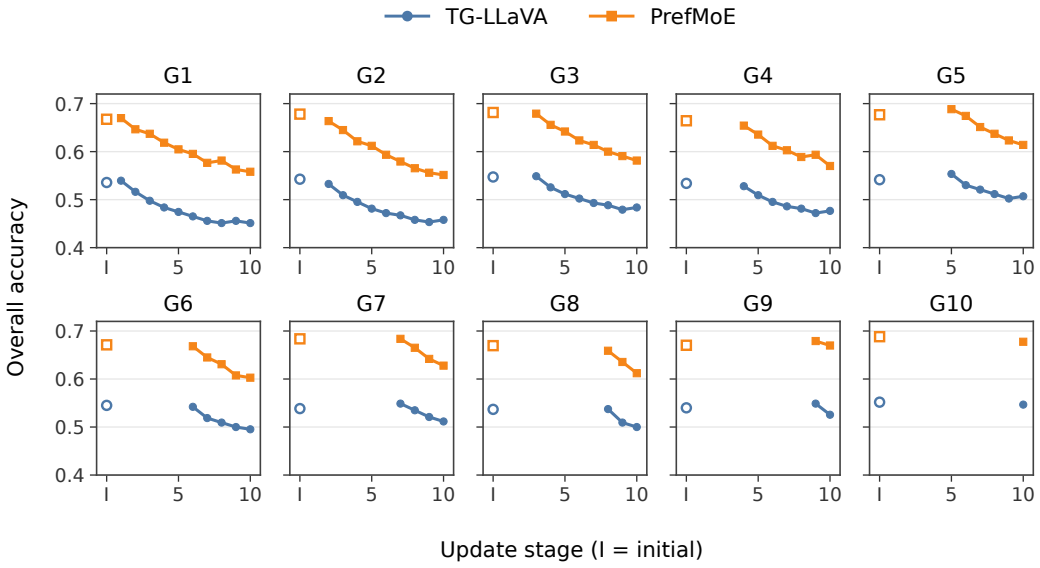

Figure 13: Overall accuracy across ten sequential updates, with one panel per user group. TG-LLaVA and PrefMoE are evaluated under the same protocol. Open markers at I show initial scores; filled markers show reported update-stage scores. Group t changes at update t, and unreported pre-update entries are omitted without interpolation. All panels share the same axes.