Uncertainty-Aware Variational Reward Factorization via Probabilistic Preference Bases for LLM Personalization

Gyuseok Lee[1] **Wonbin Kweon**[1] **Zhenrui Yue**[1]
SeongKu Kang[2] **Jiawei Han**[1] **Dong Wang**[1]
[1]University of Illinois Urbana-Champaign [2]Korea University
{gyuseok2,wonbin,zhenrui3,hanj,dwang24}@illinois.edu seongkukang@korea.ac.kr

Abstract

Reward factorization personalizes large language models (LLMs) by decomposing rewards into shared basis functions and user-specific weights. Yet, existing methods estimate user weights from scarce data in isolation and as deterministic points, leading to inaccurate and unreliable inference. We introduce Variational Reward Factorization (VRF), an uncertainty-aware framework that represents each user's preferences as a variational distribution in a shared preference space. VRF infers user distributions via a variational encoder, derives weights through Wasserstein distance matching with shared probabilistic bases, and downweights uncertain estimates through a variance-attenuated loss. On three benchmarks, VRF outperforms all baselines across seen and unseen users, few-shot scenarios, and varying uncertainty levels, with gains extending to downstream alignment. Our code is available at https://github.com/Gyu-Seok-Lee/VRF_COLM26.

1 Introduction

Aligning large language models (LLMs) with human preferences has attracted growing interest (Ziegler et al., 2019; Bai et al., 2022; Touvron et al., 2023; Rafailov et al., 2024). One prominent line of research is reward modeling, which first learns preferences from pairwise feedback, then guides LLM alignment (e.g., RLHF) (Christiano et al., 2017; Stiennon et al., 2020; Ouyang et al., 2022). However, conventional approaches learn a single, shared reward model that inevitably collapses toward the average user, neglecting the diverse and often conflicting preferences (Bakker et al., 2022; Casper et al., 2023; Durmus et al., 2024).

Beyond this one-size-fits-all assumption, personalized alignment of LLMs has been explored (Kirk et al., 2024a; Sorensen et al., 2024). A leading approach is reward factorization (Bose et al., 2025; Shenfeld et al., 2025), which decomposes a scalar reward as $r_u = \mathbf{w}_u^\top \boldsymbol{\phi}$, where $\mathbf{w}_u \in \Delta^{K-1}$ are user-specific weights and $\boldsymbol{\phi} \in \mathbb{R}^K$ are shared basis reward functions, with each ϕ_k capturing a distinct preference aspect. This factorization enables efficient personalization without training a per-user reward model (Zollo et al., 2024; Cai et al., 2026).

However, existing reward factorization methods face two fundamental challenges in modeling user weights $\mathbf{w}_u$: (C1) *isolated preference modeling* and (C2) *deterministic point estimates*.

(C1) Inaccurate estimation from isolated preference modeling: Inferring user preferences requires sufficient observations, yet per-user preference data is typically scarce[1] (Christiano et al., 2017; Ryan et al., 2025), leading to inaccurate estimation. While users often share common preference patterns that could compensate for limited data, existing methods estimate each user's weight in isolation, failing to leverage this commonality. This problem extends to unseen users, who need to estimate $\mathbf{w}_u$ from scratch without any learned structure. We empirically confirm that performance generally degrades with less data in our experiments.

(C2) Unreliable inference from deterministic point estimates: User preference histories inherently carry uncertainty (Casper et al., 2023; Siththaranjan et al., 2024), arising from

[1]e.g., users in PRISM (Kirk et al., 2024b) have only 5.7 conversations on average.

User A: Consistent Preferences (Certain)			User B: Diverse Preferences (Uncertain)		
Chosen	Rejected	Preferences	Chosen	Rejected	Preferences
"Here are the key facts..."	"I feel like the answer is..."	Factual	"Here are the key facts..."	"I feel like the answer is..."	Factual
"The data shows that..."	"Maybe it could be..."	Factual	"Great question! Let me..."	"Per the specification, ..."	Empathetic
"According to studies..."	"I suppose it's..."	Factual	"Sure thing! It's A."	"The answer is as follows."	Casual

Figure 1: Motivating example. Preferences can be consistent (certain) or diverse (uncertain).

both data scarcity and preference diversity. Yet existing methods estimate the user weight as a deterministic point, which cannot express how reliably the estimate captures the user's underlying preferences. As illustrated in Figure 1, users exhibit varying preference patterns: User A is consistent within a single style, while User B spreads across multiple styles. Intuitively, the estimate for User B should entail greater uncertainty than that for User A. However, without uncertainty quantification, deterministic point estimates implicitly assign uniform certainty to every user, leading to unreliable inference of user preferences.

In this paper, we propose **V**ariational **R**eward **F**actorization (**VRF**) for uncertainty-aware LLM personalization. The core idea of VRF is to represent each user not as a deterministic point but as a variational distribution in a shared preference space, from which user-specific weights are derived. Specifically, VRF introduces three key components: (1) a variational encoder that infers each user's preferences as a Gaussian distribution from a small reference set, (2) shared probabilistic preference bases with Wasserstein distance-based matching that jointly accounts for preference direction and confidence, and (3) a variance-attenuated preference loss that downweights uncertain reward estimates during training.

VRF systematically addresses both challenges. For **C1**, the shared preference bases encode population-level patterns, allowing data-scarce users to benefit from collective signals. For **C2**, the probabilistic formulation of users and bases enables uncertainty-aware preference modeling, supported by the variance-attenuated loss. At inference, the variational encoder with shared bases enables effective user weight estimation for unseen users in a single forward pass without any optimization. Our core contributions are as follows:

- We identify two fundamental challenges of existing reward factorization approaches that remain underexplored in the literature, highlighting the need for systematic investigation.
- We propose VRF, an uncertainty-aware reward factorization framework that represents users and preference bases as probabilistic distributions over a shared preference space.
- We validate VRF on three datasets, showing consistent improvements over state-of-the-art baselines on seen and unseen users. Further analyses highlight its robustness across varying data scarcity and uncertainty levels, with gains extending to downstream alignment.

2 Related Work

LLM Personalization. Personalizing LLMs to individual user preferences has been explored along several axes. Prompt-based methods incorporate user histories (Dai et al., 2023; Li et al., 2023) or retrieval-augmented context (Richardson et al., 2023; Salemi et al., 2024) into the input without additional training. Other approaches personalize LLMs at the parametric level, including per-user adapters (Tan et al., 2024a;b), user embeddings (Doddapaneni et al., 2024; Liu et al., 2025), and steering vectors (Cao et al., 2024; Zhang et al., 2025). While these methods implicitly encode user preferences at either the input or parametric level, our approach explicitly models them within the reward modeling framework.

Personalized Reward Modeling. Reward modeling directly learns human preferences from pairwise comparisons to guide LLM alignment (e.g., RLHF) (Christiano et al., 2017; Stiennon et al., 2020; Ouyang et al., 2022). However, conventional approaches learn a single, shared reward model that fails to capture diverse user preferences (Bakker et al., 2022; Casper et al., 2023; Durmus et al., 2024). To address this, reward factorization represents the personalized reward using shared basis functions and user-specific weights (Bose et al., 2025; Chen et al., 2025; Shenfeld et al., 2025). Yet, existing methods face two key

limitations in modeling user weights: (1) isolated per-user estimation that ignores shared preference patterns, and (2) deterministic point estimates that lack uncertainty quantification. Although PAL (Chen et al., 2025) employs shared prototypes, they are designed for reward computation rather than user weight estimation, and its ideal point formulation lacks uncertainty quantification.[2] VPL (Poddar et al., 2024) introduces variational inference for measuring uncertainty, but infers each user in isolation, leading to posterior collapse under sparse feedback (Kim & Kim, 2026). These challenges motivate an uncertainty-aware reward factorization that leverages shared preference patterns beyond individual observations.

3 Preliminaries

3.1 Problem Formulation

For each user $u \in \mathcal{U}$, we partition pairwise preference data into two disjoint sets: a reference set $\mathcal{C}_u$ for inferring user preferences and a target set $\mathcal{D}_u$ for computing the personalized reward. Each set contains triplets (x, y^+, y^-), where $x \in \mathcal{X}$ is a query and $y^+, y^- \in \mathcal{Y}$ are chosen and rejected responses. Our goal is to learn a personalized reward function $r_u(x, y|\mathcal{C}_u)$ over $\mathcal{D}_u$. The reward function is then used for personalized alignment of a policy model, either at training time like RLHF (Christiano et al., 2017; Ouyang et al., 2022) or at inference time like best-of-N sampling (Stiennon et al., 2020; Touvron et al., 2023).

3.2 User-Agnostic Reward Modeling

Early approaches to reward modeling (Ziegler et al., 2019; Stiennon et al., 2020; Ouyang et al., 2022) adopt the Bradley-Terry (BT) model (Bradley & Terry, 1952) to formulate the probability that response y^+ is preferred over y^- given x as:

$$P(y^+ \succ y^- \mid x) = \sigma\big(r_\theta(x, y^+) - r_\theta(x, y^-)\big), \tag{1}$$

where $r_\theta(x, y) \in \mathbb{R}$ is a single reward function shared across all users and $\sigma(\cdot)$ is the sigmoid function. r_θ is optimized by minimizing the negative log-likelihood as follows:

$$\mathcal{L}_{\text{BT}}(\theta) = -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}}\big[\log \sigma\big(r_\theta(x, y^+) - r_\theta(x, y^-)\big)\big], \tag{2}$$

where $\mathcal{D} = \bigcup_{u\in\mathcal{U}} \mathcal{D}_u$. Since r_θ is optimized over the aggregated data across users, it collapses heterogeneous preferences into a generic reward. For instance, given a prompt x, if User A prefers a formal response y_1 over a casual one y_2, while User B prefers the opposite, r_θ averages these conflicting signals ($r_\theta(x, y_1) \approx r_\theta(x, y_2)$), thereby failing to satisfy either user.

3.3 Reward Factorization

To capture individual user preferences, recent work (Bose et al., 2025; Shenfeld et al., 2025) factorizes the single scalar reward as follows:

$$r_u(x, y) = \mathbf{w}_u^\top \boldsymbol{\phi}(x, y), \tag{3}$$

where $\boldsymbol{\phi}(x, y) = [\phi_1(x, y), \ldots, \phi_K(x, y)]^\top \in \mathbb{R}^K$ denotes K shared basis reward functions parameterized by a neural network, with each ϕ_k scoring a distinct preference style (e.g., conciseness or formality). $\mathbf{w}_u \in \Delta^{K-1}$ is a user weight that personalizes the reward by determining the relative importance among the bases. The training objective is as follows:

$$\mathcal{L}_{RF}(\mathbf{w}, \boldsymbol{\phi}) = -\mathbb{E}_{u\sim\mathcal{U}}\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}_u}\left[\log \sigma\big(r_u(x, y^+) - r_u(x, y^-)\big)\right]. \tag{4}$$

[2]Other approaches leverage cross-user signals but do not adopt reward factorization (Li et al., 2024; Choi et al., 2025), or share reward heads but route at the textual level without per-user inference (Shen et al., 2025). None of these methods address uncertainty in user weight estimation. DPL (Siththaranjan et al., 2024) captures reward uncertainty but does not personalize to individual users.

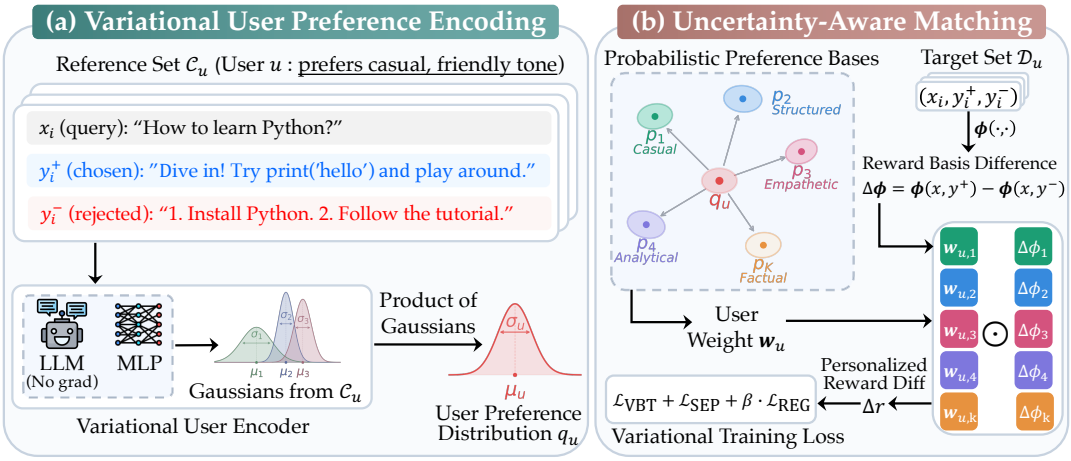


Figure 2: Overall framework of VRF. Best viewed in color.

However, existing reward factorization methods estimate $\mathbf{w}_u$ in isolation without leveraging shared preference patterns (C1), and represent it as a deterministic point without uncertainty quantification (C2), resulting in inaccurate and unreliable personalization.

4 Variational Reward Factorization

To address these challenges, we propose **Variational Reward Factorization** (**VRF**), which represents each user's preferences not as a deterministic point, but as a variational distribution in a shared preference space, from which the user weight is derived. We first infer the user preference distribution (§4.1), then derive the user weight in probabilistic preference space (§4.2). We further propose uncertainty-aware training objectives (§4.3). Finally, we present few-shot inference for unseen users (§4.4). Figure 2 shows an overview of VRF.

4.1 Variational User Preference Encoding

We propose a variational user encoder that infers each user's preference distribution $q_u = \mathcal{N}(\boldsymbol{\mu}_u, \text{diag}(\boldsymbol{\sigma}_u^2))$ from a small reference set $\mathcal{C}_u$. Here, $\boldsymbol{\mu}_u$ encodes the preference location and $\boldsymbol{\sigma}_u$ captures its uncertainty, which is inherently absent in deterministic point estimates.

Variational Pairwise Encoding. Given the reference set $\mathcal{C}_u$ for user u, we form a pairwise representation $\mathbf{v}_i \in \mathbb{R}^{2H}$ from each triplet[3] (x_i, y_i^+, y_i^-). Then, a two-layer MLP maps $\mathbf{v}_i$ to Gaussian parameters $\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i \in \mathbb{R}^D$ (Kingma & Welling, 2014), resulting in $|\mathcal{C}_u|$ Gaussians (i.e., $\{\mathcal{N}(\boldsymbol{\mu}_i, \text{diag}(\boldsymbol{\sigma}_i^2))\}_{i=1}^{|\mathcal{C}_u|}$).

User Preference Aggregation. To aggregate $|\mathcal{C}_u|$ Gaussians into a single user distribution $q_u = \mathcal{N}(\boldsymbol{\mu}_u, \text{diag}(\boldsymbol{\sigma}_u^2))$, we adopt the product of Gaussians (Bishop, 2006), which yields a closed-form aggregation:

$$\boldsymbol{\mu}_u = \frac{\sum_{i=1}^{M_u} \boldsymbol{\mu}_i \odot \boldsymbol{\sigma}_i^{-2}}{\sum_{i=1}^{M_u} \boldsymbol{\sigma}_i^{-2}}, \quad \boldsymbol{\sigma}_u^2 = \frac{1}{\sum_{i=1}^{M_u} \boldsymbol{\sigma}_i^{-2}}, \tag{5}$$

where $\odot$ is element-wise multiplication. Unlike simple averaging, the product of Gaussians enables more reliable estimation, weighting each observation by its precision $\boldsymbol{\sigma}_i^{-2}$. Thus, $\boldsymbol{\mu}_u$ reflects more confident observations while $\boldsymbol{\sigma}_u$ decreases as more observations are available.

[3] $\mathbf{v}_i = [\frac{\mathbf{h}_i^+ + \mathbf{h}_i^-}{2} \| \mathbf{h}_i^+ - \mathbf{h}_i^-] \in \mathbb{R}^{2H}$, where $\mathbf{h}_i^+ = \text{LLM}(x_i \| y_i^+) \in \mathbb{R}^H$ and $\mathbf{h}_i^- = \text{LLM}(x_i \| y_i^-) \in \mathbb{R}^H$ are last-token hidden states extracted under no gradient and $\|$ denotes concatenation. The mean and difference capture shared content and preference direction, respectively.

4.2 Uncertainty-Aware Preference Matching

Given the user distribution $q_u = \mathcal{N}(\boldsymbol{\mu}_u, \mathrm{diag}(\boldsymbol{\sigma}_u^2))$, we derive the user weight $\mathbf{w}_u \in \Delta^{K-1}$ over K shared probabilistic preference bases (i.e., Gaussians). By representing both users and bases as distributions, we enable uncertainty-aware preference matching between them.

Probabilistic Preference Bases. We maintain K shared preference bases $\mathcal{P} = \{p_k\}_{k=1}^K$, where $p_k = \mathcal{N}(\boldsymbol{\mu}_k, \mathrm{diag}(\boldsymbol{\sigma}_k^2))$ with $\boldsymbol{\mu}_k, \boldsymbol{\sigma}_k \in \mathbb{R}^D$ as learnable parameters. Each p_k represents a distinct preference style (e.g., conciseness, formality, or empathy), corresponding to the k-th reward function $\phi_k(x, y)$. The bases $\mathcal{P}$ are randomly initialized and learned to capture domain-specific preference patterns that minimize the overall training objective.

The preference bases $\mathcal{P}$ offer three key benefits: (1) sharing bases across users encodes population-level patterns, allowing data-sparse and unseen users to benefit from collective signals (C1); (2) modeling bases as Gaussians enables uncertainty-aware matching with q_u, naturally incorporating preference uncertainty into $\mathbf{w}_u$ (C2); and (3) $\mathcal{P}$ is parameter-efficient, as the number of bases remains fixed regardless of the user population size.

User Weight Derivation via Wasserstein Distance. We derive the user weight $\mathbf{w}_u$ via the squared 2-Wasserstein distance[4] $\mathcal{W}_2(\cdot,\cdot)^2$ between the user distribution q_u and each preference basis p_k. For diagonal Gaussians, this admits a closed form $\mathcal{W}_2(q_u, p_k)^2 = \frac{1}{D}(\|\boldsymbol{\mu}_u - \boldsymbol{\mu}_k\|_2^2 + \|\boldsymbol{\sigma}_u - \boldsymbol{\sigma}_k\|_2^2)$ (Villani, 2008), which naturally accounts for both preference location and uncertainty. We then derive $\mathbf{w}_u$ from these distances via softmax:

$$\mathbf{w}_u = [w_{u,1}, \ldots, w_{u,K}]^\top \in \Delta^{K-1}, \quad w_{u,k} = \frac{\exp(-\mathcal{W}_2(q_u, p_k)^2/\tau_d)}{\sum_{j=1}^K \exp(-\mathcal{W}_2(q_u, p_j)^2/\tau_d)}, \tag{6}$$

where τ_d is a temperature hyperparameter. The resulting $\mathbf{w}_u$ encodes an uncertainty-aware soft assignment over preference bases, yielding the personalized reward in Eq. (3).

4.3 Training Objectives

Variance-Attenuated Bradley-Terry Loss. Under the original Bradley-Terry loss $\mathcal{L}_{\mathrm{BT}}$, the pairwise preference probability is computed as $P(y^+ \succ y^- \mid x) = \sigma(\Delta r_{\mathrm{BT}})$, where $\Delta r_{\mathrm{BT}} = r_\theta(x, y^+) - r_\theta(x, y^-)$ is a scalar reward difference. However, this formulation does not account for uncertainty in reward estimation. To address this, we propose a variance-attenuated preference loss that replaces the scalar Δr_{BT} with a Gaussian $\Delta r_u \sim \mathcal{N}(\mu_\Delta, \sigma_\Delta^2)$.

ddLet $\Delta\boldsymbol{\phi} = [\Delta\phi_1, \ldots, \Delta\phi_K]^\top \in \mathbb{R}^K$ denote the reward basis differences, where $\Delta\phi_k = \phi_k(x, y^+) - \phi_k(x, y^-)$. Given the user weight $\mathbf{w}_u$, the Gaussian parameters are computed as:

$$\mu_\Delta = \sum_{k=1}^K w_{u,k}\Delta\phi_k, \quad \sigma_\Delta^2 = \sum_{k=1}^K w_{u,k}(\Delta\phi_k - \mu_\Delta)^2, \tag{7}$$

where μ_Δ and σ_Δ^2 are the mean and variance[5] of the reward basis differences weighted by $\mathbf{w}_u$. We approximate $\Delta r_u \sim \mathcal{N}(\mu_\Delta, \sigma_\Delta^2)$ via moment matching (Minka, 2001). The pairwise preference probability of user u then becomes $P(y^+ \succ y^- \mid x, u) = \mathbb{E}_{\Delta r_u}[\sigma(\Delta r_u)]$. As this expectation is analytically intractable[6], we apply MacKay's Approximation (MacKay, 1992) to derive a closed-form variance-attenuated Bradley-Terry loss:

$$\mathcal{L}_{\mathrm{VBT}}(\mathbf{w}, \boldsymbol{\phi}, \mathcal{P}) = -\mathbb{E}_{u\sim\mathcal{U}}\, \mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}_u}\left[\log\sigma\left(\frac{\mu_\Delta}{\sqrt{1+\pi\sigma_\Delta^2/8}}\right)\right]. \tag{8}$$

[4]Unlike KL divergence, $\mathcal{W}_2$ is symmetric and considers the geometric structure of the latent space.

[5]σ_u and σ_Δ capture uncertainty at different levels: σ_u over the user preference distribution q_u, and σ_Δ over the reward estimate for a specific pair given $\mathbf{w}_u$.

[6]$\mathbb{E}_{\Delta r_u}[\sigma(\Delta r_u)] = \int_{-\infty}^{\infty} \sigma(\Delta r_u)\mathcal{N}(\Delta r_u \mid \mu_\Delta, \sigma_\Delta^2)\, d(\Delta r_u)$, which has no analytic solution.

$\mathcal{L}_{\text{VBT}}$ adaptively scales the gradient via σ_Δ^2, which depends on the concentration of $\mathbf{w}_u$: (1) when $\mathbf{w}_u$ is concentrated (e.g., one-hot), $\sigma_\Delta^2 = 0$ and $\mathcal{L}_{\text{VBT}}$ recovers $\mathcal{L}_{\text{BT}}$; (2) as $\mathbf{w}_u$ becomes more diffuse, σ_Δ^2 grows and attenuates the gradient (see Appendix C.1 and C.2). Consequently, $\mathcal{L}_{\text{VBT}}$ enables uncertainty-aware reward estimation.

Variational Training Objective. Following variational inference (Kingma & Welling, 2014), we derive the final training objective as:

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\phi}, \mathcal{P}) = \mathcal{L}_{\text{VBT}} + \mathcal{L}_{\text{SEP}} + \beta \cdot \mathcal{L}_{\text{REG}}. \tag{9}$$

$\mathcal{L}_{\text{SEP}} = \sum_{1 \le i < j \le K} \max(0, \tau_{\text{m}} - \mathcal{W}_2(p_i, p_j)^2)$ promotes diversity among the K preference bases with margin τ_m. $\mathcal{L}_{\text{REG}} = \mathbb{E}_{u \sim \mathcal{U}}[D_{\text{KL}}(q_u \parallel p(\mathbf{z}))]$ regularizes q_u to lie within the learned preference space using a mixture-of-Gaussians prior[7] $p(\mathbf{z}) = \frac{1}{K}\sum_{k=1}^{K} p_k$ instead of a standard $\mathcal{N}(\mathbf{0}, \mathbf{I})$. β controls the regularization strength.

4.4 Few-Shot Inference for Unseen Users

At inference time, the basis reward functions $\boldsymbol{\phi}$ are known, but the user weight $\mathbf{w}_{u'}$ for an unseen user u' is derived on-the-fly.[8] Given a small reference set $\mathcal{C}_{u'}$, the variational encoder infers the user distribution $q_{u'}$ (§4.1), from which $\mathbf{w}_{u'}$ is derived (§4.2). The personalized reward is then obtained via Eq. (3). Unlike methods requiring gradient-based optimization (Bose et al., 2025; Chen et al., 2025; Shenfeld et al., 2025), VRF estimates $\mathbf{w}_{u'}$ via a single forward pass without additional optimization.

5 Experiments

5.1 Experimental Setup

Datasets. We evaluate on three benchmarks: **PersonalLLM** (Zollo et al., 2024), where $\alpha \in \{0.001, 0.01, 0.1\}$ controls preference diversity; **TL;DR** (Stiennon et al., 2020), where $n \in \{100, 150\}$ denotes the number of training triplets per seen user; and **PRISM** (Kirk et al., 2024b). Detailed statistics are provided in Appendix A.1.

Evaluation Metric. We report pairwise preference accuracy as the primary metric: $\text{Acc} = \frac{1}{|\mathcal{D}_{\text{test}}|}\sum_{(x,y^+,y^-,u)\in\mathcal{D}_{\text{test}}} \mathbb{1}[r_u(x, y^+) > r_u(x, y^-)]$. Results are averaged over 5 random seeds, controlling model initialization and data splits. Performance is evaluated across three distinct splits: Seen, Unseen, and Overall (the macro-average of Seen and Unseen).

Baselines. We compare VRF with several baselines: the pretrained Ref (Qwen Team, 2025), non-personalized BT (Stiennon et al., 2020; Ouyang et al., 2022), and personalized methods including VPL (Poddar et al., 2024), PAL (Chen et al., 2025), LoRe (Bose et al., 2025), and PReF (Shenfeld et al., 2025). Detailed explanations of each baseline are in Appendix A.2.

Implementation Details. We employ Qwen3-0.6B (Qwen Team, 2025) as the backbone for the basis reward functions[9] $\boldsymbol{\phi}$, fully fine-tuned given its compact size. For VRF and VPL, the backbone is additionally used as the variational encoder without gradient updates.[10] All baselines are tuned following their original protocols. Detailed hyperparameter configurations are provided in Appendix A.3. Backbone scale from 0.6B to 8B is analyzed in Appendix B.1.

[7] Since D_{KL} against a mixture-of-Gaussians prior lacks a closed form, we estimate it via Monte Carlo sampling with the reparameterization trick (see Appendix C.3).

[8] For seen users, the learned weights $\mathbf{w}_u$ are cached during training and applied directly.

[9] $\boldsymbol{\phi}$ comprises the backbone and a two-layer MLP ($H \to H/4 \to K$) on the last-token hidden state.

[10] As the shared backbone is updated through $\boldsymbol{\phi}$, the encoder reflects preference learning without additional backward passes.

Table 1: Overall performance comparison on three datasets.

Method	PersonalLLM								
	Very Diverse (α=0.001)			Moderately Diverse (α=0.01)			Near Uniform (α=0.1)		
	Seen	Unseen	Overall	Seen	Unseen	Overall	Seen	Unseen	Overall
Ref	58.6±0.2	58.9±0.9	58.8±0.4	59.1±0.3	59.0±0.6	59.1±0.4	59.9±0.2	60.0±0.2	60.0±0.2
BT	90.3±0.3	90.5±0.4	90.4±0.2	91.1±0.4	91.0±0.3	91.1±0.2	94.8±0.2	94.6±0.4	94.7±0.2
VPL	86.4±0.4	86.9±0.5	86.6±0.2	87.3±0.5	87.4±0.3	87.3±0.2	91.7±0.2	91.5±0.5	91.6±0.3
PAL	88.8±0.5	89.3±0.1	89.0±0.3	89.3±0.5	89.2±0.3	89.2±0.2	93.0±0.2	92.8±0.4	92.9±0.3
LoRe	89.4±1.2	88.6±0.3	89.0±0.5	89.6±0.9	88.7±0.2	89.2±0.6	92.7±0.2	92.5±0.3	92.6±0.2
PReF	94.4±0.2	81.1±0.1	87.7±0.1	94.0±0.3	82.4±0.5	88.2±0.2	94.3±0.1	90.4±0.4	92.3±0.2
VRF	**95.6±0.4**	**95.1±0.3**	**95.3±0.3**	**95.5±0.1**	**94.7±0.2**	**95.1±0.1**	**96.8±0.0**	**96.4±0.2**	**96.6±0.1**

Method	TL;DR (n=100)			TL;DR (n=150)			PRISM		
	Seen	Unseen	Overall	Seen	Unseen	Overall	Seen	Unseen	Overall
Ref	51.9±0.2	51.8±0.7	51.8±0.3	51.9±0.2	51.8±0.7	51.8±0.3	54.7±0.7	54.9±0.3	54.8±0.5
BT	59.9±0.7	59.0±0.7	59.5±0.7	61.5±0.6	60.4±0.6	60.9±0.5	60.0±0.7	60.2±0.9	60.1±0.5
VPL	58.6±1.0	57.5±0.4	58.1±0.7	58.7±1.2	56.7±0.3	57.7±0.5	59.6±0.7	58.8±1.0	59.2±0.7
PAL	60.4±0.7	59.1±0.3	59.8±0.4	61.7±0.5	60.6±0.4	61.2±0.1	60.8±0.4	60.8±0.9	60.8±0.5
LoRe	61.6±0.5	61.0±0.6	61.3±0.6	62.3±0.8	61.0±0.7	61.7±0.6	60.4±1.0	59.7±0.9	60.0±0.9
PReF	55.7±2.5	56.7±2.2	56.2±1.9	59.9±1.3	59.2±1.8	59.5±1.5	53.1±1.6	51.4±1.2	52.3±1.3
VRF	**62.3±1.2**	**61.5±0.9**	**61.9±1.0**	**63.3±1.1**	**62.4±1.1**	**62.9±1.1**	**62.1±0.6**	**61.6±0.5**	**61.8±0.3**

5.2 Performance Comparison

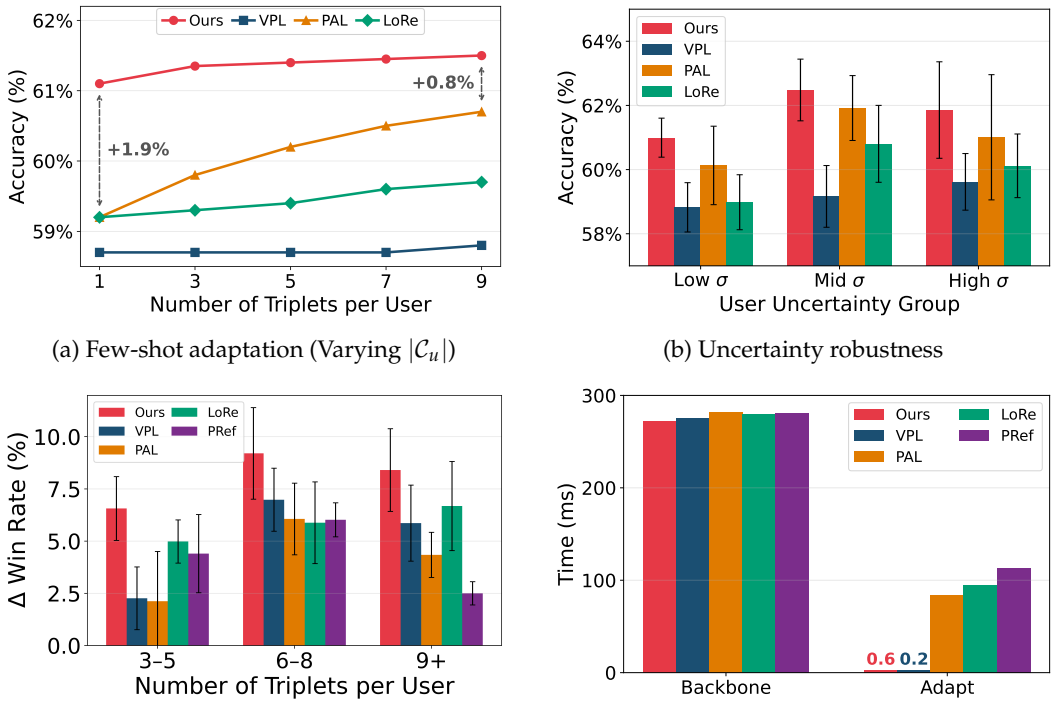
Main Results. Table 1 shows that VRF consistently achieves the best performance across all datasets and splits, indicating its effectiveness for personalized reward modeling across synthetic and real-world settings. The gains over BT highlight the importance of user-specific modeling, while improvements over VPL and PAL suggest that combining variational user representations with shared preference bases provides complementary benefits. Unlike LoRe and PReF, which rely on deterministic user representations, VRF represents both users and preference bases as probabilistic distributions, enabling uncertainty-aware preference modeling. This allows the model to quantify preference uncertainty and attenuate ambiguous preference signals during training, leading to more robust preference learning. Notably, on PersonalLLM, the performance gap widens as user preferences become more diverse (α=0.001), suggesting that VRF more effectively captures heterogeneous user preferences.

In-Depth Comparison. Figure 3 examines unseen users in PRISM across four dimensions: (a) few-shot adaptation, (b) uncertainty robustness, (c) inference-time alignment, and (d) adaptation efficiency. For (a) and (b), PReF is excluded due to its unstable performance. Note that (a) evaluates reward model accuracy by varying $|\mathcal{C}_u|$, while (c) measures downstream alignment quality via best-of-N sampling, grouping users by their original reference set size. Additional few-shot adaptation and uncertainty robustness results on PersonalLLM and TL;DR are provided in Appendix B.3.

(a) **Few-Shot Adaptation.** Figure 3a reports accuracy by varying $|\mathcal{C}_u| \in \{1, 3, 5, 7, 9\}$. VRF achieves the highest accuracy across all values of $|\mathcal{C}_u|$. Notably, the improvement over the best competitor (PAL) is largest at $|\mathcal{C}_u| = 1$ (+1.9%), indicating that VRF effectively infers user preferences even from minimal user history. This result suggests that shared preference bases enable data-scarce users to leverage collective knowledge from the user population. The variational encoder, regularized by the mixture-of-Gaussians prior, further maps users within this learned preference space. Together, these components address the isolated estimation problem (C1).

(b) **Uncertainty Robustness.** Figure 3b categorizes unseen users into three bins (low, mid, high) based on their preference uncertainty σ_u.[11] VRF consistently outperforms all baselines across all uncertainty levels. This robustness stems from our uncertainty-aware dis-

[11]To obtain a single uncertainty score per user, we average $\boldsymbol{\sigma}_u \in \mathbb{R}^D$ across dimensions.



(a) Few-shot adaptation (Varying $|\mathcal{C}_u|$)

(b) Uncertainty robustness

(c) Inference-time alignment (grouped by $|\mathcal{C}_u|$)

(d) Per-user adaptation time (ms)

Figure 3: In-depth analysis of unseen users in PRISM. VRF maintains superior accuracy across all user groups (a–c) while significantly reducing adaptation latency (d).

tributional matching between users and preference bases and variance-attenuated training, jointly mitigating the unreliable inference from deterministic point estimates (C2).

(c) **Inference-Time Alignment.** We evaluate inference-time alignment via best-of-N sampling to assess whether improved reward estimation translates to better response selection (see Appendix A.4 for detailed setup). Figure 3c shows that VRF consistently achieves the highest Δ Win Rate across all user groups. Even with only 3–5 reference pairs, VRF outperforms all baselines. These results suggest that the accuracy gains observed in (a) and (b) translate to tangible improvements in downstream alignment.

(d) **Adaptation Efficiency.** Figure 3d reports per-user adaptation latency. 'Backbone' denotes the time for LLM forward pass over the reference set, and 'Adapt' measures the time for estimating user weights. VRF and VPL compute user weights via a single forward pass (0.6ms and 0.2ms, respectively), whereas gradient-based methods (PAL, LoRe, PReF) require 83–114ms. This advantage becomes critical at scale; for 1M users, VRF's adaptation completes in under 10 minutes, whereas PReF requires over 30 hours. VRF thus enables efficient personalization with negligible adaptation overhead, making it practical for large-scale deployment.

Taken together, VRF effectively addresses both the isolated estimation (C1) and unreliable inference (C2) challenges across few-shot scenarios and diverse uncertainty levels. These gains translate to downstream personalized alignment while maintaining practical efficiency.

Uncertainty Calibration. To examine whether VRF's uncertainty-aware modeling translates into well-calibrated preference predictions, we compare it against BT on three calibration metrics: NLL (Gneiting & Raftery, 2007), Brier score (Brier, 1950), and ECE (Guo et al., 2017). BT predicts a deterministic preference probability (Eq. 1), whereas VRF attenuates it by the reward variance (Eq. 8). Table 2 shows that VRF achieves better calibration than BT across most settings, since attenuating the prediction under high variance prevents overconfidence when a user's preference is inferred from limited observations. The gain is large on PersonalLLM (α=0.001) and PRISM, where ECE drops by 39% and 68%. These

Table 2: Calibration of predicted preference probabilities on the overall split. Lower is better for all metrics; ECE uses ten equal-width bins.

Dataset	Model	NLL (↓)	Brier (↓)	ECE (↓)
PersonalLLM (α=0.001)	BT	0.2584±0.0069	0.0734±0.0017	0.0162±0.0034
	VRF	**0.1747**±0.0195	**0.0464**±0.0027	**0.0099**±0.0095
PRISM	BT	0.6862±0.0071	0.2402±0.0027	0.0844±0.0075
	VRF	**0.6348**±0.0049	**0.2237**±0.0018	**0.0269**±0.0127
TL;DR (n=100)	BT	0.7572±0.0399	0.2600±0.0086	0.1353±0.0292
	VRF	**0.6862**±0.0310	**0.2394**±0.0099	**0.0761**±0.0348
TL;DR (n=150)	BT	0.7277±0.0488	0.2499±0.0103	**0.1155**±0.0356
	VRF	**0.7203**±0.0183	**0.2452**±0.0060	0.1216±0.0087

Table 3: Ablation study on VRF across three datasets.

Model Variant	PersonalLLM (α=0.001)			TL;DR (n=150)			PRISM		
	Seen	Unseen	Overall	Seen	Unseen	Overall	Seen	Unseen	Overall
(a) w/o $\mathcal{L}_{\text{VBT}}$	92.7±0.3	92.5±0.7	92.6±0.3	61.4±0.5	60.8±0.3	61.1±0.4	58.3±0.2	58.1±0.5	58.2±0.4
(b) w/o $\mathcal{L}_{\text{SEP}}$	91.9±0.3	91.4±0.7	91.7±0.3	61.5±0.6	60.8±0.3	61.2±0.5	58.2±0.1	58.2±0.5	58.2±0.3
(c) w/o $\mathcal{L}_{\text{REG}}$	93.2±0.2	93.5±0.3	93.3±0.2	61.2±0.6	60.3±0.4	60.8±0.4	60.8±0.4	60.8±0.9	60.8±0.7
(d) w/o MoG	91.9±0.3	91.4±0.7	91.7±0.2	61.3±0.5	60.8±0.3	61.1±0.4	60.2±0.3	60.3±0.8	60.2±0.6
(e) w/o PoG	94.8±0.2	95.0±0.2	94.9±0.2	62.7±0.2	61.8±0.4	62.2±0.3	61.7±0.5	60.6±0.4	61.1±0.2
Full Model	**95.6**±0.4	**95.1**±0.3	**95.3**±0.3	**63.3**±1.1	**62.4**±1.1	**62.9**±1.1	**62.1**±0.6	**61.6**±0.5	**61.8**±0.3

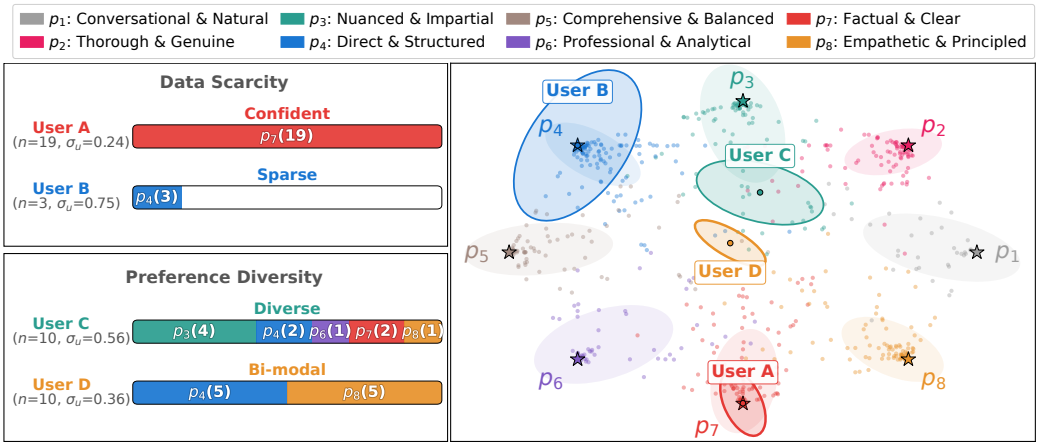
are the settings with the most diverse preferences and the fewest observations per user. A controlled comparison between the two TL;DR settings, which share the same users and differ only in the number of training triplets per user, shows that the gain is larger under sparser data (n=100). Overall, VRF's uncertainty-aware preference modeling yields more reliable predictions than deterministic point estimates (C2).

5.3 Study of VRF

Ablation Study. Table 3 reports the contribution of each component in VRF. (a) Removing $\mathcal{L}_{\text{VBT}}$ (replaced with $\mathcal{L}_{\text{BT}}$) consistently degrades accuracy across all datasets, confirming its role in adaptively scaling the gradients under preference uncertainty. (b) Removing $\mathcal{L}_{\text{SEP}}$ causes the largest drop on PersonalLLM (-3.6%), indicating that basis separation is critical for capturing distinct preferences. (c) Removing $\mathcal{L}_{\text{REG}}$ results in smaller but consistent degradation across all datasets. (d) Replacing the mixture-of-Gaussians (MoG) prior with $\mathcal{N}(\mathbf{0}, \mathbf{I})$ degrades performance on all datasets, highlighting the benefit of regularizing user distributions to lie within the learned preference space. (e) Replacing the product-of-Gaussians (PoG) with mean pooling degrades performance, showing that uncertainty-aware aggregation of user observations yields more reliable estimation. Since preference bases are integral to VRF and cannot be ablated, we analyze sensitivity to the number of bases (K), along with the regularization strength (β), in Appendix B.2.

Case Studies on Learned Preference Space. Figure 4a shows four users across two axes. In terms of data scarcity, both User A (n=19) and User B (n=3) exhibit consistent preferences, but the limited observations for User B result in higher uncertainty (σ_u=0.75). In terms of preference diversity, User C and User D have the same reference set size (n=10), yet User C spreads across five bases while User D concentrates on two (p_4, p_8), resulting in higher uncertainty for User C (σ_u=0.56).

Figure 4b shows how these users' preference distributions are represented in the learned preference space. User A exhibits a tight ellipse near p_7, reflecting high confidence from abundant, consistent data. User B, despite concentrating on p_4, shows a large ellipse due to data scarcity. User C spans multiple bases, capturing diverse preferences. User D sits



(a) User Preference in Reference Set (b) Learned Preference Space

Figure 4: Case studies and the learned preference space on PRISM (K=8). (a) Four users with varying data scarcity and preference diversity. n is the reference set size. Bars show the preference distribution classified by Claude Sonnet 4.6 (see Appendix D.1 for details). (b) Learned preference space. Stars and shaded regions denote the preference bases. For user ellipses, the center point represents $\mathbf{w}_u$, and the size reflects σ_u.

between p_4 and p_8 with moderate uncertainty. This visualization shows that VRF effectively disentangles what a user prefers from how confident the estimate is.

6 Conclusion

In this paper, we present VRF for uncertainty-aware LLM personalization, addressing two key challenges in reward factorization. For **C1**, VRF introduces shared preference bases, allowing data-scarce and unseen users to leverage collective preference patterns. For **C2**, VRF represents both users and bases as Gaussians for uncertainty-aware distributional matching, further enhanced by variance-attenuated training. Experiments on three benchmarks show consistent improvements over all baselines, robustness to scarcity and uncertainty, and gains in downstream alignment. We believe VRF provides an effective and scalable path for reliable LLM personalization. Future directions include query-adaptive personalization across diverse contexts and temporal adaptation to evolving user preferences.

Acknowledgments

This work used Delta at the National Center for Supercomputing Applications through allocation CIS260153 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program (Boerner et al., 2023), which is supported by U.S. National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296. Jiawei Han's research is supported by NSF-2505932, NSF IIS 25-37827, and NSF-2118329.

References

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Michiel Bakker, Martin Chadwick, Hannah Sheahan, Michael Tessler, Lucy Campbell-Gillingham, Jan Balaguer, Nat McAleese, Amelia Glaese, John Aslanides, Matt Botvinick, et al. Fine-tuning language models to find agreement among humans with diverse preferences. *Advances in neural information processing systems*, 35:38176–38189, 2022.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

Timothy J. Boerner, Stephen Deems, Thomas R. Furlani, Shelley L. Knuth, and John Towns. Access: Advancing innovation: Nsf's advanced cyberinfrastructure coordination ecosystem: Services & support. In *Practice and Experience in Advanced Research Computing (PEARC '23)*. ACM, 2023. doi: 10.1145/3569951.3597559.

Avinandan Bose, Zhihan Xiong, Yuejie Chi, Simon Shaolei Du, Lin Xiao, and Maryam Fazel. LoRe: Personalizing LLMs via low-rank reward modeling. *arXiv preprint arXiv:2504.14439*, 2025.

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.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

Hongru Cai, Yongqi Li, Tiezheng Yu, Fengbin Zhu, Wenjie Wang, Fuli Feng, and Wenjie Li. One adapts to any: Meta reward modeling for personalized LLM alignment. *arXiv preprint arXiv:2601.18731*, 2026.

Yuanpu Cao, Tianrong Zhang, Bochuan Cao, Ziyi Yin, Lu Lin, Fenglong Ma, and Jinghui Chen. Personalized steering of large language models: Versatile steering vectors through bi-directional preference optimization. *Advances in Neural Information Processing Systems*, 37:49519–49551, 2024.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando Ramirez, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *Transactions on Machine Learning Research*, 2023.

Daiwei Chen, Yi Chen, Aniket Rege, Zhi Wang, and Ramya Korlakai Vinayak. PAL: Sample-efficient personalized reward modeling for pluralistic alignment. In *International Conference on Learning Representations*, 2025.

Youngbin Choi, Seunghyuk Cho, Minjong Lee, MoonJeong Park, Yesong Ko, Jungseul Ok, and Dongwoo Kim. CoPL: Collaborative preference learning for personalizing LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025.

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

Sunhao Dai, Ninglu Shao, Haiyuan Zhao, Weijie Yu, Zihua Si, Chen Xu, Zhongxiang Sun, Xiao Zhang, and Jun Xu. Uncovering ChatGPT's capabilities in recommender systems. In *Proceedings of the 17th ACM conference on recommender systems*, pp. 1126–1132, 2023.

Sumanth Doddapaneni, Krishna Sayana, Ambarish Jash, Sukhdeep Sodhi, and Dima Kuzmin. User embedding model for personalized language prompting. In *Proceedings of the 1st Workshop on Personalization of Generative AI Systems (PERSONALIZE 2024)*, pp. 124–131, 2024.

Esin Durmus, Karina Nguyen, Thomas Liao, Nicholas Schiefer, Amanda Askell, Anton Bakhtin, Carol Chen, Zac Hatfield-Dodds, Danny Hernandez, Nicholas Joseph, et al. Towards measuring the representation of subjective global opinions in language models. In *First Conference on Language Modeling*, 2024.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 1321–1330, 2017.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. *ICLR*, 2022.

Gihoon Kim and Euntai Kim. Swap-guided preference learning for personalized reinforcement learning from human feedback. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=nc28mSbyVG.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6114.

Hannah Rose Kirk, Bertie Vidgen, Paul Röttger, and Scott A Hale. The benefits, risks and bounds of personalizing the alignment of large language models to individuals. *Nature Machine Intelligence*, 6(4):383–392, 2024a.

Hannah Rose Kirk, Alexander Whitefield, Paul Röttger, Andrew Bean, Katerina Margatina, Juan Ciro, Rafael Mosquera, Max Bartolo, Adina Williams, He He, Bertie Vidgen, and Scott A. Hale. The PRISM alignment project: What participatory, representative and individualised human feedback reveals about the alignment problem. In *Advances in Neural Information Processing Systems*, volume 37, 2024b.

Xinyu Li, Ruiyang Zhou, Zachary C. Lipton, and Liu Leqi. Personalized language modeling from personalized human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Zhiyu Li, Yanfang Chen, Xuan Zhang, and Xun Liang. BookGPT: A general framework for book recommendation empowered by large language model. *Electronics*, 12(22):4654, 2023.

Jiongnan Liu, Yutao Zhu, Shuting Wang, Xiaochi Wei, Erxue Min, Yu Lu, Shuaiqiang Wang, Dawei Yin, and Zhicheng Dou. LLMs+ persona-plug = personalized LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9373–9385, 2025.

David JC MacKay. The evidence framework applied to classification networks. *Neural computation*, 4(5):720–736, 1992.

Thomas P. Minka. Expectation propagation for approximate Bayesian inference. In *Uncertainty in Artificial Intelligence*, volume 17, pp. 362–369, 2001.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Sriyash Poddar, Yanming Wan, Hamish Ivison, Abhishek Gupta, and Natasha Jaques. Personalizing reinforcement learning from human feedback with variational preference learning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 52516–52544, 2024.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, 2024.

Chris Richardson, Yao Zhang, Kellen Gillespie, Sudipta Kar, Arshdeep Singh, Zeynab Raeesy, Omar Zia Khan, and Abhinav Sethy. Integrating summarization and retrieval for enhanced personalization via large language models. *arXiv preprint arXiv:2310.20081*, 2023.

Michael J. Ryan, Omar Shaikh, Aditri Bhagirath, Daniel Frees, William Held, and Diyi Yang. SynthesizeMe! inducing persona-guided prompts for personalized reward models in LLMs. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8045–8078, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.397. URL https://aclanthology.org/2025.acl-long.397/.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. LaMP: When large language models meet personalization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7370–7392, 2024.

Jingyan Shen, Jiarui Yao, Rui Yang, Yifan Sun, Feng Luo, Rui Pan, Tong Zhang, and Han Zhao. MICRO: Mixture modeling and context-aware routing for personalized preference learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 17458–17474, 2025.

Idan Shenfeld, Felix Faltings, Pulkit Agrawal, and Aldo Pacchiano. Language model personalization via reward factorization. *arXiv preprint arXiv:2503.06358*, 2025.

Anand Siththaranjan, Cassidy Laidlaw, and Dylan Hadfield-Menell. Distributional preference learning: Understanding and accounting for hidden context in RLHF. In *Proceedings of the Twelfth International Conference on Learning Representations*, 2024.

Taylor Sorensen, Jared Moore, Jillian Fisher, Mitchell L Gordon, Niloofar Mireshghallah, Christopher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, et al. Position: A roadmap to pluralistic alignment. In *International Conference on Machine Learning*, pp. 46280–46302. PMLR, 2024.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

Zhaoxuan Tan, Zheyuan Liu, and Meng Jiang. Personalized pieces: Efficient personalized large language models through collaborative efforts. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 6459–6475, 2024a.

Zhaoxuan Tan, Qingkai Zeng, Yijun Tian, Zheyuan Liu, Bing Yin, and Meng Jiang. Democratizing large language models via personalized parameter-efficient fine-tuning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 6476–6491, 2024b.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Cédric Villani. *Optimal Transport: Old and New*. Springer, 2008.

Jinghao Zhang, Yuting Liu, Wenjie Wang, Qiang Liu, Shu Wu, Liang Wang, and Tat-Seng Chua. Personalized text generation with contrastive activation steering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7128–7141, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. *Advances in neural information processing systems*, 36: 46595–46623, 2023.

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.

Thomas P. Zollo, Andrew Wei Tung Siah, Naimeng Ye, Ang Li, and Hongseok Namkoong. PersonalLLM: Tailoring LLMs to individual preferences. *arXiv preprint arXiv:2409.20296*, 2024.

Table 4: Average dataset statistics per user. For seen users, C_u is sampled from training data per epoch, and the full training set serves as C_u at evaluation. For unseen users, only C_u is provided at inference to estimate user weights. Both settings are evaluated on held-out D_u.

Dataset	Seen					Unseen		
	#Users	Training		Evaluation		#Users	Evaluation	
		$\lvert\mathcal{C}_u\rvert$	$\lvert\mathcal{D}_u\rvert$	$\lvert\mathcal{C}_u\rvert$	$\lvert\mathcal{D}_u\rvert$		$\lvert\mathcal{C}_u\rvert$	$\lvert\mathcal{D}_u\rvert$
PersonalLLM	1,000	9.0	36.0	45.0	45.0	1,000	9.0	9.0
TL;DR (n=100)	20	50.0	50.0	100.0	1,449.8	20	50.0	1,146.4
TL;DR (n=150)	20	50.0	100.0	150.0	1,449.8	20	50.0	1,146.4
PRISM	623	3.0	6.1	9.1	8.6	623	9.6	9.1

A Experimental Setup

A.1 Datasets

Following state-of-the-art reward factorization work (Chen et al., 2025; Bose et al., 2025; Shenfeld et al., 2025), we evaluate on three benchmarks spanning synthetic and real-world personalized preference data. Detailed statistics are provided in Table 4.

- **PersonalLLM** (Zollo et al., 2024) provides 9,402 prompts, each with 8 candidate responses scored by 10 reward models (i.e., $r \in \mathbb{R}^{10}$). To construct personalized preference pairs, we synthesize a user preference vector $\mathbf{w} \sim \text{Dirichlet}(\alpha) \in \mathbb{R}^{10}$ and compute a composite score $s = \mathbf{w}^\top \mathbf{r}$. For each prompt, we form a preference pair from the highest- and lowest-scoring responses. We synthesize 2,000 users, equally split into seen and unseen. We evaluate across three diversity levels, $\alpha \in \{0.001, 0.01, 0.1\}$, where a smaller α produces more heterogeneous preferences, making personalization more critical.
- **TL;DR** (Stiennon et al., 2020) contains pairwise human preference data over Reddit post summaries, organized by individual annotators. We retain annotators with at least 50 annotations and randomly select 40, partitioning them into 20 seen and 20 unseen users. Seen users have $n \in \{100, 150\}$ triplets for training.
- **PRISM** (Kirk et al., 2024b) collects multi-turn conversations between crowdworkers and LLMs, where annotators score multiple model responses per interaction. We use the demographically balanced subset (623 seen / 623 unseen users) and construct pairwise preference data from each interaction using the highest- and lowest-scored responses.

A.2 Baselines

We compare against state-of-the-art personalized reward modeling methods. To ensure a fair comparison, baselines include $\mathcal{C}_u$ in the training set for seen users.

- **Ref** (Qwen Team, 2025) is a reference baseline that uses the pretrained LLM's per-token log-probability as a reward signal (i.e., $r(x, y) = \frac{1}{|y|} \sum_i \log P(y_i \mid x, y_{<i})$), without any training or personalization.
- **BT** (Stiennon et al., 2020; Ouyang et al., 2022) is a standard Bradley–Terry (Bradley & Terry, 1952) reward model fine-tuned on all users' preference data jointly, without any user-specific personalization.
- **VPL** (Poddar et al., 2024) encodes each user's preference history into a variational latent variable, which conditions the reward prediction.
- **PAL** (Chen et al., 2025) represents each user as an ideal point in a shared preference latent space to compute personalized rewards.
- **LoRe** (Bose et al., 2025) models individual user preferences as weighted combinations of shared basis reward functions.
- **PReF** (Shenfeld et al., 2025) represents user-specific rewards as a linear combination of basis reward functions with SVD initialization and L2 regularization.

Table 5: Best hyperparameter configurations for VRF across all datasets.

Dataset	Epochs	Batch size	Learning rate	Dropout	Weight decay	K
PersonalLLM (α=0.001)	5	8	2e-5	0.0	0.01	8
PersonalLLM (α=0.01)	5	8	2e-5	0.0	0.001	8
PersonalLLM (α=0.1)	5	8	2e-5	0.0	0.01	8
TL;DR (n=100)	3	8	6e-5	0.1	0.001	8
TL;DR (n=150)	3	8	6e-5	0.1	0.001	8
PRISM	5	8	6e-5	0.0	0.01	12

Table 6: Effect of backbone scale on accuracy and efficiency for the PRISM dataset.

Backbone	Seen	Unseen	Overall	Training (min)	Inference (min)
0.6B	61.31±0.57	60.58±0.53	60.94±0.18	**80.8**	**20.1**
1.7B	61.28±0.32	60.44±0.27	60.86±0.21	131.4	33.6
4B	61.72±0.51	**61.31±0.49**	61.51±0.43	288.1	74.6
8B	**62.14±0.46**	61.15±0.46	**61.64±0.32**	446.8	117.3

A.3 Implementation Details

All experiments are conducted on NVIDIA A40 GPUs and AMD EPYC 7763 CPUs. We optimize with AdamW and cosine learning rate scheduling. We fix the latent dimension D=64, and regularization strength β=0.001 across all datasets, as performance is insensitive to these values. We perform grid search over learning rate $\in \{2e{-}5, 4e{-}5, 6e{-}5\}$, dropout $\in \{0.0, 0.1\}$, weight decay $\in \{0.001, 0.01\}$, $K \in \{4, 8, 12, 16, 20\}$, temperature hyperparameters $\tau_d, \tau_m \in \{0.1, 1.0\}$. The best hyperparameters per dataset are reported in Table 5.

A.4 Inference-Time Alignment Setup

Figure 3c shows the inference-time alignment results via best-of-N sampling on PRISM unseen users. For each user's test query, we generate N=8 candidate responses using GPT-5 mini as the policy model, each conditioned on a distinct style inferred from the preference basis label (see Figure 4). Following the LLM-as-a-judge framework (Zheng et al., 2023), GPT-5.1 evaluates each method's selected response against BT's in a pairwise manner, with the user's profile (demographics, preferences, and values) as context and randomized response ordering to mitigate position bias. We report the Δ Win Rate, the win rate over BT normalized by the global minimum. Full prompt templates are provided in Appendix D.2.

B Additional Experiments

B.1 Backbone Scaling

Table 6 examines the effect of backbone scale by varying Qwen3 from 0.6B to 8B parameters with LoRA (Hu et al., 2022) fine-tuning (r=16, α=32) under identical hyperparameters. While the 8B backbone achieves the best overall accuracy, the improvement over 0.6B is marginal (+0.7%) relative to its cost (5.5× training and 5.8× inference time). This accuracy–efficiency trade-off suggests that smaller backbones are a practical choice for VRF, as shared preference bases and uncertainty-aware distributional matching compensate for reduced backbone capacity.

B.2 Hyperparameter Sensitivity

Number of preference bases (K). Figure 5 (left) shows the effect of varying $K \in \{4, 8, 12, 16, 20\}$ on three datasets. Performance peaks at K=8 on PersonalLLM and TL;DR and at K=12 on PRISM, with marginal declines beyond. These results suggest that too many bases may introduce redundancy or optimization difficulty. In other words, a small number of shared preference bases suffices to capture population-level preference diversity, making VRF scalable as K remains fixed regardless of user population size.

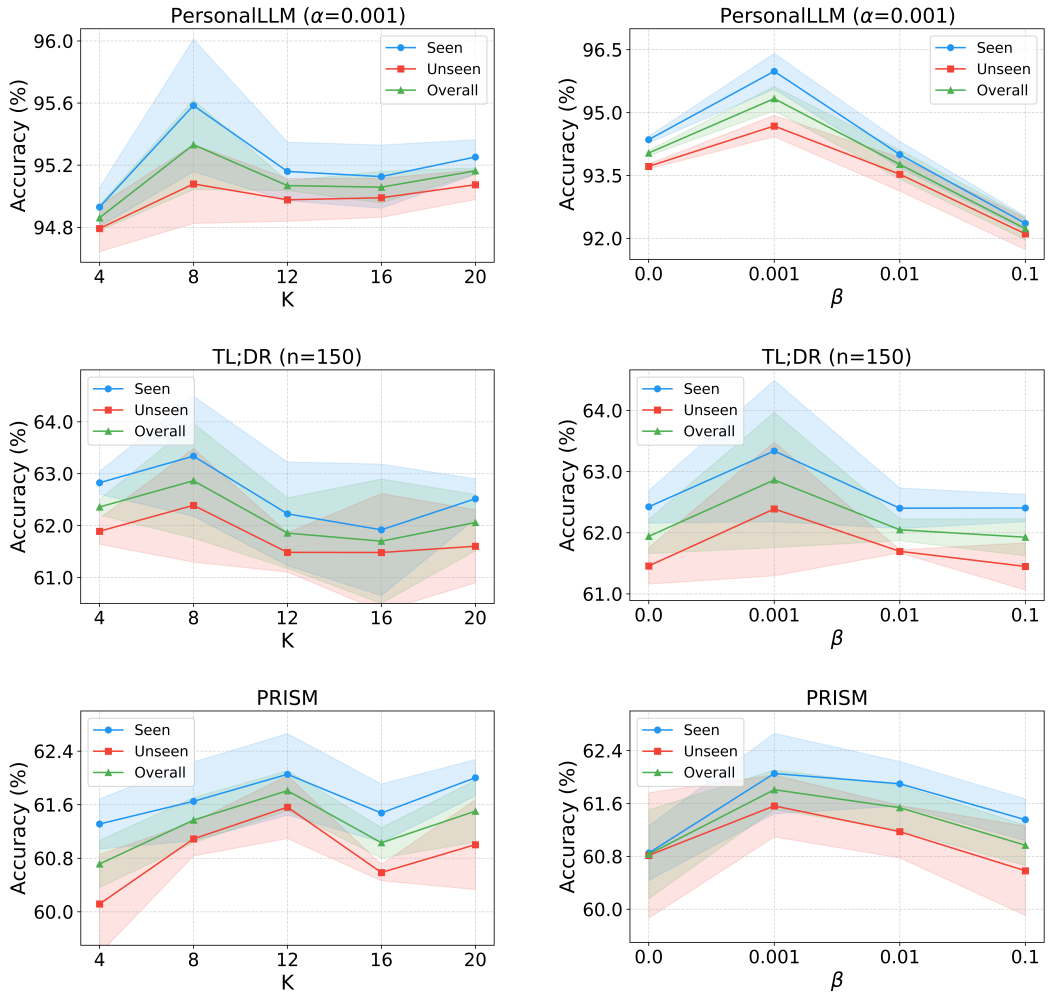


Figure 5: Hyperparameter sensitivity analysis on three datasets.

Regularization strength (β). Figure 5 (right) shows the effect of varying $\beta \in \{0.0, 0.001, 0.01, 0.1\}$. Across all three datasets, performance peaks at β=0.001 and degrades at both extremes. Without regularization (β=0.0), user distributions lack structural guidance from the mixture-of-Gaussians prior, scattering arbitrarily in the latent space. Conversely, excessive regularization (β=0.1) over-concentrates user distributions toward the prior, potentially suppressing user-specific information. We adopt β=0.001 as the default.

B.3 Few-shot, Uncertainty and Efficiency Analysis on Additional Datasets

Figure 6 extends the few-shot adaptation, uncertainty robustness and adaptation efficiency analysis of Figure 3 to PersonalLLM (α=0.001) and TL;DR (n=150). VRF achieves the highest accuracy across few-shot settings and all uncertainty groups while maintaining near-zero adaptation overhead, confirming that the findings on PRISM generalize across datasets. Notably, in few-shot adaptation (Figure 6a), LoRe and PReF show increasing accuracy as $|\mathcal{C}_u|$ grows, whereas VRF achieves the highest accuracy even with $|\mathcal{C}_u|$=1, highlighting that the variational user encoder effectively infers the preferences of unseen users.

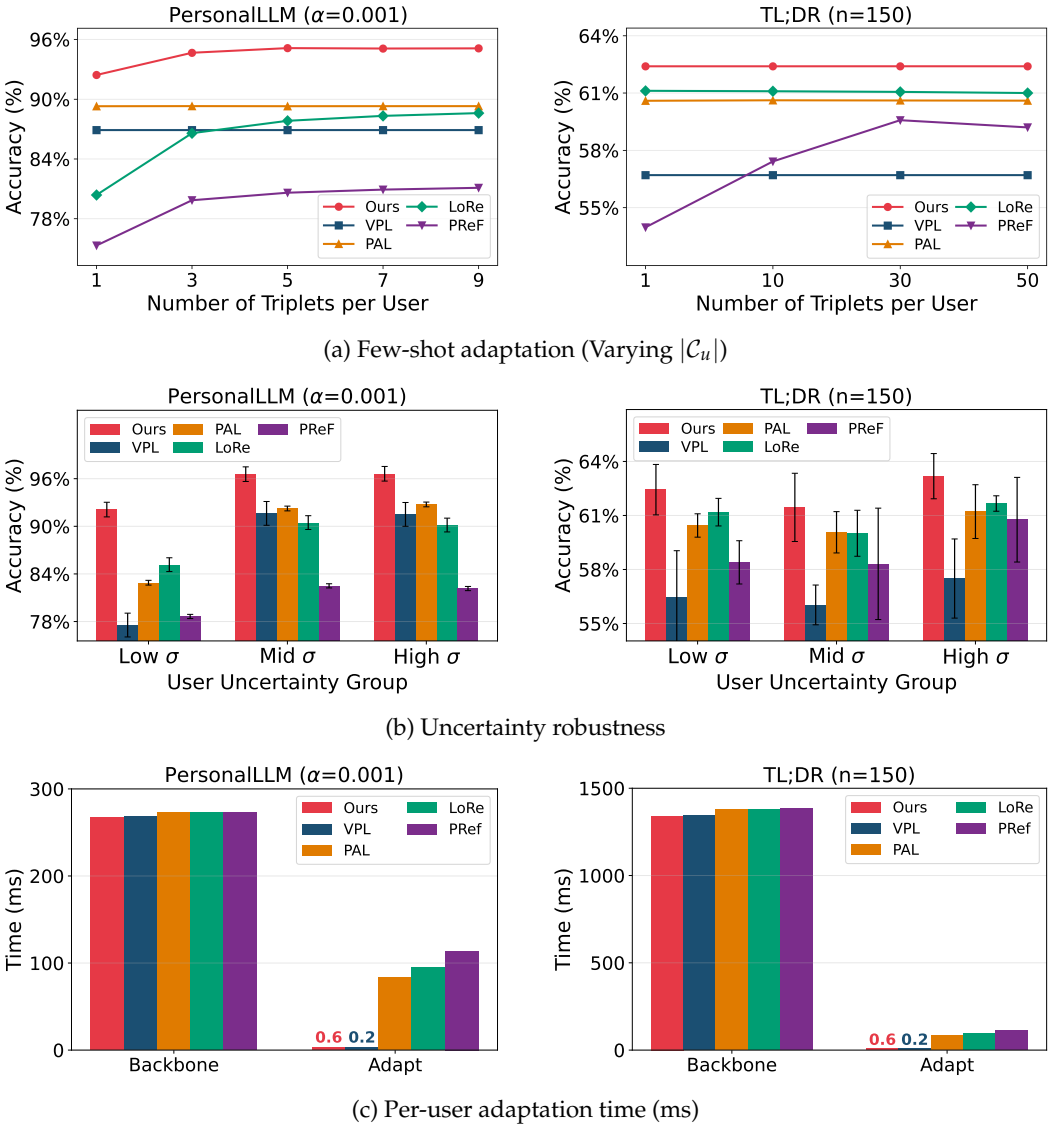


Figure 6: In-depth comparison on PersonalLLM (α=0.001) and TL;DR (n=150) unseen users.

B.4 Coverage and Generalization of Preference Bases

Figure 7 visualizes the distribution of dominant preference bases across seen and unseen users on PRISM, derived from the learned preference space in Figure 4. Two observations emerge: (1) users are spread across all K preference bases, suggesting that the bases effectively capture diverse user preferences. (2) The two distributions exhibit similar trends, indicating that the learned preference bases generalize to unseen users.

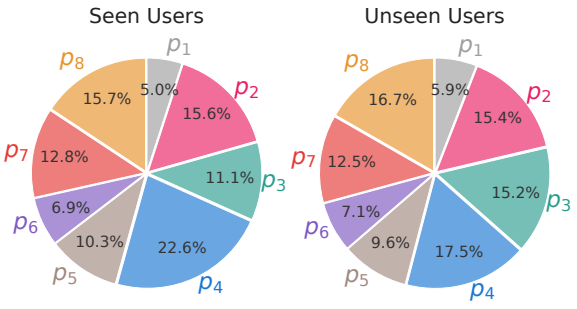


Figure 7: Distribution of dominant preference bases for seen and unseen users on PRISM.

C Proofs and Derivations

C.1 Concavity of σ_Δ^2 with Respect to $\mathbf{w}_u$

Proposition 1. *$\sigma_\Delta^2 = \sum_{k=1}^{K} w_{u,k}(\Delta\phi_k - \mu_\Delta)^2$ is concave with respect to $\mathbf{w}_u$.*

Proof. The Hessian of σ_Δ^2 with respect to $\mathbf{w}_u$ is:

$$\nabla^2_{\mathbf{w}_u}\sigma_\Delta^2 = -2\Delta\boldsymbol{\phi}\Delta\boldsymbol{\phi}^\top. \tag{10}$$

For any vector $\mathbf{v} \in \mathbb{R}^K$,

$$\mathbf{v}^\top \nabla^2_{\mathbf{w}_u}\sigma_\Delta^2\,\mathbf{v} = -2\mathbf{v}^\top(\Delta\boldsymbol{\phi}\Delta\boldsymbol{\phi}^\top)\mathbf{v} = -2(\mathbf{v}^\top\Delta\boldsymbol{\phi})^2 \leq 0, \tag{11}$$

so the Hessian is negative semi-definite and σ_Δ^2 is concave. At the vertices of the simplex (one-hot $\mathbf{w}_u$), σ_Δ^2 reduces to zero and increases as $\mathbf{w}_u$ becomes more diffuse. □

C.2 Proof of Gradient Attenuation in $\mathcal{L}_{\text{VBT}}$

Proposition 2. *As $\sigma_\Delta^2 \to \infty$, the gradients of $\mathcal{L}_{\text{VBT}}$ with respect to all parameters ϕ_k and $w_{u,k}$ vanish, regardless of the sign of μ_Δ.*

Proof. Let $s = \sigma_\Delta^2$, $c = \frac{\pi}{8}$, and $z = \frac{\mu_\Delta}{\sqrt{1+cs}}$. We first compute the partial derivatives of $\mathcal{L}_{\text{VBT}} = -\log\sigma(z)$ with respect to the intermediate variables μ_Δ and s:

$$\frac{\partial\mathcal{L}_{\text{VBT}}}{\partial\mu_\Delta} = -(1-\sigma(z))\cdot\frac{1}{\sqrt{1+cs}}, \tag{12}$$

$$\frac{\partial\mathcal{L}_{\text{VBT}}}{\partial s} = \frac{\mu_\Delta c(1-\sigma(z))}{2(1+cs)^{3/2}}. \tag{13}$$

As $s \to \infty$, $\frac{1}{\sqrt{1+cs}} \to 0$ and $\frac{1}{(1+cs)^{3/2}} \to 0$. Since $(1-\sigma(z))$ is bounded by $[0,1]$, both $\frac{\partial L_{VBT}}{\partial\mu_\Delta}$ and $\frac{\partial L_{VBT}}{\partial s}$ converge to zero.

Gradient with respect to ϕ_k. By chain rule:

$$\frac{\partial\mathcal{L}_{\text{VBT}}}{\partial\phi_k} = \frac{\partial\mathcal{L}_{\text{VBT}}}{\partial\mu_\Delta}\cdot w_{u,k} + \frac{\partial\mathcal{L}_{\text{VBT}}}{\partial s}\cdot\frac{\partial s}{\partial\phi_k}. \tag{14}$$

Gradient with respect to $w_{u,k}$. By chain rule:

$$\frac{\partial\mathcal{L}_{\text{VBT}}}{\partial w_{u,k}} = \frac{\partial\mathcal{L}_{\text{VBT}}}{\partial\mu_\Delta}\cdot\Delta\phi_k + \frac{\partial\mathcal{L}_{\text{VBT}}}{\partial s}\cdot(\Delta\phi_k - \mu_\Delta)^2. \tag{15}$$

Since all terms in Eqs. (14) and (15) involve the vanishing partial derivatives $\frac{\partial L_{VBT}}{\partial\mu_\Delta}$ and $\frac{\partial L_{VBT}}{\partial s}$ multiplied by bounded coefficients, it follows that $\frac{\partial L_{VBT}}{\partial\phi_k} \to 0$ and $\frac{\partial L_{VBT}}{\partial w_{u,k}} \to 0$ as $s \to \infty$. □

C.3 Monte Carlo Estimation of $\mathcal{L}_{\text{REG}}$

Since $D_{KL}(q_u \| p(z))$ against a MoG prior is intractable in closed form, we decompose it as:

$$D_{KL}(q_u\|p(z)) = -\mathcal{H}(q_u) - \mathbb{E}_{q_u}[\log p(z)], \tag{16}$$

where $\mathcal{H}(q_u)$ is the entropy of q_u. Each term is then estimated separately:

Entropy term. The entropy of a diagonal Gaussian $q_u = \mathcal{N}(\boldsymbol{\mu}_u, \text{diag}(\boldsymbol{\sigma}_u^2))$ is given by:

$$\mathcal{H}(q_u) = \frac{1}{2} \sum_{d=1}^{D} \left(1 + \log(2\pi) + \log \sigma_{u,d}^2\right). \tag{17}$$

Cross-entropy term. $\mathbb{E}_{q_u}[\log p(z)]$ is estimated via the reparameterization trick (Kingma & Welling, 2014) with S=5 Monte Carlo samples:

$$\mathbb{E}_{q_u}[\log p(z)] \approx \frac{1}{S} \sum_{s=1}^{S} \log p(z^{(s)}), \quad z^{(s)} = \boldsymbol{\mu}_u + \boldsymbol{\sigma}_u \odot \boldsymbol{\epsilon}^{(s)}, \quad \boldsymbol{\epsilon}^{(s)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{18}$$

where $z^{(s)}$ is a reparameterized sample that allows gradients to flow back to $\boldsymbol{\mu}_u$ and $\boldsymbol{\sigma}_u$. The MoG log-probability is computed as:

$$\log p(z) = \log \frac{1}{K} \sum_{k=1}^{K} \mathcal{N}(z; \boldsymbol{\mu}_k, \text{diag}(\boldsymbol{\sigma}_k^2)) = \text{logsumexp}_k \left(\log \mathcal{N}(z; \boldsymbol{\mu}_k, \boldsymbol{\sigma}_k^2)\right) - \log K, \tag{19}$$

with logsumexp applied for numerical stability. Thus, combining both terms yields a differentiable estimate of $D_{KL}(q_u \parallel p(z))$ that can be optimized end-to-end via backpropagation.

D Prompts

D.1 Preference Space Analysis

To interpret the learned preference space in Figure 4, we label each preference basis and classify user triplets using Claude Sonnet 4.6.

Preference Basis Labeling. For each preference basis, we sample 10 triplets from the top-5 nearest users and use the following prompt:

> The following are 10 triplets from users with similar preferences.
> Describe what drives their preference for the chosen over the rejected responses.
>
> [1] (x, y^+, y^-)
> ...
> [10] (x, y^+, y^-)

The resulting labels (e.g., *conversational & natural*, *direct & structured*) are reported in Figure 4.

Triplet Classification. Using the K labels, each triplet is classified by prompting:

> Given a preference triplet (x, y^+, y^-), which of the following preference basis labels best explains the user's choice?
> [1] Conversational & Natural
> [2] Thorough & Genuine
> ...
> [8] Empathetic & Principled
> Answer with only the number.

D.2 Inference-Time Alignment

In Figure 3c, we evaluate whether the learned personalized rewards can guide response selection at inference time via best-of-N sampling.

Response Generation. We generate N=8 candidate responses using GPT-5 mini as a policy model, each conditioned on a distinct preference basis label as a response style.

> You are a helpful AI assistant. Generate a {*style*} response to the following query: {*query*}

LLM-as-a-Judge. We employ GPT-5.1 as an evaluator, comparing each method's selected response against BT's, with the user's profile (e.g., demographics, LLM preferences, personal values) to guide evaluation. Response order is randomized to mitigate position bias.

> You are an impartial judge. Given the user's profile, the query, and the two responses, which is more appropriate for this user?
>
> [User Profile] {*user_profile*}
>
> [Query] {*query*}
>
> [Response A] {*response_a*}
>
> [Response B] {*response_b*}
>
> Reply with ONLY "A" or "B".