Direct Preference Optimization with Unobserved Preference Heterogeneity: The Necessity of Ternary Preferences

Keertana Chidambaram **Karthik Vinary Seetharaman** **Vasilis Syrgkanis**
Stanford University

Abstract

Reinforcement Learning from Human Feedback (RLHF) has become central to aligning large language models with human values, typically by first learning a reward model from preference data which is then used to update the model with reinforcement learning. Recent alternatives such as Direct Preference Optimization (DPO) simplify this pipeline by directly optimizing on preferences. However, both approaches often assume uniform annotator preferences and rely on binary comparisons, overlooking two key limitations: the diversity of human evaluators and the limitations of pairwise feedback. In this work, we address both these issues. First, we connect preference learning in RLHF with the econometrics literature and show that binary comparisons are insufficient for identifying latent user preferences from finite user data and infinite users, while (even incomplete) rankings over three or more responses ensure identifiability. Second, we introduce methods to incorporate heterogeneous preferences into alignment algorithms. We develop an Expectation-Maximization adaptation of DPO that discovers latent annotator types and trains a mixture of LLMs accordingly. Then we propose an aggregation algorithm using a min-max regret fairness criterion to produce a single generative policy with equitable performance guarantees. Together, these contributions establish a theoretical and algorithmic framework for fairness and personalization for diverse users in generative model alignment.

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

1 INTRODUCTION

Reinforcement Learning from Human Feedback (RLHF) has emerged as a leading approach to align language models (LMs) with human preferences by learning a single reward model from preference data and using it to fine-tune the LM (Ouyang et al. (2022); Stiennon et al. (2020); Wang et al. (2023b)). Direct Preference Optimization (DPO) (Rafailov et al. (2023)) sidesteps the reinforcement learning step by directly optimizing policies from preference comparisons, but like RLHF, it implicitly assumes a single reward model and, hence, homogeneous preferences in the target population. Therefore, these methods run the risk of aligning only to the preferences of majority groups and neglecting underrepresented groups with heterogeneous preferences, leading to suboptimal behavior for several parts of the population and to bias and discrimination.

Existing attempts to address this issue typically learn more expressive reward models and then perform RL (e.g., with PPO), but DPO offers advantages in stability and simplicity by avoiding explicit reward modeling. For instance, (Zhou et al. (2023)) builds on DPO to implicitly learn a multi-objective reward model, and other work generalizes to multi-dimensional rewards (Wang et al. (2024); Zhou et al. (2023)), yet such approaches face two challenges: (i) they often require annotators to provide multi-dimensional ratings (e.g., safety, accuracy), which are more costly and harder to obtain than binary preference data (Casper et al. (2023)); and (ii) the objectives must be fixed before data collection, which is problematic since many latent cultural, political, or geographical factors shape preferences in ways that are difficult to anticipate (Siththaranjan et al. (2023); Bai et al. (2022)).

The main contributions of this paper are summarized as follows:

- We introduce Expectation-Maximization Direct Preference Optimization (**EM-DPO**), a novel clustering algorithm using expectation–maximization that simultaneously uncovers latent user preference types and trains an ensemble of LLMs, where each element in the ensemble is tailored to each type directly using preference data. By discovering hidden patterns in user preferences and learning separate models for distinct preference groups, EM-DPO enables genuine personalization of LLMs to diverse user populations.

- We propose MinMax Regret Aggregation (**MMRA**), an aggregation algorithm that combines the LLM ensembles learned with EM-DPO based on a min–max regret fairness criterion. This enables robust deployment for pluralistic alignment when individual user types are unknown at inference time, ensuring no preference group is severely underserved.
- We establish a fundamental connection between preference learning in LLMs and the econometrics literature, revealing when identification of heterogeneous preferences is provably achievable under a simple linear reward model. We uncover that latent heterogeneous preferences are *not identifiable* when each user provides a binary comparison, but they *become identifiable* when each user reports a single data point of their preferred response among three options. This identification result has profound implications for LLM personalization, as it demonstrates that the standard binary comparison paradigm is fundamentally insufficient for learning diverse user preferences, regardless of dataset size. The solution is simple and surprising; ask the user to choose among three options, instead of just two. We also validate this theory with empirical evidence proving that ternary preferences exhibit stronger performance than binary preferences.

The research contributions closest to our work are Chakraborty et al. (2024) and Park et al. (2024). Our approach offers several key advantages: First, EM-DPO directly performs expectation-maximization, providing better theoretical guarantees than the hard-clustering algorithms of Chakraborty et al. (2024) and Park et al. (2024). Hard clustering forces discrete group membership, discarding the natural overlap in user preferences—users rarely belong cleanly to a single group. EM-based soft clustering captures this more naturally through continuous weights, offering better training stability and stronger theoretical guarantees Kearns et al. (1998).

Second, motivated by our identification results demonstrating that multi-item preferences are necessary for learning heterogeneous user types, EM-DPO is specifically designed to handle multi-item comparisons in addition to binary preferences, a capability absent in prior work. Third, both our algorithms are reward-free similar to Park et al. (2024), avoiding the need for explicit reward modeling. Finally, our fairness criterion based on min-max regret differs fundamentally from Chakraborty et al. (2024), which uses max-min rewards. We next discuss further related work on RLHF with preference heterogeneity and defer further comparisons to related work on reward modeling and preference-based reinforcement learning to the Appendix.

RLHF With Diverse Preferences. Diversity in annotator preferences has been recognized as a chief issue in RLHF Dumoulin et al. (2023). Several studies have tried to solve the diverse population problem by learning more expressive reward functions and then using them to perform RLHF. For example, Rame et al. (2024); Jang et al. (2023); Chakraborty et al. (2024) maintains and learns several reward models at once. Similarly, Wang et al. (2024) learns a multi-dimensional reward model where each dimension provides rewards based on a different objective such as safety or usefulness. Yang et al. (2024a) proposes a policy-agnostic method to perform multi-objective LLM alignment. Alternatively, Siththaranjan et al. (2023); Li et al. (2024) learns a distribution over fixed reward models. Finally, these reward models are combined using various strategies Bakker et al. (2022); Jang et al. (2023); Rame et al. (2024) to get a final reward model which is then used to perform RLHF. Chakraborty et al. (2024) also learns multiple reward models, but performs RL by maximizing the minimum reward thereby ensuring that the final model is fair. The paper draws on elements of social choice theory, which Conitzer et al. (2024) argues is an effective path forward for RLHF research in general, specifically regarding issues with aggregating preferences. Dai and Fleisig (2024) outlines a correspondence between the key principles and desiderata of social choice into the RLHF context. In an orthogonal approach, Zhong et al. (2024) utilizes meta-learning to learn diverse preferences. In general, trying to do RLHF with many reward models becomes expensive, making extending DPO Rafailov et al. (2023) an attractive alternative. Swamy et al. (2024) proposes SPO to sidestep reinforcement learning using the concept of a minimax winner from social choice theory, but only in the case of homogeneous preferences. Ramé et al. (2024) also deals with the idea of aggregating reward models to increase robustness. We instead propose a complete pipeline to learn one equitable policy for a heterogeneous population without appealing to reward model estimation at all.

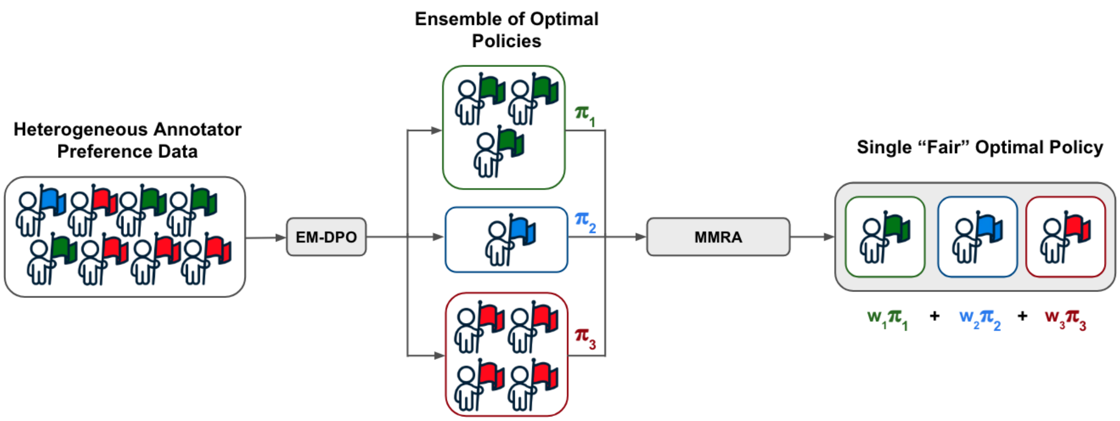


Figure 1: Proposed pipeline for learning an equitable policy. Step 1: Collect binary preferences from heterogeneous annotators. Step 2: Use EM-DPO to cluster annotators and derive an ensemble of optimal policies. Step 3: Apply MMRA to combine these policies into a single fair policy.

2 BACKGROUND

The RLHF (Ziegler et al. (2019); Stiennon et al. (2020); Ouyang et al. (2022)) pipeline has two main inputs. The first is a language model, denoted as π_{SFT}, which is pre-trained on large-scale data and then fine-tuned using supervised learning. The second input is a static annotator preference dataset, $\mathcal{D} = \{x, y_w, y_l, h\}$, collected as follows: for a given prompt x, pairs of responses (y_1, y_2) are generated from $\pi_{\text{SFT}}(\cdot|x)$, and a human annotator $h \in \mathcal{H}$ selects the preferred response. The winning and losing responses are denoted y_w and y_l, respectively.

To model the ground truth of annotator choices, a common assumption links the observed preferences to a latent reward function via the Bradley–Terry–Luce model (Bradley and Terry (1952); Ouyang et al. (2022); Rafailov et al. (2023)). Let $r^*(x, y)$ denote the true reward function. Then, for any pair (y_1, y_2):

$$p_{r^*}(y_1 \succ y_2|x) = \frac{\exp(r^*(x, y_1))}{\exp(r^*(x, y_1)) + \exp(r^*(x, y_2))} = \sigma(r^*(x, y_1) - r^*(x, y_2)) \quad (1)$$

Where σ represents the sigmoid function. The first step of RLHF fits a reward model $r_\psi(x, y)$ to approximate r^* by minimizing the log-likelihood over the observed preferences:

$$\mathcal{L}(r_\psi; \mathcal{D}) = -\mathbb{E}_{\mathcal{D}}[\sigma(r_\psi(x, y_1) - r_\psi(x, y_2))]$$

The second step fine-tunes the language model using reinforcement learning (PPO Schulman et al. (2017)) to maximize expected reward while regularizing deviation from the supervised model:

$$\pi^*_\phi = \arg\max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(y|x)}\left[r_\psi(y, x)\right] - \beta\mathbb{D}_{\text{KL}}[\pi_\phi(y|x) \,\|\, \pi_{\text{SFT}}(y|x)] \quad (2)$$

Direct Preference Optimization (DPO) (Rafailov et al. (2023)) bypasses the intermediate reward model by directly optimizing the policy using the preference dataset. Combining the Bradley–Terry model with the KL-regularized objective, DPO minimizes:

$$\mathcal{L}(\pi_\phi; \pi_{\text{SFT}}, \mathcal{D}, \beta) = -\mathbb{E}_{\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\phi(y_w|x)}{\pi_{\text{SFT}}(y_w|x)} - \beta\log\frac{\pi_\phi(y_l|x)}{\pi_{\text{SFT}}(y_l|x)}\right)\right]$$

$$\pi^*_\phi = \arg\min_{\pi_\phi} \mathcal{L}(\pi_\phi; \pi_{\text{SFT}}, \mathcal{D}, \beta)$$

This framework forms the foundation for our extension to diverse annotators, which we address in the following sections.

3 DPO FOR DIVERSE ANNOTATORS

Both Reinforcement Learning from Human Feedback (RLHF) and Direct Preference Optimization (DPO) assume uniform preferences across the population and learn a single reward model $(r_\psi(y, x))$ either implicitly or explicitly. However, human preferences and values are inherently diverse. For example, when asked "give me feedback on my writing", a majority of annotators may prefer direct, blunt critique, while a minority might prefer indirect, softened feedback that is more constructive and culturally appropriate. RLHF and DPO tend to align with the majority opinion among annotators, introducing bias and potentially marginalizing minority perspectives. To mitigate this issue, we propose a pipeline consisting of two algorithms: Expectation-Maximization DPO (EM-DPO) for soft-clustering diverse preference distributions and learning the optimally aligned policy for each cluster and Min-Max Regret Aggregation (MMRA), which fairly aggregates the learned policies to minimize worst-case regret for any sub-group of annotators.

We do not impose structural assumptions on how preference data varies across annotators, instead modeling this variation as arising from an unobserved latent factor. We assume only that the latent factors are discrete and finite, without access to any observable indicators such as group labels for annotators. To handle this, we extend the expectation-maximization (EM) algorithm (Dempster et al. (1977); Moon (1996)) to our setting, leveraging its ability to handle mixture data. The resulting algorithm, EM-DPO, soft-clusters annotators based on their observed preference data and learns an optimal policy for each cluster. We first look at the data generating process under this setting and then derive the EM algorithm.

Data Generating Process Let Z represent an annotator's latent factor capturing unobserved heterogeneity. We make the following assumption that there are a finite number of latent variables similar to (Chakraborty et al. (2024); Ramesh et al. (2024)):

Assumption 1 (Finite Latent Types)**.** For all Z, $Z \in \{z_1, \ldots, z_K\}$, where $K \in \mathbb{N}$ is some finite value.

Note that the value of k need not be fixed a priori and can be treated as a hyperparameter (see Appendix for hyper-parameter tuning). The true reward function of an annotator is thus $r^*(y, x; Z)$, where x and y denote the prompt and the response respectively. Suppose there are n annotators, i.e. $|\mathcal{H}| = n$, and for simplicity assume m preference data points per annotator. For each annotator h indexed by $i \in [n]$, the preference data is generated by first sampling a latent factor Z_i followed by m observed preferences with $V_{i,j} = (x^{i,j}, y_w^{i,j}, Y_r^{i,j})$ for $1 \leq j \leq m$ conditioned on Z_i, where $y_w^{i,j}$ is the preferred response and $Y_r^{i,j}$ is the set of rejected response(s).

In the context of LLMs, prompts X_{ij} are randomly assigned to annotators, ensuring no correlation between an annotator's preference type and the assigned prompt. Thus, prompts are equally likely across preference types. We formalize this observation in the following assumption:

Assumption 2 (Un-correlated Contexts and Latent Preference Types)**.** For all $k, \ell \in [K]$:

$$\begin{aligned} p(X_{ij} \mid Z_i = z_k; \theta) = &\; p(X_{ij} \mid Z_i = z_\ell; \theta) \\ := &\; \rho(X_{ij}) \end{aligned}$$

3.1 EM Algorithm

We start off with an offline dataset of annotator-level preferences $\mathcal{D}$ and k policies (LLMs). Let θ be the set of parameters that parametrize both the distribution of Z and the distribution of V conditioned on Z, i.e. $p(Z; \theta)$ and $p(V \mid Z; \theta)$. At time step t of the algorithm, let θ_t be the candidate parameters of the algorithm. Now the EM algorithm can be succinctly written as:

$$\theta_{t+1} = \arg\max_\theta Q(\theta \mid \theta_t)$$
$$\text{where, } Q(\theta \mid \theta_t) = \mathbb{E}_{Z \sim p(\cdot \mid V, \theta_t)} \left[\log(p(V, Z \mid \theta)) \right]$$

Breaking this down, there are two-steps, the E-step which is computing $Q(\theta \mid \theta_t)$ and the M-step which is finding the optimal value $\arg\max_\theta Q(\theta \mid \theta_t)$. We defer to the Appendix for the derivation of both the E and M steps, that give rise to Algorithm 1.

Note that if we do not share parameters across the policies for each preference type z, i.e. we have separate parameters ϕ_z for each $z \in \{z_1, \ldots, z_K\}$, then the optimization in the final step of EM-DPO also decomposes into separate policy optimization problems for each preference type:

$$\phi_{z_k} = \arg\max_{\phi_{z_k}} \sum_{i \in \mathcal{I}} \sum_{j=1}^{m_i} \gamma_{i,k} \, \log\big(P_\phi(y_w^{i,j} \succ Y_r^{i,j} \mid x^{i,j}, z_k)\big)$$

Note that the latter is simply a weighted version of the multi-item DPO formulation (Chen et al. (2024)), where each demonstration $V_{i,j}$, which corresponds to the j-th demonstrations from annotator i, is assigned weight $\gamma_{i,k}$ when optimizing the policy parameters for preference type z_k. Alternatively, some parameters can be shared across policies for each preference type, in which case the final optimization problem should be solved simultaneously via stochastic gradient descent over the joint parameters ϕ.

3.2 Fair Aggregation via Min-Max Regret

So far, we have outlined how to train an ensemble of LLMs, each optimized for one of the K preference sub-groups. We now turn to the problem of aggregating this ensemble into a single fair policy that balances group preferences by minimizing worst-case regret across sub-populations.

We assume that EM-DPO is able to correctly identify data belonging to each sub-group and outputs

Algorithm 1 EM-DPO

Input: Preference dataset $\mathcal{D}$ from annotators $\mathcal{H}$, with m_i demonstrations from annotator i
Input: Supervised fine-tuned model π_{SFT}
Input: K copies of π_{SFT}, denoted π_{ϕ_0,z_k} for $k = 1, \ldots, K$
Initialize mixture weights $(\eta_{1,0}, \ldots, \eta_{K,0}) = (1/K, \ldots, 1/K)$
for $t = 0, 1, \ldots, T$ **do**
  E-step: For each annotator $i \in \mathcal{H}$, compute

$$\gamma_{i,k} = \frac{\eta_{k,t} \prod_{j=1}^{m_i} P_{\phi_t}(y_w^{i,j} \succ Y_r^{i,j} \mid x^{i,j}, z_k)}{\sum_{\ell=1}^{K} \eta_{\ell,t} \prod_{j=1}^{m_i} P_{\phi_t}(y_w^{i,j} \succ Y_r^{i,j} \mid x^{i,j}, z_\ell)}$$

  where $Y_r^{i,j}$ is the set of rejected items in demonstration j for annotator i, and

$$P_{\phi_t}(y_w \succ Y_r \mid x, z_k) = \frac{\exp\Big(\log \frac{\pi_{\phi_t,z_k}(y_w|x)}{\pi_{\text{SFT}}(y_w|x)}\Big)}{\sum_{y \in \{y_w\} \cup Y_r} \exp\Big(\log \frac{\pi_{\phi_t,z_k}(y|x)}{\pi_{\text{SFT}}(y|x)}\Big)}$$

  M-step: Update

$$\eta_{k,t+1} = \frac{1}{|\mathcal{H}|} \sum_{i \in \mathcal{H}} \gamma_{i,k}, \qquad \phi_{t+1} = \arg\max_{\phi} \sum_{i \in \mathcal{H}} \sum_{k=1}^{K} \gamma_{i,k} \sum_{j=1}^{m_i} \log P_{\phi}(y_w^{i,j} \succ Y_r^{i,j} \mid x^{i,j}, z_k)$$

end for
Return: Policies $\{\pi_k^* = \pi_{\phi_T,z_k}\}_{k=1}^{K}$ and annotator weights $\{\gamma_{i,k}\}_{i \in \mathcal{H}}$

policies π_k^*s that optimize the true reward for each group, $r^*(y, x; z_k)$. For ease of notation let $\mathbb{E}_\pi[\cdot] := \mathbb{E}_{x \sim \mathcal{D},\, y \sim \pi(\cdot|x)}[\cdot]$, then:

$$\pi_k^* = \arg\max_{\pi} \mathbb{E}_\pi \left[r^*(y, x; z_k)\right] - \beta \mathbb{D}_{\text{KL}} \left(\pi(y|x) || \pi_{\text{SFT}}(y|x)\right)$$

Define $R_k(\pi)$ as the expected regret for population $Z = z_k$, measured relative to its population-optimal policy π_k^*:

$$R_k(\pi) := \mathbb{E}_{\pi_k^*} \left[r^*(y, x; z_k)\right] - \mathbb{E}_\pi \left[r^*(y, x; z_k)\right] \qquad (3)$$

This captures the loss in expected reward for subgroup k when following π instead of its optimal policy π_k^*. Our fairness criterion is to minimize the worst-case subgroup regret. Accordingly, we seek a policy $\pi \in \Pi$ that solves:

$$\pi_* = \arg\min_{\pi \in \Pi} \max_{w \in \Delta^{K-1}} \sum_k w_k([R_k(\pi)]^+ + \beta \mathbb{D}_{KL}(\pi || \pi_{\text{SFT}}))$$

$$\beta \mathbb{D}_{KL}(\pi || \pi_{\text{SFT}})) = E_\pi \left[\beta \log \frac{\pi(y|x)}{\pi_{\text{SFT}}(y|x)}\right] \qquad (4)$$

$$R_k(\pi) = \mathbb{E}_{\pi_k^*} \left[\beta \log \frac{\pi_k^*(y|x)}{\pi_{\text{SFT}}(y|x)}\right] - \mathbb{E}_\pi \left[\beta \log \frac{\pi_k^*(y|x)}{\pi_{\text{SFT}}(y|x)}\right]$$

where $[x]^+ = \max\{x, 0\}$, β is the regularization parameter and ϕ parametrizes π. Ideally, this objective could be directly optimized, with the maximizing player using multiplicative weights updates and the minimizing player applying gradient descent, as described in Algorithm 3 in the Appendix. However, this approach is computationally and memory intensive, requiring continuous generation from π and simultaneous access to all K policies to compute the log probabilities needed for estimating the $R_k(\pi)$ terms. We show that restricting to the space of affine combinations of the ensemble models yields an efficient solution with provable guarantees, given in Algorithm 4 in the Appendix. In this paper, however, we forgo the affine assumption and instead deploy a lightweight approximation that retains the multiplicative-weights-versus-gradient-descent structure, outlined in Algorithm 2. The key computational heuristic is as follows: training DPO with weights $\gamma_{i,k}$ yields the optimal policy for group k as in EM-DPO, and weights w_k^t are set proportional to the exponentiated regrets of policy k. Combining these, weighted DPO with weights $\gamma_{i,k} \cdot w_k^t$ post-trains the model favoring high-regret groups, reducing the full minimax objective to a weighted DPO loss.

4 IDENTIFIABILITY: THE VALUE OF THREE CHOICES

A natural question arises: under what conditions can we reliably recover the preference distribution in the population from observed data? In particular, since

Algorithm 2 MMRA-LW (MinMax Regret Aggregation - Lightweight)

Input: EM-DPO ensemble policies $\{\pi_1^*, .., \pi_K^*\}$ & weights $\{\gamma_{i,k} : i \in \mathcal{H}, k \in [K]\}$; SFT policy π_{SFT}; preference dataset $\mathcal{D}$; learning rate η; regularization parameter β;
Initialize: weights $(w_1^0, ..., w_K^0) = (1/K, ..., 1/K)$; current policy $\pi^0 = \pi_{\text{SFT}}$ for $t = 0$
for $t = 0, 1, \ldots, T$ **do**
    Compute weights as $\gamma_i = \sum_k w_k^{t-1} \gamma_{i,k}$
    for a few batches in $\mathcal{D}$ **do**
        Perform weighted DPO as in Algorithm 1 to get π^t
    end for
    Compute regrets $R_k(\pi^t)$ as per 4
    Update weights $w_k^t \propto \exp(R_k(\pi) \times \eta)$
end for

we assume no access to annotator identities or group labels, we must determine whether the latent distribution over preference types is uniquely identifiable from preference observations alone. In this section, we formalize this identifiability question and perform the analysis under a linear reward model. Somewhat surprisingly, we find that binary preferences suffer from fundamental non-identifiability, while multi-item preferences, even incomplete ternary preferences, enable unique recovery of the latent preference distribution.

Building on the Bradley–Terry framework introduced in Section 2, we consider a simplified reward model for the identification analysis in our paper where each annotator's reward function is linear in a known feature representation:

$$r^*(x, y \mid \beta) = \beta^\top \psi(x, y),$$

where $\psi(x, y) \in \mathbb{R}^d$ encodes features of prompt-response pairs, and $\beta \in \mathbb{R}^d$ captures user-specific preferences and can vary across annotators. While the main motivation for this simplification was to enable analysis, this linear model also has practical relevance. For instance, imagine a scenario where responses are evaluated along interpretable axes such as relevance, informativeness, and style. Each annotator may weigh these axes differently; one user might prioritize relevance heavily, while another emphasizes style. Representing overall preference as a weighted sum of these features naturally leads to a linear reward function. This linear reward modeling approach is also widely used in the alignment literature, particularly in works that study multi-objective rewards (Wang et al. (2024); Zhou et al. (2023); Yang et al. (2024b)). Also, note that while we focus on linear rewards for the identifiability analysis, the algorithms presented in this paper can be applied to arbitrary reward models.

When the preference vector β is drawn from a distribution $f(\beta)$, the aggregate choice probabilities correspond to the random coefficient logit model (Boyd and Mellman (1980); Cardell and Dunbar (1980)):

$$p_{r^*}(y_1 \succ y_2, ..., y_n|x) = \int \frac{\exp(\beta^\top \psi(x, y_1))}{\sum_{i=1}^n \exp(\beta^\top \psi(x, y_i))} f(\beta) d\beta \quad (5)$$

In this setting, the primary goal is to uncover f, which is the distribution over β across the population, since this distribution determines both aggregate choice probabilities and the range of individual user preferences. Identification means that the true distribution f_0 is the unique f that solves this equation for all possible values of the feature vectors $\psi(x, y_i)$. Equivalently, for any two distinct distributions $f_0 \neq f_1$, there exists some configuration of features (x, y) where the choice probabilities differ: $p_{r^*}(y_1 \succ y_2, ..., y_n|x; f_0) \neq p_{r^*}(y_1 \succ y_2, ..., y_n|x; f_1)$. Understanding the conditions under which this distribution can be uniquely identified is the focus of our subsequent analysis.

Case 1: Single Binary Preference per User, Infinite Users. This setting closely resembles real-world scenarios. In this case, the distribution over user parameters, $f(\beta)$, is not fully identifiable:

Lemma 4.1. *Under the random coefficient logit model* (5) *with binary preferences, f is not identifiable.*

Proof. Consider a distribution f where, with probability 1/2, the user parameter is β, and with probability 1/2 it is $-\beta$. For any pair (x, y_1, y_2), the aggregate preference is

$$\begin{aligned} p_{r^*}(y_1 \succ y_2 \mid x, f) &= 0.5\, \sigma(\beta^\top (\psi(x, y_1) - \psi(x, y_2))) \\ &\quad + 0.5\, \sigma(-\beta^\top (\psi(x, y_1) - \psi(x, y_2))) \\ &= 0.5 \end{aligned}$$

where we used $\sigma(x) = 1 - \sigma(-x)$. Hence, multiple distinct distributions (here, β vs. $-\beta$) yield the same aggregate probabilities, proving f is not identifiable. ∎

Case 2: Single Ternary Preference per User, Infinite Users. Comparisons over at least three items resolve the non-identifiability problem that arises in binary preferences, where even infinitely many users with finite observations per user cannot provably recover the underlying preference distribution. Under mild conditions, Fox et al. (2012) shows that the distribution $f(\beta)$ becomes uniquely identifiable with infinitely many users, at least one preference per user, and sufficient variability in the observed ternary (even incomplete) rankings. We re-state this theorem here for completeness, the complete proof can be found in Fox et al. (2012):

Theorem 4.2 (Identification of Random Coefficients Logit)**.** *If the following conditions hold:*

- *The absolute moments of f, given by $m_l = \int \|\beta\|^l f(\beta)\, d\beta$, are finite for $l \geq 1$ and satisfy the Carleman condition: $\sum_{l\geq 1} m_l^{-1/l} = \infty$.*
- *The feature vectors $\psi(x, y_i)$ for each alternative $i \in \{1, ..., n\}$ take on support in an open set containing $\psi(x, y_i) = 0$ for all i.*
- *β is independent of the features $\psi(x, y_i)$.*
- *$n \geq 3$ (at least 3 alternatives in the choice set).*

Then the density $f(\beta)$ is nonparametrically identified in the random coefficients logit model from equation (5)*.*

The assumptions presented in Theorem 4.2 have natural interpretations in the RLHF setting. The Carleman condition, satisfied by most common distributions (normal, uniform, exponential, and any bounded distribution), ensures f is uniquely determined by its moments. The assumption that feature vectors span an open set containing 0 requires the prompt-response dataset to exhibit sufficient variation along multiple dimensions (e.g., helpfulness, harmfulness, coherence) to disentangle different annotator preference types. Finally, the independence of β and the feature vectors $\psi(x, y_i)$ extends Assumption 2, which assumes prompts are independent of user type, to require that both prompts and responses shown to a user are independent of the user's type. The standard practice of generating responses from supervised-fine tuned models for all users naturally satisfies this assumption.

Case 3: Many Diverse Binary Preferences for Each User If we have many binary preference observations from a single user with sufficiently diverse comparisons across items, their parameter vector β can be identified directly:

Lemma 4.3. *Define the matrix $U \in \mathbb{R}^{m\times d}$ whose rows are $\phi(x, y_1) - \phi(x, y_2)$ across m preference pairs. If U is full rank, then*

$$\beta^\top(\psi(x, y_1) - \psi(x, y_2)) = \log \frac{p_{r^*}(y_1 \succ y_2|x, \beta)}{1 - p_{r^*}(y_1 \succ y_2|x, \beta)}$$

can be solved uniquely for β.

However, this solution requires a substantially rich set of responses per user. In particular, the number of diverse binary preferences per user needs to scale with the number of parameters in the reward model. For realistic reward models, we expect the feature map ψ to be very high-dimensional and, therefore, this solution to the non-identifiability problem would require a very rich set of responses per user, essentially growing to infinity, as we consider richer and richer reward models.

5 EXPERIMENTS

In this section, we provide empirical evidence to answer the questions: (1) Does the EM-DPO algorithm learn high-quality clusters? (2) How fair are the final policies learned by Min-Max Regret DPO? (3) Do ternary preferences out-perform binary preferences in the adversarial case we discussed in section 4.1.

5.1 Results

Datasets: We evaluate on two datasets that contain subgroups with differing preferences and imbalanced sizes, making them well-suited for studying pluralistic alignment: (1) GlobalOpinionQA (Durmus et al. (2023)), and (2) MPI (Jiang et al. (2022)).

GlobalOpinionQA. This dataset contains country-level polling data on politics, religion, economics, and related topics. For each question, annotators respond according to their country-specific distribution, after which a second option is randomly rejected. For example, a Mexican annotator answering "Do you support or oppose using the army to fight drug traffickers?" responds with probabilities of 84% support, 13% oppose, and 3% refuse/don't know. We focus on four countries: Britain (15%), Indonesia (20%), Mexico (30%), and Pakistan (35%), comprising a total of 48,000 preference pairs. Each question is augmented with 10 GPT-4 generated rephrasings, yielding 11 variants per question (original plus 10 rephrasings).

For training, eight variants are used and three are reserved for validation and testing. To construct a user, we sample 32 unique questions and assign a random rephrasing from the appropriate split. Using this procedure, we generate 1,500 users for training and 400

Table 1: Mean Reward Margins and Accuracies for Different Algorithms and User Types for MPI Dataset

	Reward Margins			Accuracies		
Method	**P1**	**P2**	**P3**	**P1**	**P2**	**P3**
True Label DPO	0.977	1.226	0.696	0.760	0.773	0.710
EMDPO Ternary	0.152	**0.423**	**0.729**	**0.584**	**0.584**	0.634
EMDPO Binary	0.116	0.134	0.585	0.515	0.550	0.651
Cluster DPO Ternary	**0.246**	0.257	0.688	0.560	0.562	0.591
Cluster DPO Binary	0.222	0.223	0.546	0.544	0.547	0.584
Vanilla DPO	0.031	0.020	0.228	0.532	0.512	**0.666**

Table 2: Max Mean Reward Margins and Accuracies for Different Clustering Algorithms and User Types for Global Opinions Dataset

	Reward Margins				Accuracies			
Method	**BR**	**IN**	**MX**	**PK**	**BR**	**IN**	**MX**	**PK**
True Label DPO	0.733	1.302	1.233	1.183	0.703	0.740	0.738	0.779
EMDPO	**0.853**	**1.474**	**1.558**	**1.617**	**0.710**	0.715	0.747	**0.767**
Cluster DPO	0.750	1.228	1.307	1.163	0.702	0.718	0.747	0.765
Vanilla DPO	0.613	1.139	1.187	1.187	0.681	**0.732**	**0.754**	0.754

for testing. Binary preferences are simulated from annotators and used to run the binary preference variant of EMDPO.

MPI. The MPI dataset contains 990 phrases annotated with trait scores in $\{-1, 0, +1\}$ for one of the five OCEAN personality traits McCrae and John (1992). For example, "act wild and crazy" scores $+1$ on Extraversion, while "readily overcome setbacks" scores -1 on Conscientiousness. We define three synthetic personalities as vectors in $\mathbb{R}^5$: $P1 = (3, 0, 2, 0, -2.5), P2 = (-3, 0, -2, 0, 2.5), P3 = (0, 2, 0, 2, 0)$ sampled with probabilities 0.3, 0.3, and 0.4. Note that probability of $P1$ is the same as that of $P2$ and $P2 = -P1$. We intentionally construct this adversarial user distribution to break the identification argument as discussed in Section 4. While we are not aware of any real-world alignment datasets covering this failure more, users with opposite preference weights naturally arise in many settings: (1) in pluralistic alignment, consider support for redistributive policies versus military spending left- and right-leaning users naturally have opposite signs for these weights and (2) in personalization, preferences for product attributes (bright vs. muted colors, budget vs. premium items) can exhibit opposing patterns across users.

Phrase rewards are computed as the inner product of the personality vector with phrase trait scores. To generate data, we (i) sample a personality, (ii) select one of 50 paraphrases of the instruction "Choose the option that resonates most with your personality," (iii) draw phrases from the MPI dataset, and (iv) simulate preferences using the Bradley Terry model with either two or three items. Each user contributes exactly one preference pair, preventing identifiability. Both binary and ternary preferences are simulated, and we run both versions of EMDPO for comparison.

Methods: Our approach consists of two components: clustering and aggregation. In addition to comparing with `Vanilla DPO`, which trains a single DPO model on the full dataset without clustering, we evaluate the following benchmarks: For clustering, our primary method is `EM-DPO`. For the MPI dataset, we run both the binary and ternary preference versions of `EM-DPO`. As a baseline, we use `Cluster DPO`, which partitions users with K-means clustering on response embeddings and then trains DPO separately on each cluster to produce an ensemble of policies. We also include `True Label DPO` as a benchmark, which trains an ensemble of models directly on large data from the original labels without any clustering algorithm. This is in a way the "best possible" situation where we know the latent variables and have large data.

For aggregation, our main method is `Min-Max Regret Aggregation`, which combines the ensemble of policies by minimizing the maximum regret across annotator groups inferred in the clustering step. We compare against `Uniform Sampling`, which averages the ensemble policies uniformly.

Metrics: To evaluate the quality of the learned clusters, we measure how well the ensemble covers diverse user preferences. For each user group, we identify the best LLM in the ensemble, defined as the policy that

Table 3: Regret Values for Each Latent Sub-group Across Datasets

	Global Opinions					MPI				
Algorithm	**1**	**2**	**3**	**4**	**Max**	**1**	**2**	**3**	**4**	**Max**
MMRA-LW	0	0	1.73	0	**1.73**	0	3.44	0	1.74	**3.44**
Uniform Sampling	2.84	2.61	3.54	0	3.54	5.98	6.26	4.44	4.80	6.26
Vanilla DPO	0.78	0	3.18	0	3.18	0	4.14	0	1.46	4.14

achieves the highest average reward margin on that group's evaluation dataset. This captures the extent to which clustering produces specialized policies aligned with different user populations, and it is invariant to the indexing of clusters since only the best match matters. Formally, if $\{\pi_i^*\}_{i=1}^K$ denotes the ensemble of policies returned by clustering and $\mathcal{D}_{k'}$ is the evaluation dataset for group k', the metric is:

$$\text{Max-mean reward margin}_{k'} = \max_{i\in[K]} \mathbb{E}_{(y_w,y_l,x)\sim\mathcal{D}_{k'}}\left[\beta \log \frac{\pi_i^*(y_w|x)/\pi_{\text{SFT}}(y_w|x)}{\pi_i^*(y_l|x)/\pi_{\text{SFT}}(y_l|x)}\right]$$

For the aggregation step, since we adopt the min–max regret objective as our fairness criterion, we evaluate the aggregated policy by measuring the worst-case regret across user sub-populations. This metric quantifies the maximum shortfall any sub-population experiences, ensuring that the evaluation emphasizes fairness across all groups:

$$\text{Max-regret}_{k\in[K]} = \max_{k\in[K]} R_k(\pi)$$

where $R_k(.)$ is as defined in Eq. 3 and π is the aggregated policy.

Discussion For the GlobalOpinions dataset, EM-DPO achieves strong performance on reward margins compared to baselines and delivers comparable accuracy. Notably, it outperforms models trained on large amounts of data with true labels in terms of reward margins, suggesting that EM-DPO can uncover latent structure even within annotated labels. The MMRA algorithm also performs well, yielding zero positive regret for Britain, Indonesia, and Pakistan, and only modest regret for Mexico, relative to uniform sampling from the ensemble policies and just training a single policy using Vanilla DPO.

On the MPI dataset, combining ternary preferences with EM-DPO leads to the best performance across both reward margins and accuracy. Interestingly, even Cluster-DPO with ternary preferences (which clusters prompts and chosen text before applying DPO) surpasses EM-DPO trained with binary preferences. This highlights the value of ternary preferences, particularly in the adversarial setting we constructed. For the aggregation step, we also notice that MMRA-LW out performs uniform sampling and vanilla DPO significantly, almost halving the regret values.

6 LIMITATIONS & OPEN QUESTIONS

First, we assume that annotators belong to one of K discrete latent groups, whereas in reality preferences most likely lie on a continuous spectrum. However, many areas (psychometrics, economics, recommender systems) use discrete "types" as standard approximations to continuous preference spaces (Hagenaars and McCutcheon (2002); Train (2009); Sarwar et al. (2001)). Moreover, this assumption improves interpretability and facilitates identification with limited data. Second, our method relies on the expectation–maximization (EM) algorithm, which is sensitive to initialization and may converge to saddle points. While we initialize using assignment from the K-means cluster algorithm, alternative strategies such as leveraging annotator demographics or employing an LLM-as-a-judge for initial estimates could improve robustness. Understanding the convergence and stability of EM in the context of large language models is an interesting avenue for future research. Additionally, extending the identification theory for more general classes of reward models beyond linear rewards is an open question.

Acknowledgements

We thank all reviewers who gave useful comments. KC is supported by the William R. and Sara Hart Kimball Stanford Graduate Fellowship.

References

Youssef Abdelkareem, Shady Shehata, and Fakhri Karray. Advances in preference-based reinforcement learning: A review. In *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2527–2532. IEEE, 2022.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas

Joseph, Ben Mann, Nova DasSarma, et al. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*, 2021.

Anirudhan Badrinath, Prabhat Agarwal, and Jiajing Xu. Hybrid preference optimization: Augmenting direct preference optimization with auxiliary objectives. *arXiv preprint arXiv:2405.17956*, 2024.

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.

Andreea Bobu, Andi Peng, Pulkit Agrawal, Julie Shah, and Anca D Dragan. Aligning robot and human representations. *arXiv preprint arXiv:2302.01928*, 2023.

Michael Bowling, John D Martin, David Abel, and Will Dabney. Settling the reward hypothesis. In *International Conference on Machine Learning*, pages 3003–3020. PMLR, 2023.

J Hayden Boyd and Robert E Mellman. The effect of fuel economy standards on the us automotive market: an hedonic demand analysis. *Transportation Research Part A: General*, 14(5-6):367–378, 1980.

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.

N Scott Cardell and Frederick C Dunbar. Measuring the societal impacts of automobile downsizing. *Transportation Research Part A: General*, 14(5-6): 423–434, 1980.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*, 2023.

Souradip Chakraborty, Jiahao Qiu, Hui Yuan, Alec Koppel, Furong Huang, Dinesh Manocha, Amrit Singh Bedi, and Mengdi Wang. Maxmin-rlhf: Towards equitable alignment of large language models with diverse human preferences. *arXiv preprint arXiv:2402.08925*, 2024.

Yuxin Chen, Junfei Tan, An Zhang, Zhengyi Yang, Leheng Sheng, Enzhi Zhang, Xiang Wang, and Tat-Seng Chua. On softmax direct preference optimization for recommendation. *Advances in Neural Information Processing Systems*, 37:27463–27489, 2024.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Vincent Conitzer, Rachel Freedman, Jobst Heitzig, Wesley H Holliday, Bob M Jacobs, Nathan Lambert, Milan Mossé, Eric Pacuit, Stuart Russell, Hailey Schoelkopf, et al. Social choice for ai alignment: Dealing with diverse human feedback. *arXiv preprint arXiv:2404.10271*, 2024.

Jessica Dai and Eve Fleisig. Mapping social choice theory to rlhf. *arXiv preprint arXiv:2404.13038*, 2024.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1):1–22, 1977.

Vincent Dumoulin, Daniel D Johnson, Pablo Samuel Castro, Hugo Larochelle, and Yann Dauphin. A density estimation perspective on learning from pairwise human preferences. *arXiv preprint arXiv:2311.14115*, 2023.

Esin Durmus, Karina Nguyen, Thomas I 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. *arXiv preprint arXiv:2306.16388*, 2023.

Owain Evans, Andreas Stuhlmüller, and Noah Goodman. Learning the preferences of ignorant, inconsistent agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

Jeremy T Fox, Kyoo il Kim, Stephen P Ryan, and Patrick Bajari. The random coefficients logit model is identified. *Journal of Econometrics*, 166(2):204–212, 2012.

Justin Fu, Anoop Korattikara, Sergey Levine, and Sergio Guadarrama. From language to goals: Inverse reinforcement learning for vision-based instruction following. *arXiv preprint arXiv:1902.07742*, 2019.

Paul Gölz, Nika Haghtalab, and Kunhe Yang. Distortion of ai alignment: Does preference optimization optimize for preferences? *arXiv preprint arXiv:2505.23749*, 2025.

Jacques A Hagenaars and Allan L McCutcheon. *Applied latent class analysis*. Cambridge University Press, 2002.

Joey Hong, Kush Bhatia, and Anca Dragan. On the sensitivity of reward inference to misspecified human models. *arXiv preprint arXiv:2212.04717*, 2022.

Minyoung Hwang, Gunmin Lee, Hogun Kee, Chan Woo Kim, Kyungjae Lee, and Songhwai Oh. Sequential preference ranking for efficient reinforcement learning from human feedback. *Advances in Neural Information Processing Systems*, 36, 2024.

Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. Reward learning from human preferences and demonstrations in atari. *Advances in neural information processing systems*, 31, 2018.

Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. Personalized soups: Personalized large language model alignment via post-hoc parameter merging. *arXiv preprint arXiv:2310.11564*, 2023.

Guangyuan Jiang, Manjie Xu, Song-Chun Zhu, Wenjuan Han, Chi Zhang, and Yixin Zhu. Mpi: Evaluating and inducing personality in pre-trained language models. *arXiv preprint arXiv:2206.07550*, 2022.

Michael Kearns, Yishay Mansour, and Andrew Y Ng. An information-theoretic analysis of hard and soft assignment methods for clustering. In *Learning in graphical models*, pages 495–520. Springer, 1998.

Hung Le, Quan Tran, Dung Nguyen, Kien Do, Saloni Mittal, Kelechi Ogueji, and Svetha Venkatesh. Multi-reference preference optimization for large language models. *arXiv preprint arXiv:2405.16388*, 2024.

Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. B-pref: Benchmarking preference-based reinforcement learning. *arXiv preprint arXiv:2111.03026*, 2021.

Dexun Li, Cong Zhang, Kuicai Dong, Derrick Goh Xin Deik, Ruiming Tang, and Yong Liu. Aligning crowd feedback via distributional preference reward modeling. *arXiv preprint arXiv:2402.09764*, 2024.

David Lindner and Mennatallah El-Assady. Humans are not boltzmann distributions: Challenges and opportunities for modelling human feedback and interaction in reinforcement learning. *arXiv preprint arXiv:2206.13316*, 2022.

Yinhan Liu. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Anirudha Majumdar, Sumeet Singh, Ajay Mandlekar, and Marco Pavone. Risk-sensitive inverse reinforcement learning via coherent risk models. In *Robotics: science and systems*, volume 16, page 117, 2017.

Robert R McCrae and Oliver P John. An introduction to the five-factor model and its applications. *Journal of personality*, 60(2):175–215, 1992.

Todd K Moon. The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6):47–60, 1996.

Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Côme Fiegel, et al. Nash learning from human feedback. In *Forty-first International Conference on Machine Learning*, 2024.

Khanh Nguyen, Hal Daumé III, and Jordan Boyd-Graber. Reinforcement learning for bandit neural machine translation with simulated human feedback. *arXiv preprint arXiv:1707.07402*, 2017.

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.

Chanwoo Park, Mingyang Liu, Dingwen Kong, Kaiqing Zhang, and Asuman Ozdaglar. Rlhf from heterogeneous feedback via personalization and preference aggregation. *arXiv preprint arXiv:2405.00254*, 2024.

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. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to q*: Your language model is secretly a q-function. *arXiv preprint arXiv:2404.12358*, 2024.

Sasha Rakhlin and Karthik Sridharan. Optimization, learning, and games with predictable sequences. *Advances in Neural Information Processing Systems*, 26, 2013.

Alexandre Rame, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. Rewarded soups: towards pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards. *Advances in Neural Information Processing Systems*, 36, 2024.

Alexandre Ramé, Nino Vieillard, Léonard Hussenot, Robert Dadashi, Geoffrey Cideron, Olivier Bachem, and Johan Ferret. Warm: On the benefits of weight averaged reward models. *arXiv preprint arXiv:2401.12187*, 2024.

Shyam Sundhar Ramesh, Yifan Hu, Iason Chaimalas, Viraj Mehta, Pier Giuseppe Sessa, Haitham Bou Ammar, and Ilija Bogunovic. Group robust preference optimization in reward-free rlhf. *Advances in Neural Information Processing Systems*, 37:37100–37137, 2024.

Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295, 2001.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Rohin Shah, Noah Gundotra, Pieter Abbeel, and Anca Dragan. On the feasibility of learning, rather than assuming, human biases for reward inference. In *International Conference on Machine Learning*, pages 5670–5679. PMLR, 2019.

Anand Siththaranjan, Cassidy Laidlaw, and Dylan Hadfield-Menell. Distributional preference learning: Understanding and accounting for hidden context in rlhf. *arXiv preprint arXiv:2312.08358*, 2023.

Joar Max Viktor Skalse and Alessandro Abate. The reward hypothesis is false. In *NeurIPS ML Safety Workshop*, 2022. URL https://openreview.net/forum?id=5l1NgpzAfH.

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.

Gokul Swamy, Christoph Dann, Rahul Kidambi, Zhiwei Steven Wu, and Alekh Agarwal. A minimaximalist approach to reinforcement learning from human feedback. *arXiv preprint arXiv:2401.04056*, 2024.

Kenneth E Train. *Discrete choice methods with simulation*. Cambridge university press, 2009.

Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse kl: Generalizing direct preference optimization with diverse divergence constraints. *arXiv preprint arXiv:2309.16240*, 2023a.

Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. Arithmetic control of llms for diverse user preferences: Directional preference alignment with multi-objective rewards. *arXiv preprint arXiv:2402.18571*, 2024.

Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. Aligning large language models with human: A survey. *arXiv preprint arXiv:2307.12966*, 2023b.

Christian Wirth, Riad Akrour, Gerhard Neumann, and Johannes Fürnkranz. A survey of preference-based reinforcement learning methods. *Journal of Machine Learning Research*, 18(136):1–46, 2017.

Jeff Wu, Long Ouyang, Daniel M Ziegler, Nisan Stiennon, Ryan Lowe, Jan Leike, and Paul Christiano. Recursively summarizing books with human feedback. *arXiv preprint arXiv:2109.10862*, 2021.

Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jiawei Chen, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. Towards robust alignment of language models: Distributionally robustifying direct preference optimization. *arXiv preprint arXiv:2407.07880*, 2024.

Kailai Yang, Zhiwei Liu, Qianqian Xie, Tianlin Zhang, Nirui Song, Jimin Huang, Ziyan Kuang, and Sophia Ananiadou. Metaaligner: Conditional weak-to-strong correction for generalizable multi-objective alignment of language models. *arXiv preprint arXiv:2403.17141*, 2024a.

Rui Yang, Xiaoman Pan, Feng Luo, Shuang Qiu, Han Zhong, Dong Yu, and Jianshu Chen. Rewards-in-context: Multi-objective alignment of foundation models with dynamic preference adjustment. *arXiv preprint arXiv:2402.10207*, 2024b.

Yongcheng Zeng, Guoqing Liu, Weiyu Ma, Ning Yang, Haifeng Zhang, and Jun Wang. Token-level direct preference optimization. *arXiv preprint arXiv:2404.11999*, 2024.

Jiangchuan Zheng, Siyuan Liu, and Lionel M Ni. Robust bayesian inverse reinforcement learning with sparse behavior noise. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, 2014.

Huiying Zhong, Zhun Deng, Weijie J Su, Zhiwei Steven Wu, and Linjun Zhang. Provable multi-party reinforcement learning with diverse human feedback. *arXiv preprint arXiv:2403.05006*, 2024.

Li Zhou and Kevin Small. Inverse reinforcement learning with natural language goals. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11116–11124, 2021.

Zhanhui Zhou, Jie Liu, Chao Yang, Jing Shao, Yu Liu, Xiangyu Yue, Wanli Ouyang, and Yu Qiao. Beyond one-preference-fits-all alignment: Multi-objective direct preference optimization. *arXiv preprint arXiv:2310.03708*, 2023.

Zhanhui Zhou, Jie Liu, Jing Shao, Xiangyu Yue, Chao Yang, Wanli Ouyang, and Yu Qiao. Beyond one-preference-fits-all alignment: Multi-objective direct preference optimization. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 10586–10613, 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.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

APPENDIX

A ADDITIONAL RELATED WORK

DPO Generalizations: Since DPO's inception Rafailov et al. (2023), there has been a growing line of literature on its generalizations, some of which we highlight here. Le et al. (2024) generalizes DPO to the case of multiple SFT models, while Zhou et al. (2024) generalizes to multiple objectives. Zeng et al. (2024); Rafailov et al. (2024) work on extending DPO to work at the token level. Wang et al. (2023a) extends DPO to work with other types of divergence terms, while Wu et al. (2024) relates DPO to DRO in order to robustify it. Badrinath et al. (2024) augments DPO with a computable advantage function to create a hybrid between DPO and RLHF.

Preference-Based Reinforcement Learning: Reinforcement learning from preferences has been an active research area for some time, providing a way to train on tasks for which explicitly defining rewards is hard Wirth et al. (2017); Lee et al. (2021); Abdelkareem et al. (2022). In particular, Christiano et al. (2017); Ibarz et al. (2018) show that using human preferences to guide reinforcement learning (RLHF) is particularly effective on a variety of tasks, such as training robots. More recently, RLHF has become a very popular technique to fine-tune language models to do a variety of tasks such as summarization Ouyang et al. (2022); Ziegler et al. (2019); Stiennon et al. (2020); Wu et al. (2021). RLHF has also been used to align language models Bai et al. (2022); Askell et al. (2021). Casper et al. (2023) details several open problems in the field of RLHF, including those related to the feedback itself, particularly the inverse relation between richness and efficiency. Some work has been done on this problem with regards to language-based feedback in particular Fu et al. (2019); Zhou and Small (2021) as well as in more general settings Hwang et al. (2024), but specific applications to LLMs have not been fully explored.

Challenges with Reward Modeling: In general, human preferences can be difficult to represent using reward models Hong et al. (2022), and the validity of reward modeling itself is still somewhat debated Bowling et al. (2023); Bobu et al. (2023); Skalse and Abate (2022). Some work has also been done to take personality into account when reward modeling Lindner and El-Assady (2022); Lee et al. (2021), but this area remains open. In general, taking human irrationality into account when reward modeling (to optimize a more accurate reward function) leads to a trade-off between efficiency and accuracy Shah et al. (2019); Nguyen et al. (2017). Work has been done on inverse RL with particular models of suboptimality such as myopia Evans et al. (2016), noise Zheng et al. (2014), and risk-sensitivity Majumdar et al. (2017), but dealing with general irrationalities remains open. A recent work by Gölz et al. (2025) shows that under heterogeneous user preferences, standard alignment methods (PPO-based RLHF and DPO) built on the Bradley-Terry model can produce policies whose average utility falls short of the optimal achievable average utility. Gölz et al. (2025) recommends instead to use Nash Learning from Human Feedback Munos et al. (2024) as an efficient alternative to the Bradley Terry model.

B ALGORITHMS

B.1 EM-DPO Derivation

B.1.1 M-Step

First we parametrize the objective of the M-step, $Q(\theta \mid \theta_t)$. We can break down the optimization objective as follows:

$$
\begin{aligned}
Q(\theta \mid \theta_t) &= \mathbb{E}_{Z \sim p(\cdot|V,\theta_t)} \left[\log\left(p(V, Z; \theta)\right)\right] && (6)\\
&= \mathbb{E}_{Z \sim p(\cdot|V,\theta_t)} \left[\log\left(\prod_{i=1}^{n} p(V_i, Z_i; \theta)\right)\right] && (7)\\
&= \mathbb{E}_{Z \sim p(\cdot|V,\theta_t)} \left[\sum_{i=1}^{n} \log\left(p(V_i, Z_i; \theta)\right)\right] && (8)\\
&= \mathbb{E}_{Z \sim p(\cdot|V,\theta_t)} \left[\sum_{i=1}^{n} \log(p(V_i \mid Z_i; \theta)) + \log(p(Z_i; \theta))\right] && (9)\\
&= \mathbb{E}_{Z \sim p(\cdot|V,\theta_t)} \left[\sum_{i=1}^{n} \log(p(V_i \mid Z_i; \theta))\right] + \mathbb{E}_{Z \sim p(\cdot|V,\theta_t)} \left[\log(p(Z_i; \theta))\right] && (10)
\end{aligned}
$$

For our problem, θ comprises of three sets of parameters: ϕ which parametrizes the policies, η which parametrizes the latent distribution of the user-types $p(Z; \theta)$, and ρ which parametrizes the distribution of the prompts X. We will first see how this parametrizes the optimization problem.

First we look at $\log(p(Z_i; \theta))$. The latent factors Z take values from a discrete set of K values $\{z_1, \ldots, z_K\}$. In this case, we can assume a fully non-parametric likelihood $p(Z; \theta)$, where $\eta_k = p(z_k; \theta) \in \Delta(K)$, the K-dimensional simplex. Further, we note that:

$$
p(Z_i; \theta) = \sum_{k=1}^{K} \eta_k \mathbf{1}\{Z_i = z_k\} \tag{11}
$$

Further, assuming that $p(V_i \mid Z_i; \theta)$ does not depend on the vector η, so that $p(V_i \mid Z_i; \theta) = p(V_i \mid Z_i; \phi, \rho)$, the original criterion decomposes into two separate optimization problems:

$$
\begin{aligned}
\eta_{t+1} = &\quad \arg\max_{\eta} \mathbb{E}_{Z \sim p(\cdot|V,\theta_t)} \left[\sum_{i=1}^{n} \log\left(\sum_{k=1}^{K} \eta_k 1\{Z_i = z_k\}\right)\right] \\
\phi_{t+1} = &\quad \arg\max_{\phi,\rho} \mathbb{E}_{Z \sim p(\cdot|V,\theta_t)} \left[\sum_{i=1}^{n} \log(p(V_i \mid Z_i; \phi, \rho))\right]
\end{aligned} \tag{12}
$$

Next, we look at the parametrization of $p(V_i \mid Z_i; \phi, \rho)$. Note that, in our situation, both the latent factors and observed variables (Z_i, V_i) are independent across the n annotators and therefore, the likelihood and the prior factorizes across the annotators. Moreover, conditional on the latent factor Z_i, the $V_{i,j}$ are independently distributed across j and for each j the conditional likelihood takes a logistic form, as follows:

$$
\begin{aligned}
p(V_i \mid Z_i; \phi, \rho) = &\quad \prod_{j=1}^{m} p(V_{i,j} \mid Z_i; \phi, \rho) && (13)\\
= &\quad \prod_{j=1}^{m} P(y_w^{i,j} \succ Y_r^{i,j}, X^{i,j} \mid Z_i; \phi, \rho) && (14)\\
= &\quad \prod_{j=1}^{m} P(y_w^{i,j} \succ Y_r^{i,j} \mid X^{i,j}, Z_i; \phi, \rho)\, p(X^{i,j} \mid Z_i; \phi, \rho) && (15)\\
& && (16)
\end{aligned}
$$

where r^* denotes the true reward function for the annotator. Let the policy parametrized as $\pi_{\phi^*,Z}$ be the policy that is optimal for the true reward function r^* for the given latent type Z Rafailov et al. (2023). Now, the probability function in the first term can also be written in closed form in terms of these policy parameters:

$$P_{\phi_t}(y_w \succ Y_r \mid X, Z) = \frac{\exp\Big(\beta \log \frac{\pi_{\phi_t,z_k}(y_w|X)}{\pi_{\text{SFT}}(y_w|X)}\Big)}{\sum_{y\in\{y_w\}\cup Y_r} \exp\Big(\beta \log \frac{\pi_{\phi_t,z_k}(y|X)}{\pi_{\text{SFT}}(y|X)}\Big)} \tag{17}$$

where $\pi_{\phi^*,z}$ optimizes the type specific regularized objective:

$$\pi_{\phi^*,z} = \arg\max_{\pi} \mathbb{E}_{x\sim\mathcal{D},y\sim\pi(y|x)}[r^*(y,x,z)] - \beta\mathbb{D}_{\text{KL}}[\pi(y|x)||\pi_{\text{SFT}}(y|x)] \tag{18}$$

For concise writing we introduce the following shorthand notation:

$$P_\phi(V, Z) = P_{\phi_t}(y_w \succ Y_r \mid X, Z) \tag{19}$$

Thus a parameterization of the policy space $\pi_{\phi,Z}$, implies a parameterization of the likelihood:

$$p(V_i \mid Z_i; \phi, \rho) = \prod_{j=1}^{m} P_\phi(V_{i,j}, Z_i)\, p(X_{i,j} \mid Z_i; \phi, \rho) \tag{20}$$

Therefore, the M-step involves solving the following two optimization problems:

$$\begin{aligned} \eta_{t+1} = &\ \arg\max_{\eta} \mathbb{E}_{Z\sim p(\cdot|V,\theta_t)} \left[\sum_{i=1}^{n} \log\left(\sum_{k=1}^{K} \eta_k 1\{Z_i = z_k\}\right)\right] \\ \phi_{t+1} = &\ \arg\max_{\phi,\rho} \mathbb{E}_{Z\sim p(\cdot|V,\theta_t)} \left[\sum_{i=1}^{n}\sum_{j=1}^{m} \log(P_\phi(V_{i,j}, Z_i)) + \sum_{i=1}^{n}\sum_{j=1}^{m} \log(p(X_{i,j} \mid Z_i; \rho))\right] \end{aligned} \tag{21}$$

The first optimization problem for η_{t+1} admits a closed-form solution. Letting $w_{k,t} = \sum_{i=1}^{n} p(z_k \mid V_i; \theta_t)$

$$\mathbb{E}_{Z\sim p(\cdot|V,\theta_t)} \left[\sum_{i=1}^{n} \log\left(\sum_{k=1}^{K} \eta_k 1\{Z_i = z_k\}\right)\right] = \sum_{i=1}^{n}\sum_{k=1}^{K} p(z_k \mid V_i; \theta_t) \log(\eta_k) \tag{22}$$

$$= \sum_{k=1}^{K} w_{k,t} \log(\eta_k) \tag{23}$$

Thus the optimization problem that determines η_{t+1} takes the simple form $\max_{\eta\in\Delta(K)} \sum_{k=1}^{K} w_{k,t} \log(\eta_k)$. The Lagrangian of this problem is $L(\eta, w_t, \lambda) = \sum_{k=1}^{K} w_{k,t} \log(\eta_k) + \lambda^T(\eta - 1)$. The KKT condition is:

$$\frac{w_{k,t}}{\eta_{k,t+1}} = \lambda \implies \eta_{k,t+1} \propto w_{k,t} \implies \eta_{k,t+1} = \frac{w_{k,t}}{\sum_k w_{k,t}} \tag{24}$$

Moreover, since $\sum_k p(z_k \mid V_i; \theta_t) = 1$, we have $\sum_k w_{k,t} = n$. Thus, the above simplifies to:

$$\eta_{k,t+1} = \frac{1}{n} w_{k,t} = \frac{1}{n} \sum_{i=1}^{n} p(z_k \mid V_i; \theta_t) \tag{25}$$

For the second optimization problem for ϕ_{t+1}, assuming that the parameter ρ that determines that $p(X \mid Z; \rho)$ is not subject to joint constraints with the parameter ϕ, we can drop the second part in the objective to get:

$$\begin{aligned} \phi_{t+1} = &\ \arg\max_{\phi} \mathbb{E}_{Z\sim p(\cdot|V,\theta_t)} \left[\sum_{i=1}^{n}\sum_{j=1}^{m} \log(P_\phi(V_{i,j}, Z_i))\right] \\ = &\ \arg\max_{\phi} \sum_{i=1}^{n} \mathbb{E}_{Z_i\sim p(\cdot|V_i;\theta_t)} \left[\sum_{j=1}^{m} \log(P_\phi(Z_i, V_{ij}))\right] \end{aligned} \tag{26}$$

B.1.2 E-Step

The only remaining quantity we need to compute $Q(\theta \mid \theta_t)$ is the posterior distribution $p(Z \mid V; \theta) = \prod_{i=1}^{n} p(Z_i \mid V_i; \theta)$ for any given parameter θ. First we apply the Bayes rule and later substitute the parametric form of $p(V|Z;\theta)$ and $P(Z;\theta)$ that we derived in the M-Step to get:

$$
\begin{aligned}
p(z_k \mid V_i; \theta) &= \frac{p(V_i, z_k; \theta)}{p(V_i; \theta)} && (27)\\
&= \frac{p(V_i \mid z_k; \theta)\, p(z_k; \theta)}{\sum_{\ell=1}^{K} p(V_i \mid z_\ell; \theta)\, p(z_\ell; \theta)} && (28)\\
&= \frac{p(V_i \mid z_k; \phi)\, \eta_k}{\sum_{\ell=1}^{K} p(V_i \mid z_\ell; \phi)\, \eta_\ell} && (29)\\
&= \frac{\prod_{j=1}^{m} P_\phi(z_k, V_{ij})\, p(X_{ij} \mid z_k; \theta)\, \eta_k}{\sum_{\ell=1}^{K} \prod_{j=1}^{m} P_\phi(z_\ell, V_{ij})\, p(X_{ij} \mid z_\ell; \theta)\, \eta_\ell}. && (30)
\end{aligned}
$$

Now, we can invoke assumption 2 to get:

$$
p(z_k \mid V_i; \theta) = \frac{\prod_{j=1}^{m} P_\phi(z_k, V_{ij}) \rho(X_{ij})\, \eta_k}{\sum_{\ell=1}^{K} \prod_{j=1}^{m} P_\phi(z_\ell, V_{ij}) \rho(X_{ij})\, \eta_\ell} \tag{31}
$$

Note that we can write:

$$
\begin{aligned}
& \sum_{\ell=1}^{K} \prod_{j=1}^{m} P_\phi(z_\ell, V_{ij})\, \rho(X_{ij})\, \eta_\ell && (32)\\
=\; & \sum_{\ell=1}^{K} \prod_{j=1}^{m} \rho(X_{ij}) \cdot \prod_{j=1}^{m} P_\phi(z_\ell, V_{ij}) \eta_\ell && (33)\\
=\; & \prod_{j=1}^{m} \rho(X_{ij}) \cdot \sum_{\ell=1}^{K} \prod_{j=1}^{m} P_\phi(z_\ell, V_{ij}) \eta_\ell && (34)
\end{aligned}
$$

Thus, the terms $\prod_{j=1}^{m} \rho(X_j)$ cancel from the numerator and denominator in Equation (31), leading to the simplified formula that is independent of ρ:

$$
p(z_k \mid V_i; \theta) = \frac{\eta_k \prod_{j=1}^{m} P_\phi(z_k, V_j)}{\sum_{\ell=1}^{K} \eta_\ell \prod_{j=1}^{m} P_\phi(z_\ell, V_j)} \tag{35}
$$

B.1.3 Final Algorithm

Now we put together the E and M step. Let $\gamma_{i,k} = p(z_k \mid V_i; \theta_t)$, now,

$$
\begin{aligned}
\gamma_{i,k} &= \frac{\eta_k \prod_{j=1}^{m} P_\phi(z_k, V_{i,j})}{\sum_{\ell=1}^{K} \eta_\ell \prod_{j=1}^{m} P_\phi(z_\ell, V_j)} && (36)\\
\eta_{k,t+1} &= \frac{1}{n} \sum_{i=1}^{n} \gamma_{i,k} && (37)\\
\phi_{t+1} &= \arg\max_{\phi} \sum_{i \in \mathcal{I}} \sum_{k=1}^{K} \gamma_{i,k} \sum_{j=1}^{m_i} \log(P_\phi(V_{i,j}, z_k,)) && (38)
\end{aligned}
$$

C MIN-MAX REGRET AGGREGATION ALGORITHM (MMRA) VARIANTS

The algorithm that directly minimizes the regret objective provided in 4 is shown in Algorithm 3. This algorithm requires generation and scoring from the training policy as well as scoring from all the ensemble policies at every step of training, which consumes both compute and memory.

Algorithm 3 MMRA (MinMax Regret Aggregation -Original)

Input: EM-DPO ensemble policies $\{\pi_1^*, \dots, \pi_K^*\}$; SFT policy π_{SFT}; prompt dataset $\mathcal{D}_x$; preference dataset $\mathcal{D}$; regularization parameter β; learning rate η
Initialize: Current policy $\pi^0 = \pi_{\text{SFT}}$ for $t = 0$; weights $(w_1^0, \dots, w_K^0) = (1/K, \dots, 1/K)$;
Precompute: For each $k \in [K]$:
  Generate dataset $\mathcal{D}_k = \{(x, y) : x \sim \mathcal{D}_x, y \sim \pi_k^*(\cdot|x)\}$
  Approximate $\mathbb{E}_{x,y\sim\pi_k^*}\left[\beta \log \frac{\pi_k^*(y|x)}{\pi_{\text{SFT}}(y|x)}\right] \approx \frac{1}{|\mathcal{D}_k|}\sum_{(x,y)\in\mathcal{D}_k} \beta \log \frac{\pi_k^*(y|x)}{\pi_{\text{SFT}}(y|x)}$
for $t = 0, 1, \dots, T$ **do**
  Generate $\mathcal{D}^t = \{(x, y) : x \sim \mathcal{D}_x, y \sim \pi^t(\cdot|x)\}$
  Compute $\log \pi(y|x)$ and $\log \pi_k^*(y|x)$ for $x, y \sim \mathcal{D}^t$ and $\forall k \in [K]$
  Approximate:
    $\mathbb{E}_{x,y\sim\pi^t}\left[\beta \log \frac{\pi_k^*(y|x)}{\pi_{\text{SFT}}(y|x)}\right] \approx \frac{1}{|\mathcal{D}^t|}\sum_{x,y\in\mathcal{D}^t} \beta \log \frac{\pi_k^*(y|x)}{\pi_{\text{SFT}}(y|x)}, \forall k \in [K]$
    $\mathbb{E}_{x,y\sim\pi^t}\left[\beta \log \frac{\pi^t(y|x)}{\pi_{\text{SFT}}(y|x)}\right] \approx \frac{1}{|\mathcal{D}^t|}\sum_{x,y\in\mathcal{D}^t} \beta \log \frac{\pi^t(y|x)}{\pi_{\text{SFT}}(y|x)}$
  Compute the loss $\mathcal{L}_k^t = w_k^t([R_k(\pi^t)]^+ + \beta\mathbb{D}_{KL}(\pi^t||\pi_{\text{SFT}}))$.
  $\pi^t, w_k^t \leftarrow$ (regular or optimistic) GD and MWU on $\sum_k \mathcal{L}_k^t$
end for

Algorithm 4 is another computationally light-weight algorithm. This algorithm constrains the search space of the trained algorithm to the affine space of the ensemble policies. This doesn't require re-training a new policy and just involves selecting an ensemble among the already trained policies. As such, we define the ensemble space of policies as:

$$\Pi = \left\{\sum_{k=1}^{K} w_k \pi_{\phi,z_k} : w \in \Delta(K)\right\}$$

The min-max regret optimization problem can be written as a function of the policies trained using EM-DPO without any access to the explicit reward functions, as it is a difference in rewards (Rafailov et al. (2023)):

$$\min_{w\in\Delta(K)} \max_{z\in\{z_0,z_1,\dots,z_K\}} \sum_{k=1}^{K} w_k \cdot (\mathcal{L}_{z,z} - \mathcal{L}_{z,z_k})$$

where

$$\mathcal{L}_{z,z'} := \begin{cases} 0 & \text{if } z = z_0, \\ \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{z'}^*(\cdot|x)}\left[\log\left(\frac{\pi_z^*(y|x)}{\pi_{\text{SFT}}(y|x)}\right)\right] & \text{otherwise.} \end{cases}$$

Letting $\mathcal{R}$ denote the $(K+1)\times K$ matrix whose (k, k') entry (for $0 \le k \le K, 1 \le k' \le K$) corresponds to $\mathcal{R}_{k,k'} := \mathcal{L}_{z_k,z_k} - \mathcal{L}_{z_k,z_{k'}}$, we can re-write the above objective as:

$$\min_{w\in\Delta(K)} \max_{p\in\Delta(K+1)} p^\top \mathcal{R} w \tag{39}$$

This is simply a finite action zero-sum game, where the minimizing player has K actions and the maximizing player has $K+1$ actions. A large variety of methods can be utilized to calculate an equilibrium of this zero-sum game and hence identify the minimax regret optimal mixture weights w^*. For instance, we can employ optimistic Hedge vs. optimistic Hedge dynamics, which are known to achieve fast convergence rates in such finite action zero-sum games (Rakhlin and Sridharan (2013)) and then use the average of the solutions over the iterates of training. This is formalized in 4 and the solution π_* returned constitutes a $O(\log(K)\,\log(T)\,T^{-1})$-approximate solution to the min-max regret problem (a direct consequence of the results in Rakhlin and Sridharan (2013)). This completes our overall direct preference optimization procedure with unobserved heterogeneous preferences.

Algorithm 4 MMRA-AE (MinMax Regret Aggregation - Affine Ensemble Variant)

1: **Input:** Distribution $\mathcal{D}$ of contexts x.
2: **Input:** Population-specific optimal policies π_z^* returned from EM-DPO
3: **Input:** Number of iterations T and a sufficiently small, albeit constant, independent of T, step-size η
4: Calculate discrepancies for $z, z' \in \{z_1, \dots, z_k\}$:

$$\mathcal{L}_{z,z'} := \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi^*_{z'}(\cdot|x)}\left[\log\left(\frac{\pi_z^*(y|x)}{\pi_{\text{SFT}}(y|x)}\right)\right]$$

with the convention that $\mathcal{L}_{z_0,z_0} = \mathcal{L}_{z_0,z_k} = 0$
5: Calculate $(K+1)\times K$ regret matrix $\mathcal{R}$, whose $k \in \{0,\dots,K\}$ and $k' \in \{1,\dots,K\}$ entry is:

$$\mathcal{R}_{k,k'} := \mathcal{L}_{z_k,z_k} - \mathcal{L}_{z_k,z'_k}$$

6: Initialize $w_0 = (1/K, \dots, 1/K)$ and $p_0 = (1/(K+1), \dots, 1/(K+1))$
7: **for** t in $\{0,\dots,T\}$ **do**

$$w_t \propto w_{t-1}\exp\left\{-\eta\cdot\left(2\mathcal{R}^\top p_{t-1} - \mathcal{R}^\top p_{t-2}\right)\right\}$$

$$p_t \propto p_{t-1}\exp\left\{\eta\cdot\left(2\mathcal{R}w_{t-1} - \mathcal{R}w_{t-2}\right)\right\}$$

8: **end for**
9: **Return:** Policy $\pi^* = \sum_{k=1}^K w^*(k)\pi^*_{z_k}$, where $w^* = \frac{1}{T}\sum_{t=1}^T w_t$ and $w(k)$ is the value of the k^{th} co-ordinate of w.

D ADDITIONAL EXPERIMENT DETAILS

D.1 Data Generation

Global Opinion QA: This dataset contains country-level polling data on politics, religion, economics, and related topics. For each question, annotators respond according to their country-specific distribution, after which a second option is randomly rejected. For instance, a Mexican annotator answering "Do you support or oppose using the army to fight drug traffickers?" has probabilities of 84% support, 13% oppose, and 3% refuse/don't know. We focus on four countries: Britain, Indonesia, Mexico, and Pakistan. Each question is expanded with 10 GPT-4-generated rephrasings, yielding 11 variants per question (original + 10 rephrasings).

For model training, eight variants are used, while three are reserved for validation and testing. Annotators are simulated according to the following country proportions: 15% from Britain, 20% from Indonesia, 30% from Mexico, and 35% from Pakistan. To construct a user, 32 unique questions are sampled and a random rephrasing from the appropriate split is assigned for each. Using this approach, we generate 1,500 users for training and 400 for testing.

MPI: The MPI dataset contains 990 unique phrases, each scored -1, 0, or $+1$ on one of the five OCEAN traits (Openness, Conscientiousness, Extraversion, Agreeableness, Neuroticism). For example, "act wild and crazy" scores $+1$ on Extraversion, while "readily overcome setbacks" scores -1 on Conscientiousness.

We define three synthetic personalities as vectors in $\mathbb{R}^5$: $P1 = (3, 0, 2, 0, -2.5)$, $P2 = (-3, 0, -2, 0, 2.5)$, and $P3 = (0, 2, 0, 2, 0)$, sampled with probabilities 0.3, 0.3, and 0.4. Note that $P2 = -P1$, creating an adversarial pair that is non-identifiable under binary preferences. Phrase rewards are computed as the inner product of the personality vector with the phrase's trait scores. For instance, if a phrase scores $+1$ on Openness, $P1$ assigns a reward of $3 \times 1 = 3$.

Data are generated as follows: (i) sample a personality; (ii) select one of 50 paraphrases of the instruction "Choose the option that resonates most with your personality"; (iii) draw phrases from the MPI dataset; and (iv) simulate preferences using the Bradley–Terry model with either two or three items. Each user contributes exactly one preference pair, ensuring that we reconstruct the adversarial example presented in Section 4.

D.2 Cluster DPO

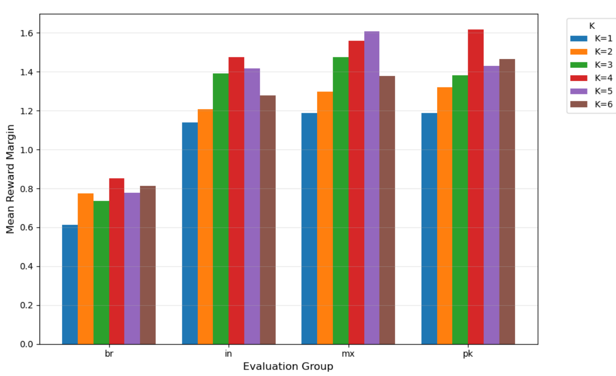
The Cluster-DPO policy is generated as follows: we naively cluster the users into 2 user sub-groups using k-means clustering on the average embedding of all the preferred texts of that user. Embeddings are generated using the RoBERTa-Large model Liu (2019). Then, we train a DPO policy on each cluster separately to get an ensemble of policies; we are essentially replacing the EM-DPO step with a k means clustering step in the proposed algorithm pipeline.

D.3 Additional results

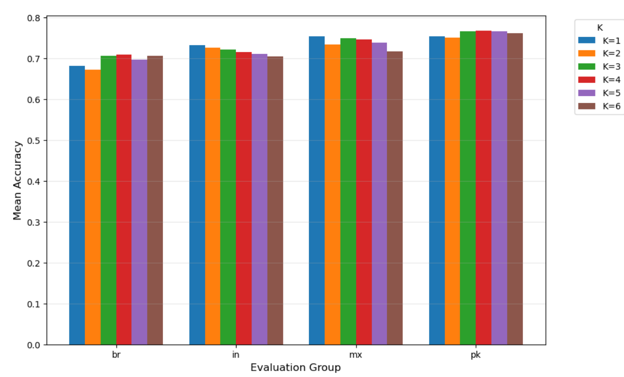
D.3.1 Hyperparameter Tuning

We don't know the exact number of latent sub-groups a priori, so we treat this as a hyper-parameter and tune it using a validation dataset. The validation set would contain extensive preference information from a small set of diverse users. In our case, we generate a dataset from each of the preference group. We then use the same accuracy and reward margins metrics from the experiment section. We plot this in the given table. While the accuracy metric doesn't observe any trends, we observe that the reward margins either peak at $k = 4$ or increase only marginally for $k > 4$ across validation data from any of the component sub-groups for both datasets. Therefore, regardless of the composition of the validation dataset (which is unobservable in practice due to the latent nature of the groups) $k = 4$ emerges as the optimal hyperparameter for this experiment and the two given datasets.

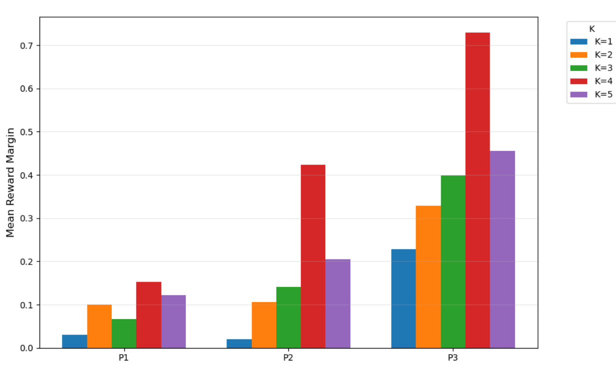
In practical scenarios, in addition to using a validation dataset, we must consider resource availability as well for fixing the number of groups. The memory and compute requirements of EM-DPO scale linearly and at least linearly, respectively, with the number of latent groups. The latter scaling depends on the number of steps required for convergence, which may increase with the number of groups.



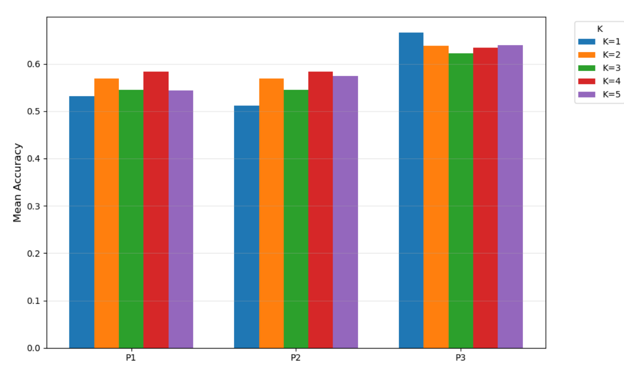
(a) Global Opinion Margin



(b) Global Opinion Accuracy



(c) MPI Margin



(d) MPI Accuracy

Figure 2: Hyper-parameter Tuning Results

D.4 Hyperparameters

The following tables show the hyperparameters for the LLM experiment. We ran the experiments on 8 NVIDIA H100 GPUs with 80GB memory per GPU. On average 1 run of DPO took about 15 mins for the EMDPO algorithm. For the MMRA algorithm, it took roughly 1 hour for the first generation and scoring phase, following by 15 mins for each iteration of the training & evaluation phase.

Parameter	**Value**
Base Model	Mistral 7B v0.3
Batch Size	4
Evaluation Batch Size	16
Learning Rate	5e-7
Gradient Accumulation Steps	1
Max Gradient Norm	10.0
Max Text Length (Prompt + Response)	512
Max Prompt Length	256
No. of Training Epochs	1
No. of Evaluation Examples	256
Optimizer	RMSprop
No. of Warmup Steps for Learning Rate	150
No. of Iterations of the EM Algorithm	5
DPO Beta	0.1

Table 4: Hyperparameters for EMDPO

Parameter	**Value**
Base Model	Mistral 7B v0.3
Batch Size	32
Evaluation Batch Size	32
Learning Rate for Policy	5e-7
Gradient Accumulation Steps	1
Max Gradient Norm	10.0
Max Text Length (Prompt + Response)	512
Max Prompt Length	256
No. of Prompts for Generation	3,072
Completions per Prompt	1
Optimizer	RMSprop
No. of MWU Iterations	20
Batches per MWU Iteration	250
MWU Learning Rate (η)	0.01

Table 5: Hyperparameters for MMRA (Lightweight)