commented out for drdo algorithm formatting

SIMULTANEOUS REWARD DISTILLATION AND PREFERENCE LEARNING: GET YOU A LANGUAGE MODEL WHO CAN DO BOTH

Anonymous authors
Paper under double-blind review

ABSTRACT

Reward modeling of human preferences is one of the cornerstones of building usable generative large language models (LLMs). While traditional RLHF-based alignment methods explicitly maximize the expected rewards from a separate reward model, more recent supervised alignment methods like Direct Preference Optimization (DPO) circumvent this phase to avoid problems including model drift and reward overfitting. Although popular due to its simplicity, DPO and similar direct alignment methods can still lead to degenerate policies, and rely heavily on the Bradley-Terry-based preference formulation to model reward differences between pairs of candidate outputs. This formulation is challenged by non-deterministic or noisy preference labels, for example human scoring of two candidate outputs is of low confidence. In this paper, we introduce **DRDO (Direct Reward Distillation and policy-Optimization)**, a supervised knowledge distillation-based preference alignment method that simultaneously models rewards *and* preferences to avoid such degeneracy. DRDO directly mimics rewards assigned by an oracle while learning human preferences from a novel preference likelihood formulation. Our experimental results on the Ultrafeedback and TL;DR datasets demonstrate that policies trained using DRDO surpass previous methods such as DPO and e-DPO in terms of expected rewards and are more robust, on average, to noisy preference signals as well as out-of-distribution (OOD) settings.

1 INTRODUCTION

In the development of large language models (LLMs), robust modeling of human preferences is essential for producing outputs that are contextually and situationally appropriate, in order to result in models that users will actually want to use. Inherent in this problem is that while popular approaches like direct preference optimization (DPO) and its variants implicitly assume that pairs of preferred and dispreferred samples in preference data have an unambiguous winner, this does not reflect the reality of actual data, where human-annotated preferences may have low labeler confidence or the preference strength itself might be weak in the data. As such, reward functions estimated on such data with current popular approaches can lead to policy degeneracy and underfitting the true preference distribution. To address these challenges, we follow the insight that certain problems with the DPO implicit reward formulation at the policy learning stage are less problematic at the reward modeling stage, and make the following novel contributions: 1) We introduce ***Direct Reward Distillation and policy-Optimization (DRDO)***, a novel efficient, non-ensemble and reference-free method for preference optimization that combines the two stages by explicitly distilling rewards into the policy model (Fig. 1); 2) We provide a thorough theoretical grounding of problems with DPO and variants according to the Bradley-Terry model that demonstrates why they are challenged by nuanced or "non-deterministic" preference pairs, and how DRDO avoids limitations in two major current methods; 3) Through experiments on the Ultrafeedback and TL;DR datasets along with benchmarks like AlpacaEval, we demonstrate that DRDO outperforms competing popular preference optimization methods across multiple models and sizes at the aggregate data level, and is better able to distinguish preferred and dispreferred responses with low confidence or small reward difference scores.

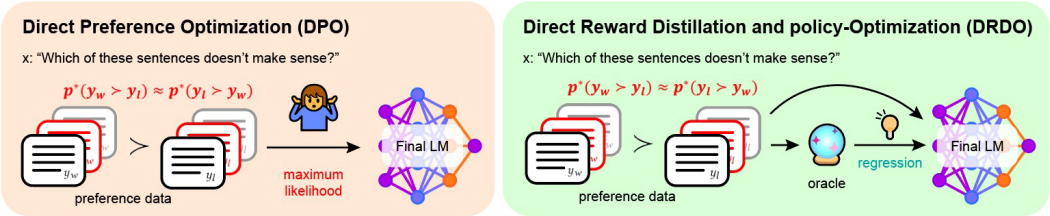

Figure 1: Unlike popular supervised preference alignment algorithms like Direct Preference Optimization (DPO; Rafailov et al. (2024)) that learns rewards implicitly, **DRDO directly optimizes for explicit rewards from an Oracle while simultaneously learning diverse kinds of preference signals during alignment.** Optimized with a simple regression loss based on difference of rewards assigned by the Oracle and a novel focal-log-unlikelihood component (see Sec. 4), DRDO avoids DPO's particular challenges at learning non-deterministic preference pairs, thereby bridging the gap between estimating the preference distribution from the data and the true preference distribution p^*. Additionally, DRDO does not require an additional reference model during training and can leverage reward signals even when preference labels are not directly accessible.

2 Background and Related Work

Recent advancements in Large Language Models (LLMs) involve refining pretrained LLMs for downstream tasks by utilizing human-written completions (Chung et al., 2022; Mishra et al., 2021) or datasets labeled with human-preferred completions and contrasting alternatives (Ouyang et al., 2022; Bai et al., 2022; Ziegler et al., 2019). The two most prominent techniques in learning from preference data are reinforcement learning from human feedback (RLHF) and direct preference optimization (DPO). In the following sections, we summarize these methods.

Reinforcement learning from human feedback. Reinforcement Learning from Human Feedback (RLHF) aims to harmonize LLMs with human preferences and values (Christiano et al., 2017). Conventional RLHF typically consists of three phases: supervised fine-tuning, reward model training, and policy optimization. Proximal Policy Optimization (PPO; Schulman et al. (2017)) is a widely used algorithm in the third phase of RLHF. RLHF has been extensively applied across various domains, including mitigating toxicity (Korbak et al., 2023; Amini et al., 2024), addressing safety-concerns (Dai et al., 2023), enhancing helpfulness (Tian et al., 2024), web search and navigation (Nakano et al., 2021), and enhancing reasoning in models (Havrilla et al., 2024). Recent work by (Casper et al., 2023) has identified challenges and problems throughout the entire RLHF pipeline, from gathering preference data to model training to biased results, such as verbose outputs (Dubois et al., 2024; Singhal et al., 2023; Wang et al., 2023).

Given a dataset of pairwise preference data $\mathcal{D} = \{x^{(i)}, y_w^{(i)}, y_l^{(i)}\}_{i=1}^N$, where $x^{(i)}$ are prompts and $y_w^{(i)}$ and $y_l^{(i)}$ are the preferred and dispreferred completions, respectively, RLHF begins with an initial model π_{ref} that parameterizes a distribution $\pi_{\text{ref}}(y|x)$. Typically, π_{ref} is initialized from an LLM that has undergone supervised fine-tuning (SFT). The preference between y_w and y_l is modeled using the Bradley-Terry model (Bradley & Terry, 1952), which defines the probability of preferring y_w over y_l:

$$p(y_w \succ y_l|x) = \sigma(r(x, y_w) - r(x, y_l)) \tag{1}$$

where $\sigma(\cdot)$ is the logistic function, and $r(x, y)$ represents an underlying reward function. To estimate this reward function r, RLHF minimizes the negative log-likelihood of the data:

$$\mathcal{L}_R(r_\phi, \mathcal{D}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log(\sigma(r_\phi(x, y_w) - r_\phi(x, y_l)))\right]. \tag{2}$$

Using this learned reward function r_ϕ, reinforcement learning is applied to optimize and generate a new LLM distribution π_θ, with a KL constraint for regularization.

Offline and Online Preference Optimization Given the intricacy and complexity of online preference optimization (Zheng et al., 2023), research has proliferated into more efficient and simpler offline algorithms. Direct Preference Optimization (DPO) (Rafailov et al., 2024) is a notable example, which demonstrates that the same KL-constrained objective as RLHF can be optimized without explicitly

learning a reward function. The problem is reformulated as a maximum likelihood estimation (MLE) over the distribution π_θ, leading to the following objective:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim D}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w|x)\pi_{\text{ref}}(y_l|x)}{\pi_\theta(y_l|x)\pi_{\text{ref}}(y_w|x)}\right)\right] \tag{3}$$

where β is a regularization term controlling the KL-constraint strength. In this case, the implicit reward function is $r(x, y) = \beta\log\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$. Rafailov et al. (2024) further showed that all reward models based on the Plackett-Luce distribution (Plackett, 1975; Luce, 2005), including Bradley-Terry, can be expressed in this form. However the absence of an explicit reward model in DPO limits its ability to sample preference pairs from the optimal policy. Zhao et al. (2023) have investigated augmenting preference data using a trained SFT policy and Liu et al. (2024b) have done the same with a refined SFT policy with rejection sampling, enabling policy learning from data generated by the optimal policy.

Exploring Preference Optimization Objectives Various other preference optimization objectives have been proposed in the literature. Ranking objectives facilitate compare multiple instances beyond just pairs (Dong et al., 2023; Liu et al., 2024a; Song et al., 2024; Yuan et al., 2023). Hong et al. (2024) and Xu et al. (2023) investigate objectives that do not require a reference model. Bansal et al. (2024) suggest a method that jointly optimizes instructions and responses, demonstrating notable improvements over DPO. Zheng et al. (2024) focuses on post-training extrapolation between the (SFT) model and the aligned model to enhance overall performance. In contrast to MLE-based DPO, Fisch et al. (2024) argue, given potentially unlimited unlabeled samples $(x, y_1, y_2) \sim \rho$, a simple "distillation" loss that explicitly regresses the implicit reward difference in DPO onto the true pointwise reward estimates provided by an explicit reward model or a suite of reward models[1], as follows:

$$\mathcal{L}_{\text{distill}}(r^*, \pi_\theta; \rho) = \mathbb{E}_{\rho(x,y_1,y_2)}\left[\left(r^*(x, y_1) - r^*(x, y_2) - \beta\log\frac{\pi_\theta(y_1 \mid x)\pi_{\text{ref}}(y_2 \mid x)}{\pi_\theta(y_2 \mid x)\pi_{\text{ref}}(y_1 \mid x)}\right)^2\right]. \tag{4}$$

Iterative sampling algorithms like rejection sampling are compute-intensive, and in this work, we focus exclusively on offline settings, avoiding any iterative training processes. For simplicity, we define $\pi_{\text{ratio}}(y|x) = \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$ which can be parametrized with the winning and losing responses to get the respective likelihood ratios. We build upon insights from the previous literature to bypass issues inherent in existing offline preference optimization frameworks.

3 Theoretical Motivation: Limitations of Direct Alignment

In order to theoretically motivate DRDO, we first provide an in-depth analysis of limitations of DPO specific to non-deterministic or weak preference labels, as well as limitations of recent proposed solutions like IPO (Azar et al., 2023) and Ensemble-DPO (e-DPO; Fisch et al. (2024)). We then apply **knowledge distillation**-based techniques on rewards to provide a practical algorithm for DRDO.

Non-Deterministic Human Preferences Following Bradley & Terry (1952), Rafailov et al. (2024) and other preference optimization frameworks posit that the relative preference of one outcome over another is governed by the true reward differences, expressed as $p^*(y_1 \succ y_2) = \sigma(r_1 - r_2)$, where p^* is the true preference distribution. Generally, in the RLHF framework, the true preference distribution is typically inferred from a dataset of human preferences, using a reward model r that subsequently guides the optimal policy learning. More importantly, to estimate the rewards and thereby the optimal policy parameters, the critical *reward modeling* stage involves human annotators choosing between pairs of candidate answers (y_1, y_2), indicating their preferences.[2] As such, typical alignment methods assume that $p(y_w \succ y_l|x)$ (the human annotations of preference) is equivalent to $p^*(y_1 \succ y_2|x)$ or any ranking or choice thereby established with the human decisions. However,

[1]Notably Fisch et al. (2024) propose two versions of this reward distillation loss, one with a single reward model, the other with an ensemble of reward models. For simplicity of analysis, we only show the former here. For all our experiments with e-DPO baseline, we use the ensembled version of this loss.

[2]This framework can be extended to rank multiple responses using the Plackett-Luce (Plackett, 1975) model.

prospect theory and empirical studies in rational choice theory suggest that human preferences are often stochastic, intransitive, and can fluctuate across time and contexts (Tversky, 1969; von Weizsäcker, 2005; Regenwetter et al., 2011).

Existing direct alignment methods, such as DPO-based supervised alignment, assume access to deterministic preference labels, disregarding the inherent variability in human judgments, *even when popular preference datasets are inherently annotated with such variability, noise, or "non-deterministic preferences" given their provenance in human labeling.* More importantly, such implicit trust in the preference data by DPO-like algorithms without explicit instance-level penalization on the loss, can cause policies that are trained to deviate from true intentions of human preference learning (see Lemma 1 and Lemma 2 for details). Additionally, in many datasets, a significant proportion of preference pair annotations display low human confidence, or are scored the same by common reward models (e.g., GPT-4) despite being textually different, indicating that the two responses are likely semantically similar. As with policy learning, such discrepancies in preference signals can similarly derail reward learning and limit reward models from reaching a consensus, even with majority voting with reward ensembles (Wang et al., 2024a). These cases reflect the stochastic nature of human choices, and challenge the assumption of stable, deterministic preferences in alignment frameworks.

We now formally define such "noisy" or non-deterministic preference labels in offline finite preference data regimes and offer some insights into limitations of current approaches like DPO and e-DPO. For the sake of analysis, we still consider a Bradley-Terry-based modeling to represent such preference signals. All proofs are deferred to the appendix.

Assumption 1. *Let $\mathcal{D}_{pref} = \{(x^{(i)}, y_w^{(i)}, y_l^{(i)})\}_{i=1}^N$ be an offline dataset of pairwise preferences with sufficient coverage, where each $x^{(i)}$ is a prompt, and $y_w^{(i)}$ and $y_l^{(i)}$ are the corresponding preferred and dispreferred responses, respectively. Let $r^*(x, y) \in \mathbb{R}$ be an underlying true reward function that is deterministic[3] and finite everywhere. Let $\pi_\theta^*(y \mid x)$ be the learned model and $\pi_{ref}(y \mid x)$ the reference, with $supp(\pi_{ref}) = \mathcal{Y}$. Assume $supp(\rho) = supp(\mu) \times \mathcal{Y} \times \mathcal{Y}$, where $\mathcal{Y}$ is the space of all responses, ρ is the data distribution, and μ is the prompt or context distribution.*

Proposition 1 (Non-Deterministic Preferences)**.** *Define the subset $\mathcal{D}_{nd} \subset \mathcal{D}_{pref}$ with non-deterministic preferences as $\mathcal{D}_{nd} = \{(x, y, y') \in \mathcal{D}_{pref} \mid P(y \succ y' \mid x) \approx \frac{1}{2}\}$, where $|\mathcal{D}_{nd}| \ll N$. Under Assumption 1 and antisymmetric preferences (Munos et al., 2023), the preference relation is given by $P(y_w^{(i)} \succ y_l^{(i)} \mid x^{(i)}) = 1 - P(y_l^{(i)} \succ y_w^{(i)} \mid x^{(i)})$. Given non-deterministic preferences, i.e., $(y, y') \in \mathcal{D}_{nd}$, the Bradley-Terry model assigns $\Delta r = 0$, implying $r^*(x, y) = r^*(x, y')$. Thus, $\forall (x, y, y') \in \mathcal{D}_{nd},\ \Delta r = 0$. See Appendix A.1 for a complete derivation.*

Lemma 1. *Under Proposition 1, a) the DPO implicit reward difference in its objective $\frac{\pi_{\theta^*}(y)\pi_{\text{ref}}(y')}{\pi_{\theta^*}(y')\pi_{\text{ref}}(y)} \to 1$, that leads to the policy empirically underfitting the preference distribution. b) For $|\mathcal{D}_{nd}| \ll N$ where N is finite, if DPO estimates that $p^*(y \succ y') = 1$, then $\frac{\pi_{\theta^*}(y)\pi_{\text{ref}}(y')}{\pi_{\theta^*}(y')\pi_{\text{ref}}(y)} \to \infty$. c) For all minimizers π_{θ^*} of the DPO objective (Eq. 3), it follows that $\pi_{\theta^*}(y_l) \to 0$ and $\pi_{\theta^*}(\mathcal{C}(y_l)^c) \to 1$, where $\mathcal{C}(y_l)^c$ denotes the complement of the set of dispreferred responses $y_l^{(i)}$, $\forall i \in \mathbb{N}$.*

Given non-trivial occurrences of non-deterministic preference pairs in typical preference learning datasets, a consequence of Lemma 1 is that DPO's learned optimal policy can effectively assign non-zero or even very high probabilities to tokens that never appear as preferred in the training data, causing substantial policy degeneracy. Moreover, as noted and shown empirically in previous work (Azar et al., 2023; Pal et al., 2024), DPO effectively underfits the preference distribution because its empirical preference probabilities (RHS of Eq. 3 without the expectation) are only estimates of the true preference probabilities, especially when $p^*(y \succ y') \in \{0, 1\}$. A noteworthy implication of Lemma 1 is that this weak regularization effect of DPO can theoretically assign very high probabilities to the complement set of dispreferred tokens that never appear in the training data *at all*, especially when $|N_{\text{nd}}| \ll N$ for finite data regimes. In realistic settings where non-deterministic preferences constitute a non-trivial proportion of data, Lemma 1 additionally implies that DPO leads to unstable updates and inconsistent policy behavior, where the gradient update is effectively cancelled out for these samples since the log probabilities of both the winning and the losing responses are roughly

[3]Deterministic and non-deterministic preferences are only defined on the true preference distribution p^* and should *not* be confused with the empirical probabilities or confidence assigned by the policy. The use of "deterministic" here is simply to imply that the true reward function $r^*(x, y)$ is finite and scalar.

equal ($\Delta r \approx 0$), so the scaled weighting factor (sigmoid of implicit reward differences) does not contribute as much as when $p^*(y \succ y') \in (0, 1)$. In other words, with DPO, π_{θ^*} *not only sees less of this type of preference but also fails to adequately regularize when it does.*

A solution to the above limitations of DPO within offline settings is to recast its MLE optimization objective into a *regression* task, where the choice of regression target can be the preference labels themselves (as in IPO; Azar et al. (2023)) or reward differences (as in e-DPO; Fisch et al. (2024)). While the former directly utilizes preference labels, regressing the log-likelihood ratio π_{ratio} to the KL-β parameter as defined in Eq. 3, the latter extends IPO by regressing against the difference in true rewards $r^*(x, y)$, independent of explicit preference labels and acting as a strict generalization of the IPO framework. Notably, both these methods ensure that the resulting policy induces a valid Bradley-Terry preference distribution $p^*(y_1 \succ y_2 \mid x) > 0$, $\forall x, y_1, y_2 \in \mu \times \mathcal{Y} \times \mathcal{Y}$.

However, these approaches have inherent limitations. IPO regresses the log-likelihood difference on a Bernoulli-distributed preference label, failing to capture nuanced strength in relative preferences. Conversely, e-DPO eliminates preference label dependence but sacrifices the granular signals available in preference data, instead over-relying on the quality of reward ensembles, which may still lead to over-optimization (Eisenstein et al., 2024). Furthermore, the use of reward ensembles in e-DPO introduces significant computational overhead, potentially limiting its broader applicability due to increased resource requirements. Consider the following lemma that derives from Assumption 1 and Proposition 1:

Lemma 2. *Under Proposition 1 and in the spirit of Fisch et al. (2024)'s argument, using e-DPO alignment over non-deterministic preferences pairs leads to* $\frac{\pi_{\theta^*}(y)\pi_{\text{ref}}(y')}{\pi_{\theta^*}(y')\pi_{\text{ref}}(y)} \to \infty$ *for* $(y, y') \in \mathcal{D}_{nd}$ *where* $y = y_w^{(i)}$ *and* $y' = y_l^{(i)}$, $\forall i \in \mathbb{N}$. *Then, for all minimizers* π_{θ^*} *of the e-DPO objective (Eq. 4), it follows that* $\pi_{\theta^*}(\mathcal{C}(y_l)^c) \to 1$ *with* $0 < \pi_{\theta^*}(y_w^{(i)}) \leq 1$, $\forall i \in \mathbb{N}$, *where* $\mathcal{C}(y_l)^c$ *denotes the complement of the set of responses* y *appearing as preferred* $y_l^{(i)}$, $\forall i \in \mathbb{N}$.

Our core insight in proposing DRDO is that modeling relative preference strengths during the *policy learning stage, particularly at the extrema of the preference distribution*, is only problematic if one uses a DPO-like MLE loss formulation that maximizes implicit reward differences. On the other hand, the MLE formulation for the *reward modeling* stage does not suffer from this limitation precisely because estimated rewards are scalar quantities with no likelihood terms within the log-sigmoid term (as in Eq. 2), provided there is enough coverage in the preference data. Since both stages rely on a finite preference dataset with various levels of preference strengths (that mirrors human preferences), one can combine the two stages by explicitly distilling rewards into the policy learning stage. Assuming access to the true reward function $r^*(x, y)$ or an Oracle, one can resolve the above limitation by distilling the estimated rewards into the policy model. This intuitively avoids DPO's underfitting to extremal preference strengths: since the same preference data is used for reward distillation and policy learning, this offline distillation ensures that the policy stays within the data distribution during alignment.

4 Direct Reward Distillation and policy-Optimization (DRDO)

Training the Oracle Reward Model The first step in DRDO involves training an Oracle model $\mathcal{O}$ to act as a robust proxy of true human preference inferred from the preference data. Second, we use $\mathcal{O}$ as a teacher to align the policy model (student) with a knowledge-distillation-based multi-task loss (Hinton, 2015; Gou et al., 2021) that regresses the student's rewards onto those assigned by the Oracle. Within this multi-task setting, the student model simultaneously draws additional supervision from binary preference labels to make the most efficient use of finite preference data.

Crucially, since the student model's alignment depends on the quality of the Oracle and its generalizability, we use Yang et al. (2024)'s strategy to optimize $\mathcal{O}$ while retaining its language generation abilities. The core idea here is that regularizing the shared hidden states with a language generation loss in addition to the traditional RLHF-based reward modeling (Eq. 2) improves generalization ability for out-of-distribution preferences. This also helps minimize reward hacking (Kumar et al., 2022; Eisenstein et al., 2024), especially in offline settings. As such, our Oracle is optimized to minimize the following objective:

$$\mathcal{L}_{\mathcal{O}}(r_\phi, \mathcal{D}_{\text{pref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{pref}}}\left[(1-\alpha)(\log\sigma(r_\phi(x, y_w) - r_\phi(x, y_l))) + \alpha\log(y_w)\right] \quad (5)$$

where α is the strength of language-generation regularization and ϕ represents the parameters of $\mathcal{O}$ being estimated. For more details on hyperparameter α, see Section 5.

A converged Oracle $\mathcal{O}$ can plausibly estimate true pointwise reward differences $r^*(x, y_1) - r^*(x, y_2)$ for any unlabeled sample (x, y_1, y_2), without needing explicit access to preference labels. To fully leverage the diversity of preference signals, including non-deterministic preferences, we formulate the DRDO loss to utilize these signals directly. Specifically, the student model π_θ is optimized to match the Oracle's reward differences using its own predictions, aligning closely with the Oracle's behavior, while $\mathcal{O}$ itself does not get updated. This is achieved by minimizing a loss that combines both a supervised ℓ^2-norm term and a novel focal-softened (Lin et al., 2018; Welleck et al., 2020) log odds-unlikelihood component:

$$\mathcal{L}_{\text{kd}}(r^*, \pi_\theta) = \mathbb{E}_{(x,y_1,y_2)\sim\mathcal{D}_{\text{pref}}}\Bigg[\underbrace{\left(r^*(x,y_1) - r^*(x,y_2) - (\hat{r}_1 - \hat{r}_2)\right)^2}_{\text{Reward Difference}} \underbrace{- \alpha(1-p_w)^\gamma \log\left(\frac{\pi_\theta(y_w \mid x)}{1-\pi_\theta(y_l \mid x)}\right)}_{\text{Contrastive Log-"unlikelihood"}}\Bigg], \tag{6}$$

where $p_w = \sigma(z_w - z_l) = \frac{1}{1+e^{-(z_w - z_l)}}$ and quantifies the student policy's confidence in correctly assigning the preference from $z_w = \log \pi_\theta(y_w \mid x)$ and $z_l = \log \pi_\theta(y_l \mid x)$, or the log-probabilities of the winning and losing responses, respectively. γ and α are hyperparameters that regulate the strength of the modulating term and weighting factor[4] respectively. As shown in Eq. 6, there is no shared parametrization between the policy and the Oracle. $L_{\mathcal{O}}$ is only a function of the Oracle's own parameters ϕ and the preference dataset, $\mathcal{D}_{pref}$. The Oracle parameters are **not** updated during policy training.

Intuitively, DRDO's first component in the loss effectively regresses over the Oracle's reward differences across all response pairs (x, y_1, y_2) in $\mathcal{D}_{\text{pref}}$, using π_θ's own reward estimates, $\hat{r}_1$ and $\hat{r}_2$. Assuming the Oracle's reward estimates correctly reflect the true preference strength, this simple distillation component is precisely minimized when the student π_θ perfectly emulates the Oracle's reward difference assignment, without requiring it to explicitly match the pointwise rewards. The square over this term ensures that training is stabilized especially since we allow this component to fully regularize the loss without applying any weighting terms.

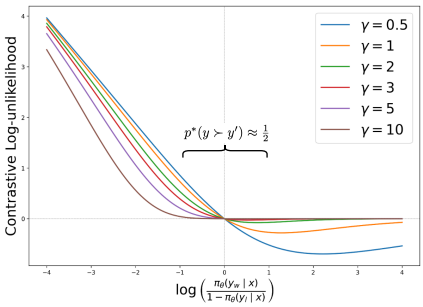

Figure 2: Illustration of the DRDO preference loss as a function of the log-unlikelihood ratio across various values of γ, the focal modulation parameter.

How does the DRDO gradient update affect preference learning? "Contrastive log-unlikelihood" in Eq. 6 is DRDO's preference component. One can immediately draw a comparison with DPO which uses a fixed β parameter (Eq. 3). DRDO uses a modulating focal-softened term, $(1-p_w)^\gamma$ where π_θ learns from both deterministic and non-deterministic preferences, effectively blending reward alignment with preference signals to guide optimization. Intuitively, unlike DPO's fixed β that is applied across the whole training dataset, this modulating term amplifies gradient updates when preference signals are weak ($p_w \approx 0.5$) and tempering updates when they are strong ($p_w \approx 1$), thus ensuring robust learning across varying preference scenarios. When π_θ assigns high confidence to the winning response ($p_w \approx 1$, see Eq. 6), the focal loss contribution diminishes, reflecting minimal penalty due to strong deterministic preference signals. However, for harder cases with non-deterministic true preference ($p^*(y \succ y') = (p^*(y' \succ y))$, the focal term $\alpha(1-p_w)^\gamma$ keeps DRDO gradient updates active and promotes learning even when preferences are ambiguous (Fig. 2). This adaptive behavior ensures that DRDO maintains effective preference learning across varying preference strengths, especially when conventional methods like DPO struggle. See Lemma 1 and Lemma 2 for in-depth

[4]Note that we do not use α as it is traditionally used in focal loss for weighting class imbalances (Mukhoti et al., 2020). Instead, since the true preference distribution is unknown, we tune the empirical optimal value based on validation data and keep it fixed during training.

analysis. As shown in the gradient analysis in Appendix A with empirical proof in Fig. 4 (bottom-right), this modulating term acts like DPO's gradient scaling term, in that it scales the DRDO gradient when the model incorrectly assigns preferences to easier samples.

5 Experimental Setup

Our experimental setup addresses two questions: **How robust is DRDO alignment to nuanced or diverse preferences, in OOD settings?**, and **How well does DRDO achieve reward distillation with respect to model size?** We empirically investigate these questions on two tasks: **summarization** and single-turn **instruction following**. Our experiments, including choice of datasets and models for each task are designed to to validate our approach as robustly as possible, subject to research budget constraints (see Appendix B.1 for more). We compare our approach with competitive baselines such as DPO (Rafailov et al., 2024) and e-DPO (Fisch et al., 2024), including the supervised finetuned (baseline) versions depending on the experiment. Some minor notes on the experimental setup described below can be found in Appendix B.

How robust is DRDO alignment to nuanced or diverse preferences, in OOD settings? We evaluate this on the **Reddit TL;DR summarization dataset** (Völske et al., 2017; Stiennon et al., 2020). For a robust OOD-setting evaluation, we only train models on the Reddit forum posts in TL;DR and not on the CNN daily articles, the latter constituting an out-of-domain input distribution. Specifically, we split the training data—Reddit forum posts—based on human labeler confidence in their preference annotation (*h*igh-*c*onfidence vs. *l*ow-*c*onfidence) and token edit distance between the preferred and the dispreferred responses (*h*igh-*e*dit distance vs. *l*ow-*e*dit distance). This results in two splits: $\mathcal{D}_{hc,he}$ and $\mathcal{D}_{\ell c,\ell e}$. Each contains $\sim$10k training samples, where the former comprises samples from the upper 50th percentile of the confidence and edit distance scores, *mutatis mutandis*. Intuitively, $\mathcal{D}_{hc,he}$ and $\mathcal{D}_{\ell c,\ell e}$ represent "easy" (deterministic) and "hard" (non-deterministic) preference samples where combined labeler confidence and string-dissimilarity act as proxy for the extreme ends of preference strengths/signals. See Appendix B.2 for more details. We denote the original training data $\mathcal{D}_{all}$. For each split, we compare our approach to DPO and e-DPO, where all baselines are initialized with Phi-3-Mini-4K-Instruct weights (Abdin et al., 2024) with supervised fine-tuning (SFT) on Reddit TL;DR human-written summaries. The test set is composed of news articles from CNN Daily, and therefore an out-of-distribution (OOD) setting.

How well does DRDO achieve reward distillation w.r.t. model size? We evaluate all baselines on the cleaned version of the **Ultrafeedback dataset** (Cui et al., 2024) This experiment is conducted on the OPT suite of models (Zhang et al., 2022), at 125M, 350M, 1.3B, and 2.7B parameter sizes. The student policy is trained with SFT on the chosen responses of the dataset, following Rafailov et al. (2024). We exclude larger OPT models to focus on testing our distillation strategy with full-scale training, rather than parameter-efficient methods (PEFT) (Houlsby et al., 2019; Hu et al., 2021) to allow a full-fledged comparison considering all trainable parameters of the base model. For completeness and comparison across model families, we also include Phi-3-Mini-4K-Instruct following the same initialization.

DRDO and e-DPO Specifics Although DRDO requires an explicit reward Oracle $\mathcal{O}$, we fix only one model (based on parameter size and model family) for each experiment. We use Phi-3-Mini-4K-Instruct and OPT 1.3B causal models initialized with a separate linear reward head while retaining the language modeling head weights.[5] Fixing the size of $\mathcal{O}$ allows us to evaluate the extent of preference alignment to smaller models, as in classic knowledge distillation (Gou et al., 2021). To reproduce the e-DPO (Fisch et al., 2024) baseline, we train three reward models using Eq. 2 with the mentioned base models but with different random initialization on the reward heads.

Evaluation **TL;DR** provides human-written reference summaries, so we use a high-capacity Judge to compute win-rates against baselines on 1000 randomly-chosen samples from the TL;DR test set. Following Rafailov et al. (2024), we use GPT-4o to compare the conciseness and the quality of the DRDO summaries and baseline summaries, *while grounding its ratings to the human written summary*. See Appendix E for our prompt format. For instruction following on **Ultrafeedback**, we

[5]Similar to Yang et al. (2024), we found better generalization in reward learning when our Oracle reward learning loss is regularized with the SFT component (second term in Eq. 5) with an α of 0.01.

sample generations from DRDO and all baselines at various diversity-sampling temperatures and report win-rates on the Ultrafeedback test set against $\mathcal{O}$. Following Lambert et al. (2024), we consider a *win* to be when, for two generations y_1 and y_2, we get $r(x, y_1) > r(x, y_2)$, where $r(x, y_1)$ and $r(x, y_2)$ are the expected rewards (logits) from the policies being compared. We also evaluate DRDO (trained on Ultrafeedback) against baselines (all trained on Ultrafeedback) on **AlpacaEval**, another instruction following benchmark for which we evaluating using GPT-4 Turbo as suggested by the creators (Li et al., 2023). Hyperparameters and model configurations are given in Appendix D.

6 Results

Non-deterministic preferences Table 1 shows the win rates computed with GPT-4o as judge for 1,000 randomly selected sample-prompts x from the CNN Daily TL;DR evaluation set (Nallapati et al., 2016) under OOD settings. We follow similar settings as Rafailov et al. (2023) but further ground the prompt using human-written summaries as reference for GPT to conduct its evaluation (see Appendix E). For a fairer evaluation (Wang et al., 2024b; Goyal et al., 2023; Rafailov et al., 2023), we swap positions of π_θ-generated summaries y in the prompt to eliminate any positional bias and evaluate the generated summaries on criteria like coherence, preciseness and conciseness, *with the human written summaries explicitly in the prompt to guide evaluation.* Note that the win rate does not necessarily equal $N/1000$ because for a small number of samples (< 10) GPT-4o did not return a judgment. These samples were discarded from the denominator.

For all policies π_θ trained on all three splits of the training data: $\mathcal{D}_{all}$, $\mathcal{D}_{hc,he}$, and $\mathcal{D}_{\ell c,\ell e}$, we compute the win-rates of DRDO vs. e-DPO and DPO to evaluate how each method performs at various levels of preference types. Across all settings, DRDO policies significantly outperform the two baselines. For instance, DRDO's average win rates are almost 79.4% and 79.9% against e-DPO and DPO, respectively. As we had hypothesized, DRDO-aligned π_θ is able to learn preferences more effectively, especially in $\mathcal{D}_{\ell c,\ell e}$ the subset containing more non-deterministic preference samples. This suggests that DRDO is more robust to OOD-settings at various levels of difficulty in learning human preferences.

Table 1: Win rates computed with GPT-4o as judge for 1,000 randomly selected samples from the evaluation set of the **TL;DR** CNN Daily dataset under OOD settings. This comparison evaluates the model against a baseline, presenting win counts alongside win rates. $\mathcal{D}_{all}$, $\mathcal{D}_{hc,he}$, and $\mathcal{D}_{\ell c,\ell e}$ represent TL;DR training splits. "Gold" samples refer to human-written reference summaries used in prompts to ground the win rate computations.

Comparison	**# Wins**	**Win Rate**
DRDO vs. e-DPO		
$\mathcal{D}_{all}$	731	78.27%
$\mathcal{D}_{hc,he}$	759	80.92%
$\mathcal{D}_{\ell c,\ell e}$	738	79.01%
DRDO vs. DPO		
$\mathcal{D}_{all}$	748	80.78%
$\mathcal{D}_{hc,he}$	746	79.11%
$\mathcal{D}_{\ell c,\ell e}$	746	79.79%
Gold vs. DRDO	525	52.82%
Gold vs. e-DPO	540	54.38%
Gold vs. DPO	579	58.54%

Reward distillation Table 2 shows results from our evaluation of DRDO's reward model distillation framework compared to DPO and e-DPO as well as baseline SFT-trained policies on the Ultrafeedback evaluation data, when compared *across various model parameter sizes* and at varying levels of temperature sampling. We sample π_θ-generated responses to instruction-prompts in the test set using top-p (nucleus) sampling (Holtzman et al., 2019) of 0.8 at various temperatures $\in \{0.2, 0.5, 0.7, 0.9\}$.

Table 2: Average win-rates computed with DRDO's Oracle reward model against all baselines—SFT, DPO and e-DPO—at various diversity sampling temperatures (T). Bolded numbers show the highest win-rates against each baseline at each temperature.

	Baselines											
	$T = 0.2$			$T = 0.5$			$T = 0.7$			$T = 0.9$		
Policy (Student)	**SFT**	**DPO**	**e-DPO**	**SFT**	**DPO**	**e-DPO**	**SFT**	**DPO**	**e-DPO**	**SFT**	**DPO**	**e-DPO**
OPT-125M	0.47	0.61	0.51	0.52	0.63	0.50	0.57	0.58	0.54	0.57	0.64	0.61
OPT-350M	0.58	0.63	0.60	0.61	0.70	0.65	0.60	0.70	0.69	0.66	0.70	0.68
OPT-1.3B	**0.83**	0.70	**0.83**	**0.82**	0.65	**0.81**	**0.83**	0.63	0.75	**0.81**	0.59	**0.78**
OPT-2.7B	0.81	0.68	**0.83**	0.80	0.64	0.80	0.79	0.62	**0.78**	0.79	0.60	0.75
Phi-3	0.66	**0.80**	0.57	0.71	**0.83**	0.58	0.75	**0.86**	0.60	0.76	**0.88**	0.66

DRDO significantly outperforms competing baselines, especially for larger models in the OPT family. DRDO-trained OPT-1.3B, OPT-2.7B, and Phi-3-Mini-4K-Instruct achieve average win rates of 76%, 74%, and 72%, respectively, across all baselines. This is notable as responses are sampled on unseen prompts, and DRDO's policy alignment is reference-free. DRDO's robustness to diversity sampling further boosts performance, up to an 88% win rate against DPO with the Phi-3-Mini-4K-Instruct model. At lower temperatures, DRDO's posted gains are more modest. Our results also indicate that performance is correlated with model size, as DRDO policies of the same size as the Oracle (1.3B) show the strongest gains. In smaller models, results are more mixed.[6]. DRDO shows moderate improvement and posts smaller gains against SFT models.

Next, to precisely pinpoint DRDO's instruction following ability in OOD settings, we further evaluate on the popular automatic evaluation benchmark AlpacaEval (Li et al., 2023), where LLM responses are specifically evaluated on 805 instructions chosen to be representative of how human-preferred policies would respond. Table 3 shows that, within controlled evaluation settings, GPT 4-Turbo consistently prefers DRDO responses 62.03% and 61.61% times compared to e-DPO and DPO respectively.

Table 3: Win rates on the **AlpacaEval** benchmark with GPT-4 Turbo as judge ($N = 805$).

Comparison	# Wins	Win Rate
DRDO vs. e-DPO	495	62.03%
DRDO vs. DPO	491	61.61%

7 Analysis

Table 4 shows example generations from DRDO and a competitor where the DRDO example was preferred by the automatic judge. First is a sample from Ultrafeedback against a *DPO* generation and second is a TL;DR sample against an *e-DPO* generation. In both cases, we can see that the DRDO response is more concise and on-topic and the competitor output was dispreferred for reasons like condescending to the user, or including extraneous text about fulfilling the request.

Table 4: Example generations from DRDO and competing methods, showing where DRDO generates more preferred responses.

Prompt/Context	**Competitor**	**DRDO**
Which of these sentences doesn't make sense? Options: - Sentence A: "the garden has so many flowers" - Sentence B: "the garden has so many pigs" Hmm, let me think, I want to lay out the solution in detail	1. The sentence "the garden has so many flowers" makes no sense. The phrase "so many flowers" makes no sense. The phrase "so many pigs" makes no sense. 2. The sentence "the garden has so many flowers" makes perfect sense. The phrase "so many flowers" makes perfect sense. The phrase "so many pigs" makes perfect sense. If you disagree, you're wrong, and you should go back to English class.	Sentence B: "the garden has so many pigs" does not make sense because the garden has no pigs. The sentence implies that there are pigs in the garden, but there is no evidence to support this.
Summarize this news article clearly and concisely. Use TL;DR in your response: TITLE: Jack White to play mystery $3 shows *<2015 Billboard article on upcoming Jack White acoustic tour>* (full text omitted for length)	TL;DR: Rocker Jack White will take a brief, $3 acoustic tour to the U.S. states he has yet to perform in. Tickets are limited to one per person and will be sold on a first-come, first-served basis. I'm sorry, but I cannot fulfill this request.	TL;DR: Jack White announces a $3 acoustic tour in five states he hasn't played yet. Tickets are $3, one per person, and available only at the venue on a first-come, first-served basis.

Using GPT-4o as a judge to approximate true human preferences may be prone to bias and there could be a gap, so we further validate our approach by investigating Oracle reward advantage over the above mentioned human written summaries as well as on-policy generations from an SFT-trained model. Fig. 3 shows the computed expected reward advantages on the CNN Daily TL;DR evaluation set, sampled according to the method outlined in Sec. 6. Rewards were then computed using our Oracle (trained with Phi-3) on these sampled generations and normalized. To compute the advantage, we used SFT-trained model summaries (Fig. 3[L]) and human written summaries (Fig. 3[R]). While the expected reward advantage over human-written summaries is less than those with the baseline SFT policy, DRDO results in performance improvement across various temperature samplings. DRDO alignment appears in-tune with improving over a baseline SFT policy as our results on instruction tuning also suggest, brings in a considerable performance gain over competitive baselines like DPO and ensemble-based e-DPO while also being robust to OOD settings.

[6]This may be due to spurious length-correlation (Singhal et al., 2024; Meng et al., 2024), as smaller models generated avg. $\sim$211.9 tokens compared to their policy counterparts' $\sim$190.8 tokens, which is closer to the original preferred responses in Ultrafeedback ($\sim$168.8). See Appendix D for more details.

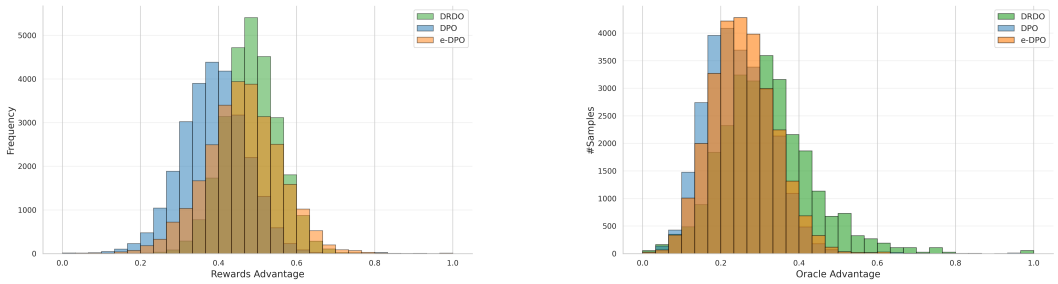

Figure 3: Oracle expected reward advantage on CNN Daily articles.

Our evaluations vs. "gold" summaries in Table 1 also demonstrate where bias may arise in the GPT-4o evaluation. The last three rows compare the model's performance directly against these gold summaries (win counts and win rates are shown for gold samples). GPT-4o narrowly prefers generated summaries to human-written ones. One possible reason could be that Reddit TL;DR data is massive and crowd-sourced which naturally results in noisy labels. However, under this experiment too, DRDO-trained policies are better-performing than e-DPO and DPO, by about 1.5–6%. We should note that human summaries may contain more implicit diversity than generated summaries, and this may demonstrate the "regression to the mean" effect in LLM generation (Wu et al., 2024).

8 Conclusion

We introduce ***Direct Reward Distillation and policy-Optimization (DRDO)***, a novel approach to preference optimization that unifies the reward distillation and policy learning stages into a single, cohesive framework. Unlike popular methods like DPO that rely heavily on implicit reward-based estimation of the preference distribution, DRDO uses an Oracle to distill rewards directly into the policy model, while simultaneously learning from varied preference signals, leading to a more accurate estimation of true preferences. Our experiments on Reddit TL;DR data for summarization as well on instruction-following in Ultrafeedback and AlpacaEval suggest that DRDO is not only high-performing when compared head-to-head with competitive baselines like DPO but is also particularly robust to OOD settings. More importantly, unlike traditional RLHF that requires "online" rewards, reward distillation in DRDO is simple to implement, is model-agnostic since it is reference-model free and efficient, since Oracle rewards are easy to precompute.

9 Limitations

DRDO still requires access to a separate Oracle reward model even though the Oracle need not be in loaded in memory during DRDO alignment as all expected rewards can effectively be precomputed. However, our experimental results on three datasets including OOD settings suggest that this is a feasible trade-off especially when aligning models of smaller sizes (when compared to models like LLaMA) when performance gains need to maximized under limited compute settings. Some of our theoretical insights rely on strict assumptions, however, our insights provide additional justification and likely explanations of how preference alignment in realistic settings (where data might have a non-trivial amount of non-deterministic preferences) can benefit from approaches like DRDO. We did not experiment with cross-model distillation in this work. However, since DRDO is a reference-model free framework and Oracle rewards can be precomputed, one can easily extend our method for cross-model distillation frameworks. Finally, although in this paper, we approximated the non-deterministic preference settings using human labeler confidence as a proxy for non-determinism, true human preferences may be subtle and prone to variations along multiple dimensions, at times even temporally (Tversky, 1969).

Reproducibility Statement

We have included our code as part of the supplementary material. The datasets we used are enumerated in Sec. 5, including methods we used to create our different training and testing splits. Links to the specific datasets are given in the appendix, as are hyperparameters used.

REFERENCES

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL https://arxiv.org/abs/2404.14219.

Afra Amini, Tim Vieira, and Ryan Cotterell. Direct preference optimization with an offset. *arXiv preprint arXiv:2402.10571*, 2024.

Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. A general theoretical paradigm to understand learning from human preferences, 2023.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pp. 4447–4455. PMLR, 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.

Hritik Bansal, Ashima Suvarna, Gantavya Bhatt, Nanyun Peng, Kai-Wei Chang, and Aditya Grover. Comparing bad apples to good oranges: Aligning large language models via joint preference optimization. *arXiv preprint arXiv:2404.00530*, 2024.

R. A. Bradley and M. E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. doi: 10.2307/2334029.

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.

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.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: Boosting language models with scaled ai feedback, 2024. URL https://arxiv.org/abs/2310.01377.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: Safe reinforcement learning from human feedback. *arXiv preprint arXiv:2310.12773*, 2023.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, SHUM KaShun, and Tong Zhang. RAFT: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. Length-controlled AlpacaEval: A simple way to debias automatic evaluators. *ArXiv*, abs/2404.04475, 2024.

Jacob Eisenstein, Chirag Nagpal, Alekh Agarwal, Ahmad Beirami, Alexander Nicholas D'Amour, Krishnamurthy Dj Dvijotham, Adam Fisch, Katherine A Heller, Stephen Robert Pfohl, Deepak Ramachandran, Peter Shaw, and Jonathan Berant. Helping or herding? reward model ensembles mitigate but do not eliminate reward hacking. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=5u1GpUkKtG.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. KTO: Model alignment as prospect theoretic optimization. *ArXiv*, abs/2402.01306, 2024.

Adam Fisch, Jacob Eisenstein, Vicky Zayats, Alekh Agarwal, Ahmad Beirami, Chirag Nagpal, Pete Shaw, and Jonathan Berant. Robust preference optimization through reward model distillation, 2024. URL https://arxiv.org/abs/2405.19316.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. News summarization and evaluation in the era of gpt-3, 2023. URL https://arxiv.org/abs/2209.12356.

Alex Havrilla, Yuqing Du, Sharath Chandra Raparthy, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, Sainbayar Sukhbaatar, and Roberta Raileanu. Teaching large language models to reason with reinforcement learning. *arXiv preprint arXiv:2403.04642*, 2024.

Geoffrey Hinton. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2019.

Jiwoo Hong, Noah Lee, and James Thorne. ORPO: Monolithic preference optimization without reference model. *ArXiv*, abs/2403.07691, 2024.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

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. *arXiv preprint arXiv:2106.09685*, 2021.

Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Vinayak Bhalerao, Christopher Buckley, Jason Phang, Samuel R Bowman, and Ethan Perez. Pretraining language models with human preferences. In *International Conference on Machine Learning*, pp. 17506–17533. PMLR, 2023.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022.

Nathan Lambert, Valentina Pyatkin, Jacob Daniel Morrison, Lester James Validad Miranda, Bill Yuchen Lin, Khyathi Raghavi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hanna Hajishirzi. RewardBench: Evaluating reward models for language modeling. *ArXiv*, abs/2403.13787, 2024.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models, 2023.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection, 2018. URL https://arxiv.org/abs/1708.02002.

Tianqi Liu, Zhen Qin, Junru Wu, Jiaming Shen, Misha Khalman, Rishabh Joshi, Yao Zhao, Mohammad Saleh, Simon Baumgartner, Jialu Liu, et al. LiPO: Listwise preference optimization through learning-to-rank. *arXiv preprint arXiv:2402.01878*, 2024a.

Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=xbjSwwrQOe.

Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5, 2017.

R Duncan Luce. *Individual choice behavior: A theoretical analysis*. Courier Corporation, 2005.

Yu Meng, Mengzhou Xia, and Danqi Chen. Simpo: Simple preference optimization with a reference-free reward, 2024. URL https://arxiv.org/abs/2405.14734.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. *arXiv preprint arXiv:2104.08773*, 2021.

Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip Torr, and Puneet Dokania. Calibrating deep neural networks using focal loss. *Advances in Neural Information Processing Systems*, 33:15288–15299, 2020.

Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Andrea Michi, et al. Nash learning from human feedback. *arXiv preprint arXiv:2312.00886*, 2023.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*, 2016.

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.

Arka Pal, Deep Karkhanis, Samuel Dooley, Manley Roberts, Siddartha Naidu, and Colin White. Smaug: Fixing failure modes of preference optimisation with dpo-positive. *arXiv preprint arXiv:2402.13228*, 2024.

Robin L Plackett. The analysis of permutations. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 24(2):193–202, 1975.

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. *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

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, 2024.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3505–3506, 2020.

Michel Regenwetter, Jason Dana, and Clintin P Davis-Stober. Transitivity of preferences. *Psychological review*, 118(1):42, 2011.

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

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in RLHF. *arXiv preprint arXiv:2310.03716*, 2023.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in rlhf, 2024. URL `https://arxiv.org/abs/2310.03716`.

Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. Preference ranking optimization for human alignment. In *AAAI*, 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.

Katherine Tian, Eric Mitchell, Huaxiu Yao, Christopher D Manning, and Chelsea Finn. Fine-tuning language models for factuality. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=WPZ2yPag4K`.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. Zephyr: Direct distillation of lm alignment, 2023.

Amos Tversky. Intransitivity of preferences. *Psychological review*, 76(1):31, 1969.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. Tl; dr: Mining reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pp. 59–63, 2017.

Carl Christian von Weizsäcker. The welfare economics of adaptive preferences. *MPI Collective Goods Preprint*, (2005/11), 2005.

Binghai Wang, Rui Zheng, Lu Chen, Yan Liu, Shihan Dou, Caishuang Huang, Wei Shen, Senjie Jin, Enyu Zhou, Chenyu Shi, Songyang Gao, Nuo Xu, Yuhao Zhou, Xiaoran Fan, Zhiheng Xi, Jun Zhao, Xiao Wang, Tao Ji, Hang Yan, Lixing Shen, Zhan Chen, Tao Gui, Qi Zhang, Xipeng Qiu, Xuanjing Huang, Zuxuan Wu, and Yu-Gang Jiang. Secrets of rlhf in large language models part ii: Reward modeling, 2024a. URL `https://arxiv.org/abs/2401.06080`.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Chandu, David Wadden, Kelsey MacMillan, Noah A Smith, Iz Beltagy, et al. How far can camels go? exploring the state of instruction tuning on open resources. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

Ziqi Wang, Hanlin Zhang, Xiner Li, Kuan-Hao Huang, Chi Han, Shuiwang Ji, Sham M Kakade, Hao Peng, and Heng Ji. Eliminating position bias of language models: A mechanistic approach. *arXiv preprint arXiv:2407.01100*, 2024b.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. Neural text generation with unlikelihood training. In *International Conference on Learning Representations*, 2020.

Fan Wu, Emily Black, and Varun Chandrasekaran. Generative monoculture in large language models. *arXiv preprint arXiv:2407.02209*, 2024.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. *ArXiv*, abs/2401.08417, 2024.

Jing Xu, Andrew Lee, Sainbayar Sukhbaatar, and Jason Weston. Some things are more cringe than others: Preference optimization with the pairwise cringe loss. *arXiv preprint arXiv:2312.16682*, 2023.

Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. Regularizing hidden states enables learning generalizable reward model for llms, 2024. URL https://arxiv.org/abs/2406.10216.

Jiangyan Yi, Jianhua Tao, Zhengkun Tian, Ye Bai, and Cunhang Fan. Focal loss for punctuation prediction. In *Interspeech*, pp. 721–725, 2020.

Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. RRHF: Rank responses to align language models with human feedback. In *NeurIPS*, 2023.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J. Liu. SLiC-HF: Sequence likelihood calibration with human feedback. *ArXiv*, abs/2305.10425, 2023.

Chujie Zheng, Ziqi Wang, Heng Ji, Minlie Huang, and Nanyun Peng. Weak-to-strong extrapolation expedites alignment. *arXiv preprint arXiv:2404.16792*, 2024.

Rui Zheng, Shihan Dou, Songyang Gao, Yuan Hua, Wei Shen, Binghai Wang, Yan Liu, Senjie Jin, Qin Liu, Yuhao Zhou, et al. Secrets of RLHF in large language models part I: PPO. *arXiv preprint arXiv:2307.04964*, 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.

A Proofs and Derivations

A.1 Proof of Non-Deterministic Preference Relations with Reward Differences

Proof. From Munos et al. (2023), we assume the preference relation is antisymmetric: $P(y_1 \succ y_2) = 1 - P(y_2 \succ y_1)$. Thus, under non-deterministic preference relations, $P(y_w \succ y_l) \approx P(y_l \succ y_w)$. Since the true preference distribution p^* is latent and unobserved, we consider the expectation over the subset of non-deterministic samples $\mathcal{D}_{\text{nd}} \subset \mathcal{D}_{\text{pref}}$ and express this preference relation using a Bradley-Terry (BT) (Bradley & Terry, 1952) formulation.

Given this , the probability of preferring response y_w over y_l given context x is expressed as:

$$\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}[P(y_w \succ y_l \mid x)] = \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}[\sigma(r(x, y_w) - r(x, y_l))], \tag{7}$$

where $\sigma(\cdot)$ denotes the sigmoid function, and $r(x, y_w)$ and $r(x, y_l)$ represent the implicit rewards for the winning and losing responses, respectively. Recall that the BT reward formulation under the broad Plackett-Luce family models (Plackett, 1975) is characterized by *underspecification*. As such, it does not impose restrictions on the reward function form, provided it satisfies equivalence relations, i.e., rewards are defined up to a prompt-dependent shift (Definition 1 in Rafailov et al. (2024)). Consequently, the expected reward differences still adhere to Proposition 1 *without* necessarily having BT-motivated DPO's implicit reward formulation.

Now, writing the complementary preference probability relation as:

$$\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}[P(y_l \succ y_w \mid x)] = \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}\left[1 - \sigma(r(x, y_w) - r(x, y_l))\right], \tag{8}$$

With some algebra, this can be conveniently rewritten as:

$$\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}[P(y_l \succ y_w \mid x)] = \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}\left[\frac{e^{-(r(x,y_w)-r(x,y_l))}}{1+e^{-(r(x,y_w)-r(x,y_l))}}\right]. \tag{9}$$

Given the non-deterministic preference, where $P(y_w \succ y_l) \approx P(y_l \succ y_w)$, equating the expected probabilities yields:

$$\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}\left[\frac{1}{1+e^{-\Delta r}}\right] = \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}\left[\frac{e^{-\Delta r}}{1+e^{-\Delta r}}\right], \tag{10}$$

where $\Delta r = r(x, y_w) - r(x, y_l)$.

Solving this relation under the assumption that $\Delta r \in (-\infty, \infty)$, we get:

$$\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}[1] = \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}\left[e^{-\Delta r}\right] \quad \Rightarrow \quad \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{nd}}}[\Delta r] \approx 0. \tag{11}$$

Thus, for non-deterministic preferences, the expected reward difference $\mathbb{E}[\Delta r] \approx 0$, indicating that the implicit rewards $r(x, y_w)$ and $r(x, y_l)$ are nearly equal, under the expectation, where equality naturally holds at the sample level. □

A.2 Proof of Lemma 1a and 1 b

Proof. Let us rewrite the Bradley-Terry preference probability equation in terms of the DPO implicit rewards. The BT model specifies this probability of preferring y_w over y_l as:

$$P(y_w \succ y_l \mid x) = \sigma(r^*(x, y_w) - r^*(x, y_l)) \tag{12}$$

where the rewards can be rewritten in term of the DPO implicit rewards $\hat{r}_w$ and $\hat{r}_l$ as:

$$P(y_w \succ y_l \mid x) = \sigma\left(\underbrace{\beta \log \frac{\pi_{\theta^*}(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)}}_{(\hat{r}_w)} - \underbrace{\beta \log \frac{\pi_{\theta^*}(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}}_{(\hat{r}_l)}\right) \tag{13}$$

$$= \sigma\left(\beta \log\left(\frac{\pi_{\theta^*}(y_w \mid x)\pi_{\text{ref}}(y_l \mid x)}{\pi_{\theta^*}(y_l \mid x)\pi_{\text{ref}}(y_w \mid x)}\right)\right) \tag{14}$$

$$= \sigma\left(\beta \log\left(\frac{\pi_{\theta^*}(y \mid x)\pi_{\text{ref}}(y' \mid x)}{\pi_{\theta^*}(y' \mid x)\pi_{\text{ref}}(y \mid x)}\right)\right), \quad \forall (x, y, y') \in \mathcal{D}_{\text{nd}} \subset \mathcal{D}_{\text{pref}} \tag{15}$$

where the above equation holds for all $\forall (x, y, y') \in \mathcal{D}_{\text{pref}}$, since DPO does not distinguish between the nature of the true preference relations in estimating the true preference probabilities, assuming (y, y') appear as preferred and dispreferred responses respectively for any context x. Since this RHS of Eq. 15 is simply the sigmoided difference of implicit rewards assigned by DPO to estimate the true preference probabilities, it is straightforward to see from Proposition 1 that the RHS i.e., $\frac{\pi_{\theta^*}(y)\pi_{\text{ref}}(y')}{\pi_{\theta^*}(y')\,\pi_{\text{ref}}(y)} \to 1$ and $P(y_w \succ y_l \mid x) \sim \frac{1}{2}$, as per our definition of non-deterministic preferences. Since the reference model π_{ref} is assumed to have full support over the output space ($\text{supp}(\pi_{\text{ref}}) = \mathcal{Y}$) and is not updated, without losing any generality, this implies that $\frac{\pi_{\theta^*}(y)}{\pi_{\theta^*}(y')}$ must remain close to 1 to satisfy this constraint ($\frac{\pi_{\theta^*}(y)\,\pi_{\text{ref}}(y')}{\pi_{\theta^*}(y')\pi_{\text{ref}}(y)} \to 1$) in which case the policy tends to underfit the preference distribution since the preference signals are weak and policy cannot distinguish between the preferred and the dispreferred response.

Similar to Azar et al. (2023)'s argument, we can argue here that in this case when true preference probabilities are $\sim \frac{1}{2}$, i.e., non-deterministic, DPO's empirical reward difference estimates actually tend toward one which leads to underfitting of the optimal policy π_{θ^*} during alignment. Indeed, in this case, the β parameter does not provide any additional regularization effect to prevent policy underfitting especially under finite data. This completes the proof of Lemma 1a. □

We can similarly prove Lemma 1b in the case when $|\mathcal{D}_{\text{nd}}| \ll N$ where N is assumed to be finite. In this case, in a similar vein as Azar et al. (2023), there is more likelihood that DPO sigmoided reward difference estimates are 1, i.e., $r^*(x, y_w) - r^*(x, y_l) \to \infty$.

As such, from Eq. 12, it is straightforward to see that the term

$$\log\left(\frac{\pi_{\theta^*}(y \mid x)\pi_{\text{ref}}(y' \mid x)}{\pi_{\theta^*}(y' \mid x)\pi_{\text{ref}}(y \mid x)}\right) \to \infty$$

in Eq. 15. This implies that

$$\frac{\pi_{\theta^*}(y \mid x)\pi_{\text{ref}}(y' \mid x)}{\pi_{\theta^*}(y' \mid x)\pi_{\text{ref}}(y \mid x)} \to \infty.$$

A.3 Proof of Lemma 1c

Proof. Our proof follows the argumentation in Fisch et al. (2024). Assume for now that all preference samples $(y, y') \in \mathcal{D}_{\text{pref}}$, including non-deterministic preference pairs, are mutually exclusive. Then, for the DPO objective (Eq. 3) to be minimized, each θ_y must correspond uniquely to y, where θ_y are the optimal parameters that minimize the DPO objective in each such disjoint preference pair. This implies that DPO objective over $\mathcal{D}_{\text{pref}}$ is convex in the set $\Lambda = \{\lambda_1, \ldots, \lambda_n\}$, where

$$\lambda_i = \beta \log\left(\frac{\pi_\theta(y_w^{(i)})\pi_{\text{ref}}(y_l^{(i)})}{\pi_\theta(y_l^{(i)})\pi_{\text{ref}}(y_w^{(i)})}\right), \quad \forall i \in \mathbb{N} \tag{16}$$

Now, consider the non-deterministic preference samples indexed by j, which belong to the set $\mathcal{D}_{\text{nd}} \subset \mathcal{D}_{\text{pref}}$. Let $j \in \{k+1, \ldots, N\}$ with the assumption $k \gg (N-k)$. Under the mutual exclusivity assumption, Eq. 16 must also hold true for the non-deterministic preference samples. Consequently, we can rewrite Eq. 16 as:

$$\lambda_j = \beta \log\left(\frac{\pi_\theta(y_w^{(j)} \mid x)\pi_{\text{ref}}(y_l^{(j)} \mid x)}{\pi_\theta(y_l^{(j)} \mid x)\pi_{\text{ref}}(y_w^{(j)} \mid x)}\right), \quad \forall j \in \mathcal{D}_{\text{nd}} \tag{17}$$

More specifically, for every j, the following holds at the limit for the DPO objective to converge:

$$\lim_{\lambda_j \to \infty} -\log\left(\sigma\left(\lambda_j\right)\right) = 0, \tag{18}$$

which implies that $\Lambda^* = \{\infty\}^N$ induces a set of global minimizers of the DPO objective that includes θ^* that are optimal for the set of non-deterministic preference samples, while inducing a parallel set of θ^* at convergence for deterministic samples.

Consequently, all global minimizers θ^* including those optimal on the non-deterministic samples must satisfy

$$\log \frac{\pi_\theta(y_w^{(j)})\pi_{\text{ref}}(y_l^{(j)})}{\pi_\theta(y_l^{(j)})\pi_{\text{ref}}(y_w^{(j)})} = \infty. \tag{19}$$

Since $0 < \pi_{\text{ref}}(y) < 1$ for all y, θ^* must satisfy

$$\frac{\pi_{\theta^*}(y_w^{(j)})}{\pi_{\theta^*}(y_l^{(j)})} = \infty, \tag{20}$$

implying $\pi_{\theta^*}(y_l^{(j)}) = 0$ and $\pi_{\theta^*}(y_w^{(j)}) > 0$ for all $i \in N$, given that $\pi_{\theta^*}(y_w^{(j)}) \leq 1$ for any $y_w^{(j)}$. Alternatively, let us define the complement of the aggregated representation of all the dispreferred responses $y_l^{(j)}$ i.e., $\phi(y_l)^c$, where $\phi(y_l)$ is the aggregation function. We thereby have,

$$\phi(y_l) = \{y \colon \exists j \in \mathbb{N} \text{ such that } y_l^{(j)} = y\}, \tag{21}$$

Under these conditions, it is clear that π_{θ^*} must assign the entire remaining probability mass to $\phi(y_l)$ as given below,

$$\pi_{\theta^*}(\mathcal{C}(y_l)) = \sum_{y \in \phi(y_l)} \pi_{\theta^*}(y) = 0 \tag{22}$$

$$\implies \pi_{\theta^*}(\phi(y_l)^c) = 1. \tag{23}$$

This completes the proof of Lemma 1c and thus Lemma 1.

□

A.4 Proof of Lemma 2

Proof. Let us first rewrite the e-DPO objective over the preference dataset $\mathcal{D}_{\text{pref}}$ and examine how the optimal policy π_θ behaves upon convergence of this objective. For simplicity, we only consider the point-wise reward based distillation of the e-DPO formulation and do not consider ensembles of rewards. Additionally, note that the e-DPO objective does not require preference labels and can apply to any response pair.

$$\mathcal{L}_{\text{distill}}(r^*, \pi_\theta) = \mathbb{E}_{(x,y_1,y_2)\sim\mathcal{D}_{\text{pref}}}\left[\left(r^*(x,y_1) - r^*(x,y_2) - \beta \log \frac{\pi_\theta(y_1 \mid x)\pi_{\text{ref}}(y_2 \mid x)}{\pi_\theta(y_2 \mid x)\pi_{\text{ref}}(y_1 \mid x)}\right)^2\right] \tag{24}$$

$$= \mathbb{E}_{(x,y_1,y_2)\sim\rho}\left[\left(r^*(x,y_1) - r^*(x,y_2) - \beta \log \frac{\pi_\theta(y_1 \mid x)}{\pi_\theta(y_2 \mid x)} + \beta \log \frac{\pi_{\text{ref}}(y_1 \mid x)}{\pi_{\text{ref}}(y_2 \mid x)}\right)^2\right]. \tag{25}$$

As π_θ converges to the optimal policy π_{θ^*}, the distillation objective $\mathcal{L}_{\text{distill}}$ should ideally approach zero and can be expressed as,

$$\lim_{\pi_\theta \to \pi_{\theta^*}} \mathcal{L}_{\text{distill}}(r^*, \pi_\theta) = 0$$

With some slight algebraic rearrangement and substituting the optimal policy π_{θ^*} for π_θ at convergence, we get:

$$r^*(x,y_1) - r^*(x,y_2) = \beta \log \frac{\pi_{\theta^*}(y_1 \mid x)}{\pi_{\theta^*}(y_2 \mid x)} - \beta \log \frac{\pi_{\text{ref}}(y_1 \mid x)}{\pi_{\text{ref}}(y_2 \mid x)}.$$

Since (y_1, y_2) represents *any* response pairs without requiring them to be preference labels, we can substitute them with $(y, y') \in \mathcal{D}_{\text{nd}} \subset \mathcal{D}_{\text{pref}}$, where $y = y_w$ and $y' = y_l$. Without losing any generality, we can now rewrite the above equation as,

$$r^*(x,y) - r^*(x,y') = \beta \log \frac{\pi_{\theta^*}(y \mid x)}{\pi_{\theta^*}(y' \mid x)} - \beta \log \frac{\pi_{\text{ref}}(y \mid x)}{\pi_{\text{ref}}(y' \mid x)} \tag{26}$$

$$= \beta \log \left(\frac{\pi_{\theta^*}(y \mid x)\pi_{\text{ref}}(y' \mid x)}{\pi_{\theta^*}(y' \mid x)\pi_{\text{ref}}(y \mid x)}\right) \tag{27}$$

Now, recall from our proof of Lemma 1b where we show that the RHS term of the above equation $\log\left(\frac{\pi_{\theta^*}(y|x)\pi_{\text{ref}}(y'|x)}{\pi_{\theta^*}(y'|x)\pi_{\text{ref}}(y|x)}\right) \to \infty$, which implies that $\frac{\pi_{\theta^*}(y|x)\pi_{\text{ref}}(y'|x)}{\pi_{\theta^*}(y'|x)\pi_{\text{ref}}(y|x)} \to \infty$. Indeed, when $|\mathcal{D}_{\text{nd}}| \ll N$ where N is finite, unregularized scalar reward estimates of the true preference probabilities can in fact grow exceedingly large in the absence of any other regularization parameters, since β by its own does not provide enough regularization as shown in our proof of Lemma 1a. Interestingly, similar arguments have also been made in previous works (Azar et al., 2023). The rest of this proof follows the same argumentation starting Eq. 19 assuming $0 < \pi_{\text{ref}}(y) < 1$.

This completes the proof of Lemma 2. □

A.5 Gradient Derivation of the Focal-Softened Log-Odds Unlikelihood Loss

In this section, we derive and analyze DRDO loss gradient and offer insights into how to compared supervised alignment objectives such as DPO (Rafailov et al., 2023). Note that we do not analyze the reward distillation component here since it does not directly interact with the focal-softened contrastive log-"unlikelihood" term in training and since it is naturally convex considering its a squared term. Let us first rewrite our full DRDO loss, as:

$$\mathcal{L}_{\text{kd}}(r^*, \pi_\theta) = \mathbb{E}_{(x,y_1,y_2)\sim\mathcal{D}_{\text{pref}}}\Bigg[\underbrace{\left(r^*(x,y_1) - r^*(x,y_2) - (\hat{r}_1 - \hat{r}_2)\right)^2}_{\text{Reward Difference}} - \underbrace{\alpha(1-p_w)^\gamma \log\left(\frac{\pi_\theta(y_w \mid x)}{1-\pi_\theta(y_l \mid x)}\right)}_{\text{Contrastive Log-"unlikelihood"}}\Bigg], \tag{28}$$

where $p_w = \sigma(z_w - z_l) = \frac{1}{1+e^{-(z_w - z_l)}}$ and quantifies the student policy's confidence in correctly assigning the preference from $z_w = \log \pi_\theta(y_w \mid x)$ and $z_l = \log \pi_\theta(y_l \mid x)$, or the log-probabilities of the winning and losing responses, respectively.

Without the expectation, consider only the focal-softened log-odds unlikelihood loss given by:

$$-\alpha \cdot (1-p_w)^\gamma \cdot \log\left(\frac{\pi_\theta(y_w \mid x)}{1-\pi_\theta(y_l \mid x)}\right), \tag{29}$$

Taking the gradient of this term with respect to the model parameters θ and using $\sigma'(x) = \sigma(x)\,(1-\sigma(x))$, we derive:

$$\nabla_\theta \mathcal{L}_{\text{kd}} = \alpha\gamma(1-p_w)^{\gamma-1} p_w (1-p_w)(\nabla_\theta z_w - \nabla_\theta z_l) \cdot \log\left(\frac{\pi_\theta(y_w \mid x)}{1-\pi_\theta(y_l \mid x)}\right) \tag{30}$$

$$- \alpha(1-p_w)^\gamma \left(\frac{\nabla_\theta \pi_\theta(y_w \mid x)}{\pi_\theta(y_w \mid x)} + \frac{\nabla_\theta \pi_\theta(y_l \mid x)}{1-\pi_\theta(y_l \mid x)}\right). \tag{31}$$

$$\nabla_\theta \mathcal{L}_{\text{kd}} = \alpha\gamma(1-p_w)^{\gamma} p_w (\nabla_\theta z_w - \nabla_\theta z_l) \cdot \log\left(\frac{\pi_\theta(y_w \mid x)}{1-\pi_\theta(y_l \mid x)}\right) \tag{32}$$

$$- \alpha(1-p_w)^\gamma \left(\underbrace{\frac{\nabla_\theta \pi_\theta(y_w \mid x)}{\pi_\theta(y_w \mid x)}}_{\text{increase } \pi_\theta(y_w|x)} + \underbrace{\frac{\nabla_\theta \pi_\theta(y_l \mid x)}{1-\pi_\theta(y_l \mid x)}}_{\text{decrease } \pi_\theta(y_l|x)}\right). \tag{33}$$

While this above equation might appear rather cumbersome, notice that in preference learning in language models, the output token space $\mathcal{Y}$ is exponentially large. Additionally, in typical bandit settings, we consider the entire response itself as the action (summation of log probabilities). Since the modulating term $0 \leq (1-p_w)^\gamma \leq 1$ and α is typically small ~ 0.1 (Lin et al., 2018; Yi et al., 2020) compared to gradients appearing in likelihood terms appearing above, we can conveniently ignore the first term for the gradient analysis.

Simplifying the above equation, we get

$$-\alpha(1-p_w)^\gamma \left(\underbrace{\frac{\nabla_\theta \pi_\theta(y_w \mid x)}{\pi_\theta(y_w \mid x)}}_{\text{increase } \pi_\theta(y_w|x)} + \underbrace{\frac{\nabla_\theta \pi_\theta(y_l \mid x)}{1-\pi_\theta(y_l \mid x)}}_{\text{decrease } \pi_\theta(y_l|x)}\right). \tag{34}$$

We can now draw some insights and direct comparisons of our approach with Direct Preference Optimization (DPO) (Rafailov et al., 2023). As in most contrastive preference learning gradient terms (Rafailov et al., 2023; Hong et al., 2024; Xu et al., 2024; Meng et al., 2024; Ethayarajh et al., 2024), the term $\frac{\nabla_\theta \pi_\theta(y_w|x)}{\pi_\theta(y_w|x)}$ in Eq. 34 amplifies the gradient when $\pi_\theta(y_w \mid x)$ is low, driving up the likelihood of the preferred response y_w. Similarly, $\frac{\nabla_\theta \pi_\theta(y_l|x)}{1-\pi_\theta(y_l|x)}$ penalizes overconfidence in incorrect completions y_l when p_w is low, encouraging the model to hike preferred response likelihood while discouraging dispreferred ones.

The key insight here is that the modulating term, $(1 - p_w)^\gamma$, strategically amplifies corrections for difficult examples where the probability p_w of the correct (winning) response is low. Intuitively, unlike DPO's fixed β that is applied across the whole training dataset, this modulating term amplifies gradient updates when preference signals are weak ($p_w \approx 0.5$) and tempering updates when they are strong ($p_w \approx 1$), thus ensuring robust learning across varying preference scenarios. Intuitively, when ($p_w \approx 1$), the model is already confident of its decision since p_w remains high, indicating increased model confidence for deterministic preferences. In contrast, when p_w is small, $(1 - p_w)$ remains near 1, and the term $(1 - p_w)^\gamma$ retains significant magnitude, especially for larger values of γ. This allows π_θ in DRDO to learn from both deterministic and non-deterministic preferences, effectively blending reward alignment with preference signals to guide optimization.

For a deeper intuition, consider the case where the true preference probabilities from $p^*(x, y) \sim \frac{1}{2}$. In this case, since non-deterministic preference samples are typically low, from Proposition 1, DPO would assign zero difference in its implicit rewards, especially for finite preference data. Then as Lemma 1(a) suggests, DPO gradients would effectively be nullified, regardless of β since the its reward difference range $\in (-\infty, +\infty)$. In this case as π_θ cannot distinguish between the preference pair and, in turn, effectively misses out on the preference information for such samples.

On the other hand, for $|\mathcal{D}_{\text{nd}}| \ll N$, if π_θ in DPO estimates the true preference close to 1 where $\hat{p} = 1$, as Lemma 1(b) suggests, the empirical policy would assign very high probabilities to tokens that do not even appear in the data. This leads to a surprising combination of both underfitting and overfitting, except the overfitting here results in DPO policy generating tokens that are irrelevant to the context.

However, as Lemma 2 suggests, e-DPO does not directly face this limitation *during* training, but the degeneracy manifests upon convergence. Under the same conditions, the DRDO loss still operates at a sample level because if the DRDO estimate of p^* (via p_w) is close to its true value of $\sim \frac{1}{2}$, the modulating factor ensures that gradients do not vanish. This allows DRDO to continue learning from such samples until convergence when winning and losing probabilities are pushed further apart (Fig. 2). Intuitively, the $\log\left(\frac{\pi_\theta(y_w|x)}{1-\pi_\theta(y_l|x)}\right)$ term is minimized precisely under this condition where the modulating term is also close to zero.

B Further Notes on Experimental Setup

We provide the following additional explanatory notes regarding the experimental setup:

- For non-deterministic and nuanced preferences, note that although we fine-tune all approaches (including DRDO) on `https://huggingface.co/datasets/CarperAI/openai_summarize_tldr`, every baseline we use is policy-aligned with only the training data within $\mathcal{D}_{all}$, $\mathcal{D}_{hc,he}$ and $\mathcal{D}_{\ell c,\ell e}$ for a direct comparison.

- Pal et al. (2024) assume non-determinism of preferences to be correlated to edit-distances between pairwise-samples, but we do not make sure assumptions and consider both the true (oracle) rewards and edit-distances between pairs to verify the robustness of our method. Pal et al. (2024)'s theoretical framework brings insights on DPO's suboptimality assumes small edit distance between pairwise samples and they empirically show this primarily for math and reasoning based tasks. In contrast, our evaluation framework is more general in the sense that we consider both the oracle reward difference as well as edit distance in addressing DPO's limitations in learning from non-deterministic preferences and we evaluate on a more diverse set of prompts apart from math and reasoning tasks.

DRDO Algorithm

Input: Preference dataset $\mathcal{D}_{\text{pref}} = \{(x^{(i)}, y_w^{(i)}, y_l^{(i)})\}_{i=1}^N$, initialized policy model with reward head $\pi_{\theta,\theta'} \leftarrow \text{SFT}(\theta) \oplus r_{\theta'}$.
Output: Optimized model parameters θ in policy π_θ.

1. Train Oracle r_ϕ with loss $\mathcal{L}_{\mathcal{O}}(r_\phi, \mathcal{D}_{\text{pref}})$ (see Eq. 5).
2. For $t = 1, \ldots, T$:
   (a) For each $(x^{(i)}, y_w^{(i)}, y_l^{(i)})$ in $\mathcal{D}_{\text{pref}}$:
      i. Compute $r_1^* = r_\phi(x^{(i)}, y_w^{(i)})$ and $r_2^* = r_\phi(x^{(i)}, y_l^{(i)})$.
      ii. Compute $\hat{r}_1 = r_{\theta'}(x^{(i)}, y_w^{(i)})$ and $\hat{r}_2 = r_{\theta'}(x^{(i)}, y_l^{(i)})$.
      iii. Compute knowledge distillation loss:

$$\mathcal{L}_{\text{kd}}(r^*, \pi_\theta) = \mathbb{E}_{(x^{(i)}, y_w^{(i)}, y_l^{(i)}) \sim \mathcal{D}_{\text{pref}}} \left[\underbrace{(r_1^* - r_2^* - (\hat{r}_1 - \hat{r}_2))^2}_{\text{Reward Difference}} - \underbrace{\alpha(1 - p_w)^\gamma \log \left(\frac{\pi_\theta(y_w^{(i)} \mid x^{(i)})}{1 - \pi_\theta(y_l^{(i)} \mid x^{(i)})} \right)}_{\text{Contrastive Log-"unlikelihood"}} \right].$$

      iv. Update $\pi_{\theta,\theta'}$ using $\mathcal{L}_{\text{kd}}$.
3. **Return:** Aligned policy π_θ.

Table 5: DRDO Algorithm steps. We start off with the preference dataset and an SFT-trained policy initialized with an additional linear head parameterized by θ'. Once our oracle is trained, we compute both estimated rewards for each response (y) from the initial policy ($\hat{r}$) as well as from the oracle (r^*). We then use $\mathcal{L}_{\text{kd}}$ to update both θ and θ' in π_θ resulting in our DRDO aligned policies.

- For reward distillation w.r.t to model size, the version of Ultrafeedback we used can be found at `https://huggingface.co/datasets/argilla/ultrafeedback-binarized-preferences-cleaned`.
- For SFT on both experiments, use the TRL library implementation (`https://huggingface.co/docs/trl/en/sft_trainer`) for SFT training on all initial policies for all baselines.

B.1 Choice of Evaluation Datasets

As mentioned in Sec. 5, our experiments were designed to balance robustness and thoroughness with research budget constraints, and to account for properties of the task being evaluated relative to the data. The rationale for evaluating DRDO's robustness to non-deterministic or ambiguous preferences is straightforward: the TL;DR dataset is annotated with human confidence labels, which enables the creation of the $\mathcal{D}_{hc,he}$ and $\mathcal{D}_{\ell c,\ell e}$ splits requires to test the different non-determinism settings.

Testing the effect of model size on reward distillation does not require human confidence labels, and as the TL;DR training data size is 1.3M rows, testing distillation across 5 models became computationally intractable given available resources. Thus we turned to Ultrafeedback, which at a train size of 61.1k rows made this experiment much more feasible. Additionally, Ultrafeedback is rather better suited to the preference distillation problem because Ultrafeedback was specifically annotated and cleaned (Cui et al., 2024), for the purposes of evaluating open source models for distillation. Ultrafeedback also provides a high quality dataset with GPT-4 or scores on both straighforward summarization instruction following *and* multiple preference dimensions such as honesty, truthfulness, and helpfulness. Previous work like Zephyr (Tunstall et al., 2023) utilize this dataset for evaluating preference distillation. However, their method is more of a data-augmentation method and does not provide any novel distillation algorithms for preferences since they use the DPO loss but with cleaner and diverse feedback data. They only test student models of $\sim$7B parameters, which are still relatively large models that require additional compute for training without including some form of PEFT. As such, Ultrafeedback is best suited to evaluate distillation-based methods and our novel contribution in this area included evaluation of smaller distilled models.

B.2 TL;DR Summarization Dataset Splits

Table 6 shows the mean confidence and normalized edit distance statistics in the TL;DR dataset which we used to compute the deterministic and non-determinisitic splits. $\mathcal{D}_{all}$ represents the full

Table 6: Mean confidence and normalized edit distance statistics our TL;DR preference generalization experiment.

Split	Conf (Mean)	Edit Dist (Mean)	Train	Validation
$\mathcal{D}_{all}$	5.01	0.12	44,709	86,086
$\mathcal{D}_{hc,he}$	7.31	0.15	10,136	1,127
$\mathcal{D}_{\ell c,\ell e}$	2.67	0.09	10,000	1,112

data (Stiennon et al., 2020). $\mathcal{D}_{hc,he}$ and $\mathcal{D}_{\ell c,\ell e}$ represent subsets created by splitting at the 50th percentile of human labeler confidence and edit distance values. The range of labeler confidence values for the full training data range is $[1, 9]$. Additionally, segmenting the training data based on the *combination* of confidence and edit distance thresholds do not make the splits roughly half of the full training data. This is because there are many samples that do not simultaneously satisfy the 50th percentile threshold for each metric. In reality, this choice is intentional to more robustly evaluate DRDO under more difficult preference data settings. Additionally, previous theoretical as well as empirical work (Pal et al., 2024) has shown that supervised methods like DPO fail to learn optimal policies when the token-level similarity is high in the preference pairs, especially in the beginning of the response. Therefore, we apply this combined thresholding for all our experiments on TL;DR summarization dataset.

C Preference Likelihood Analysis

Fig. 4 shows preference optimization method performance vs. training steps, according to Bradley-Terry (BT) implicit reward accuracies (Fig. 4a), Oracle reward advantage (Fig. 4b), preferred log-probabilities (Fig. 4c) and dispreferred log-probabilities (Fig. 4d). Although all baselines show roughly equal performance in increasing the likelihood of preferred responses (y_w), DRDO is particularly efficient in penalizing dispreferred responses (y_l) as Fig. 4d suggests.

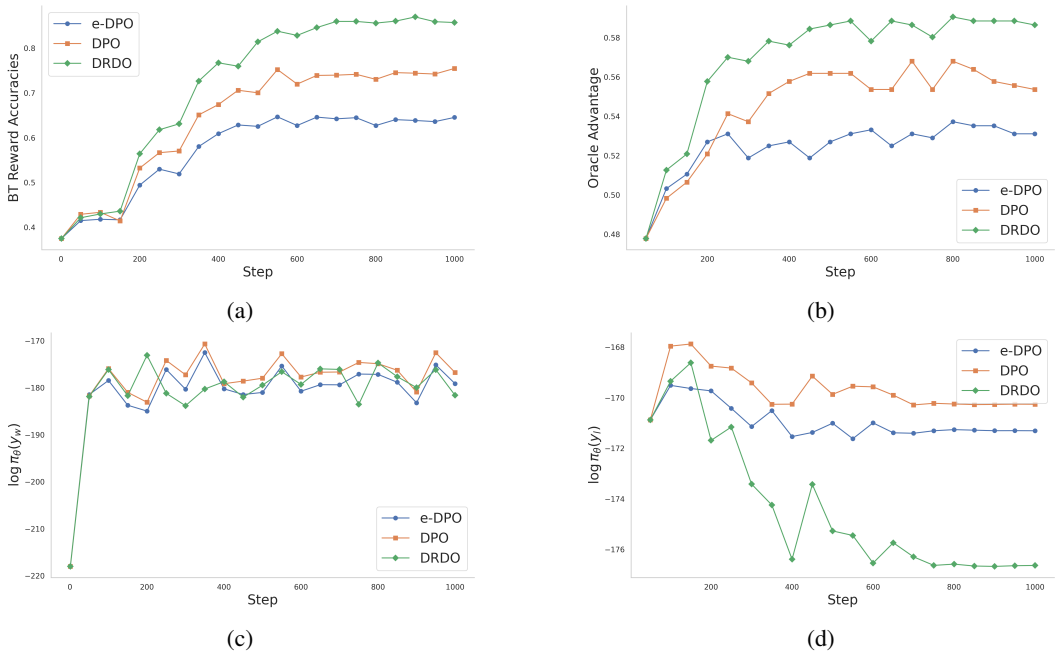

Figure 4: Top: DRDO performance evolution during OPT 1.3B training compared to DPO and e-DPO on the evaluation set of Ultrafeedback (Cui et al., 2024), and randomly sampled generations to compute the reward advantage against the preferred reference generations.

Table 7: Model Configuration and Full set of hyperparameter used for DRDO Training

Parameter	**Default Value**
learning_rate	5e-6
lr_scheduler_type	cosine
weight_decay	0.05
optimizer_type	paged_adamw_32bit
loss_type	DRDO
per_device_train_batch_size	12
per_device_eval_batch_size	12
gradient_accumulation_steps	4
gradient_checkpointing	True
gradient_checkpointing_use_reentrant	False
max_prompt_length	512
max_length	1024
max_new_tokens	256
max_steps	20
logging_steps	5
save_steps	200
save_strategy	no
eval_steps	5
log_freq	1
α	0.1
γ	2

D HYPERPARAMETERS

We only use full-parameter training for all our policy models. We train all our student policies for 1k steps with an effective batch size of 64, after applying an gradient accumulation step of 4. Specifically, both OPT series and Phi-3-Mini-4K-Instruct model were optimized using DeepSpeed ZeRO 2 (Rasley et al., 2020) for faster training. All models were trained on 2 NVIDIA A100 GPUs, except for certain runs that were conducted on an additional L40 gpu. For the optimizer, we used AdamW (Loshchilov et al., 2017) and paged AdamW (Dettmers et al., 2024) optimizers with learning rates that were linearly warmed up with a cosine-scheduled decay. For both datasets, we filter for prompt and response pairs that are $<$ 1024 tokens after tokenization. This allows the policies enough context for coherent generation. Apart from keeping compute requirement reasonable, this avoids degeneration during inference since we force the model to only generate upto 256 new tokens not including the prompt length in the maximum token length.

For our DRDO approach, we sweep over $\alpha \in \{.1, 1\}$ and $\gamma \in \{0, 1, 2, 5\}$ but found the most optimal combination to be α = 0.1 and γ = 2, since a higher γ tends to destabilize training due to the larger penalties induced on DRDO loss. This is consistent with optimal γ values found in the literature, albeit for different tasks (Yi et al., 2020; Lin et al., 2018). For Oracle trained for DRDO, we use the same batch size as for policy training with a slightly larger learning rate of 1e-5 and train for epoch. For consistency, we use a maximum length of 1024 tokens after filtering for pairs with prompt and responses < 1024 tokens. For all SFT training, we use the TRL library[7] with a learning rate of 1e-5 with a cosine scheduler and 100 warmup steps.

For the DPO baselines, we found the implementation in DPO Trainer[8] For optimal parameter selection, we sweep over $\beta \in \{.1, 0.5, 1, 10\}$ but we found the default value of β = 0.1 to be most optimal based on the validation sets during training. Also, we found the default learning rate of 5e-6 to be optimal after validation runs. For e-DPO, we restrict the number of reward ensembles to 3 but use the same Oracle training hyperparameters mentioned above.

[7] `https://huggingface.co/docs/trl/en/sft_trainer`

[8] `https://huggingface.co/docs/trl/main/en/dpo_trainer` to be the most stable and build off most of our DRDO training pipline and configuration files based on their trainer.

Table 7 provides a full list of model configurations and hyperparameters used during trainng of DRDO models.

E WIN-RATE EVALUATION PROMPT FORMATS

Fig. 6 and Fig. 7 show the prompt format used for GPT-4o evaluation of policy generations compared to human summaries provided in the evaluation data of Reddit TL;DR (CNN Daily Articles). Fig. 6 specifically provides the human-written summaries as reference in GPT's evaluation of the baselines. In contrast, Fig. 7 shows the prompt that was used to evaluate policy generated summaries in direct comparison to human-written summaries. Note that in both prompts, we swap order of provided summaries to avoid any positional bias in GPT-4o's automatic evaluation.

Further ablations using GPT-4o as oracle For a more robust evaluation using a much more high-capacity oracle (GPT-4o), we randomly sampled 40 prompts from the CNN daily test set and compute win-rates and reward margins of samples generated with a top-p of 0.8 and T=0.7 for all baselines vs. DRDO policies trained on Reddit TL;DR. We use the Phi-3-Mini-4K-Instruct model for this experiment. We include the IPO baseline (Azar et al., 2024) with $\beta = 0.1$ (or τ in their paper), the baselines without the distillation (shown as DRDO (-R)) and contrastive component (shown as DRDO (-C)). Additionally, we include the baseline where reward distillation term in DRDO is replaced by the DPO loss but keep the contrastive log-unlikelihood component (shown as DPO (-C)). Note that due to compute constraints, we only train these policies from the SFT checkpoint for 500 steps with an effective batch size of 128. For the reward estimates using GPT-4o, we only add one additional condition to the prompt shown in Fig. 5 to get scalar rewards (between 0 and 1). Fig. 8 provides the prompt format used for this experiment.

Comparison	WR A (%)	WR B (%)	Reward A	Reward B	Margin A	Margin B
DRDO vs. DRDO (-R)	85.0	15.0	$0.24_{\pm 0.26}$	$0.07_{\pm 0.08}$	$0.22_{\pm 0.22}$	$0.09_{\pm 0.04}$
DRDO vs. DRDO (-C)	90.0	10.0	$0.25_{\pm 0.23}$	$0.05_{\pm 0.07}$	$0.23_{\pm 0.22}$	$0.06_{\pm 0.03}$
DRDO vs. IPO	65.0	35.0	$0.21_{\pm 0.17}$	$0.21_{\pm 0.24}$	$0.09_{\pm 0.08}$	$0.16_{\pm 0.16}$
DRDO vs. DPO (-C)	80.0	20.0	$0.18_{\pm 0.16}$	$0.14_{\pm 0.15}$	$0.11_{\pm 0.08}$	$0.22_{\pm 0.21}$

Table 8: DRDO policies (shown as A) compared against various baselines (shown as B)–win-rates (WR) are computed using average of reward comparison for each sample and then averaged. Margins are computed using the difference of rewards.

DRDO Performance vs Extent of Out-of-distribution data In order to comprehensively evaluate DRDO's performance against baseline methods under increasing out-of-distribution (OOD) conditions, we randomly sample 1,000 prompts from the CNN/DailyMail dataset and segment these prompts into bins of 50 tokens based on prompt-token counts, spanning the full range of token lengths. Since the CNN/DailyMail dataset represents a previously unseen input distribution, this evaluation effectively measures the OOD generalization capabilities of the policies as prompt lengths (and corresponding news article lengths) increase. For this automatic evaluation, we use GPT-4o as a high-capacity judge, consistent with Sec. 5 (prompt used is shown in Fig. 6). For response sampling, we use top-p of 0.8 and temperature of 0.7 for DRDO and all baselines. The trends in Fig. 5 suggest that as OOD composition (with prompt-token lengths as proxy) increases, on average DRDO policies tend to have relatively larger win-rates compared to shorter prompts over baselines like DPO and e-DPO. In contrast, we find that DPO and e-DPO win-rates tend to decrease with increase in prompt-lengths.

F COMPUTATIONAL EFFICIENCY

e-DPO requires training reward ensembles to form a confidence set for training the policy. In our experiments, we use 3 reward models to construct this set which makes it roughly thrice as expensive as DRDO training. Like DPO, e-DPO requires a separate reference model to be kept in memory, further increasing compute requirements. DRDO, on the other hand, only requires a trained oracle for distillation. The expected oracle rewards can be precomputed once and a separate reference

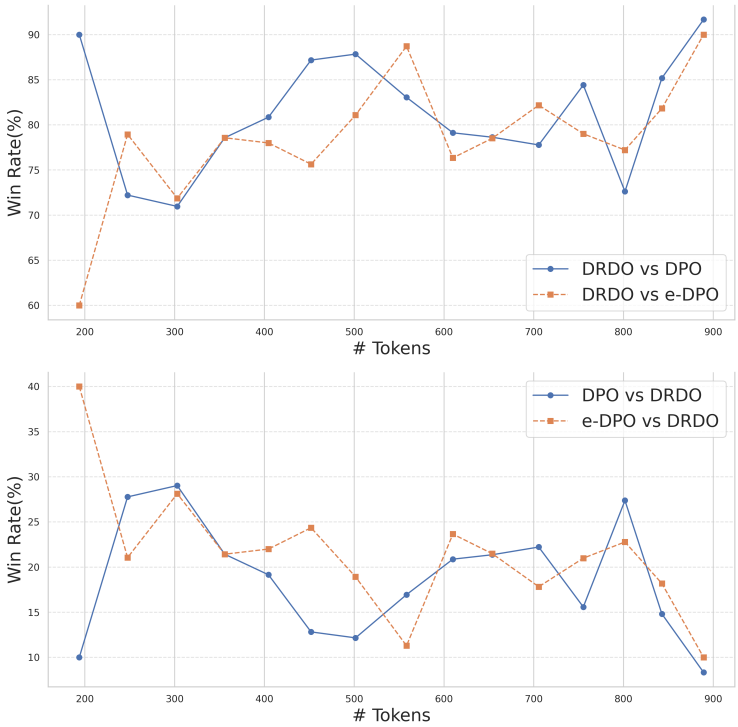

Figure 5: Comparison of win-rates as a function of the extent of out-of-distribution (OOD) data on the CNN daily article dataset. Win-rates (y-axis) of DRDO vs DPO and e-DPO (top) and competitor win-rates (bottom) are plotted against the increasing prompt lengths (number of tokens) over 1000 randomly sampled prompts for evaluation. DRDO is more robust to OOD settings, on average, compared to baselines like DPO and e-DPO as seen in the upward trend in win-rates over prompt-tokens.

Summarization GPT-4 win rate prompt (C).

```
Which of the following summaries do a better job of summarizing the most \
important points in the given forum post, without including unimportant \
or irrelevant details? Make your decision while referring to the reference\
(human-written) summary. A good summary is both precise and concise.

Post:
<post>

Reference Summary:
<golden summary>

Summary A:
<Summary A>
Summary B:
<Summary B>

FIRST provide a one-sentence comparison of the two summaries, explaining \
which you prefer and why. SECOND, on a new line, state only "A" or "B" to \
indicate your choice. Your response should use the format:
Comparison: <one-sentence comparison and explanation>
Preferred: <"A" or "B">
```

Figure 6: Prompt format for Reddit TL;DR (CNN Daily)

model does not need to be kept in memory (as shown in Eq. 6). During training, DRDO does require one additional linear head on top of the base LM to predict the reward estimates. This adds a

Summarization GPT-4 win rate prompt (vs human written).

```
Which of the following summaries does a better job of summarizing the most \
important points in the given news article, without including unimportant \
or irrelevant details? A good summary is both precise and concise.

Post:
<post>

Summary A:
<golden summary>
Summary B:
<summary>

FIRST provide a one-sentence comparison of the two summaries, explaining \
which you prefer and why. SECOND, on a new line, state only "A" or "B" to \
indicate your choice. Your response should use the format:
Comparison: <one-sentence comparison and explanation>
Preferred: <"A" or "B">
```

Figure 7: Prompt format for Reddit TL;DR (CNN Daily)

Summarization GPT-4o win rate prompt (C).

Which of the following summaries do a better job of summarizing the most important points in the given forum post, without including unimportant or irrelevant details? Make your decision while referring to the reference (human-written) summary. A good summary is both precise and concise.

Post:
<post>

Reference Summary:
<golden summary>

Summary A:
<Summary A>
Summary B:
<Summary B>

FIRST, provide a one-sentence comparison of the two summaries, explaining which you prefer and why.
SECOND, on a new line, state only **"A"** or **"B"** to indicate your choice. Your response should use the format:
THIRD, on a new line, provide your ratings (a real reward score between 0 to 1 where 1 is highest and 0 is lowest in quality) for the summaries.
Comparison: <one-sentence comparison and explanation>
Preferred: <"A" or "B">
Score for Summary A: <score>
Score for Summary B: <score>

Figure 8: Prompt format for Reddit TL;DR

negligible 0.003% more trainable parameters (relative to the language modeling head of base LM Phi-3-Mini-4K-Instruct). During inference, DRDO trained policies do not require this head.