UNA: A Unified Supervised Framework for Efficient LLM Alignment Across Feedback Types

Anonymous ACL submission

Abstract

RL alignment methods, including RLHF and DPO, are primarily based on pairwise preference data. Although scalar or score-based feedback has been collected in some settings, it is rarely used directly, and preference magnitude information is typically ignored. Furthermore, current alignment frameworks offer limited capability for unifying heterogeneous supervision signals, making it difficult to jointly leverage diverse data types within a single training paradigm. This limitation constrains the richness and scalability of the alignment process. To address this gap, we propose a **UN**ified **A**lignment (UNA) framework capable of training across different types of feedback, including binary, pairwise, and score-based, through a generalized implicit reward function. The reward function is theoretically proved to be the optimal policy by the log sum inequality. Extensive experiments on classical benchmarks consistently demonstrate the advantage of the proposed unified framework with typical LLM base models.

1 Introduction

LLMs are deployed in diverse, real-world settings where feedback is rarely uniform. Therefore, it is important to achieve alignment training across different types of supervision signals. Heterogeneous data, such as the pairwise preference data, score feedback, Likert-scale human ratings, and domain-specific evaluation metrics, capture different facets of human expectations. A framework that can unify these heterogeneous data sources enables models to leverage substantially richer information and reduces reliance on any single annotation paradigm.

Score-based feedback encodes the degree or intensity of preference, offering more fine-grained guidance than binary comparisons. Several datasets already provide such score signals. For example, OpenAI's WebGPT (Nakano et al., 2021) includes Likert-scale human ratings of quality; AnthropicHH [1] dataset has multi-level preference labels; UltraFeedback (Cui et al., 2023) provides 0–10 quality ratings across multiple dimensions; and HelpSteer (Wang et al., 2024d) and HelpSteer 2 (Wang et al., 2024c) include numeric helpfulness and safety scores. These datasets include informative supervision signals, yet few alignment methods can utilize score-based feedback.

Especially, there is no unified framework that integrates diverse forms of feedback. Existing methods often require separate training pipelines for each feedback type, e.g., Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022) with PPO for rewards, Direct Preference Optimization (DPO) (Rafailov et al., 2023) for pairwise data. Little work has been done to unify diverse forms of feedback into a single training objective. This allows training to proceed under one consistent optimization framework and has the potential to utilize a broader supervision space for more robust and effective alignment.

RLHF, as shown in Figure 1(b), is a two-stage process. First, a Reward Model (RM) is trained on pairwise preference data. Next, the LLM policy is fine-tuned through RL, typically using proximal policy optimization (PPO) (Schulman et al., 2017), where the RM evaluates the generated responses. However, RLHF faces several limitations: overfitting in RM training (Zhu et al., 2024; Huyen, 2023), unstable RL fine-tuning (Ma et al., 2024; Byun et al., 2024), and high memory requirements for maintaining multiple models (policy, reference policy, RM, and value model) (Wang et al., 2024a).

DPO simplifies this by mathematically establishing a mapping between the RM and the optimal policy, combining the RM and RL training into a single, stable binary classification problem (Figure 1(c)). This eliminates the need for an explicit RM and reduces memory costs. The policy is optimized

[1] https://huggingface.co/datasets/Anthropic/hh-rlhf

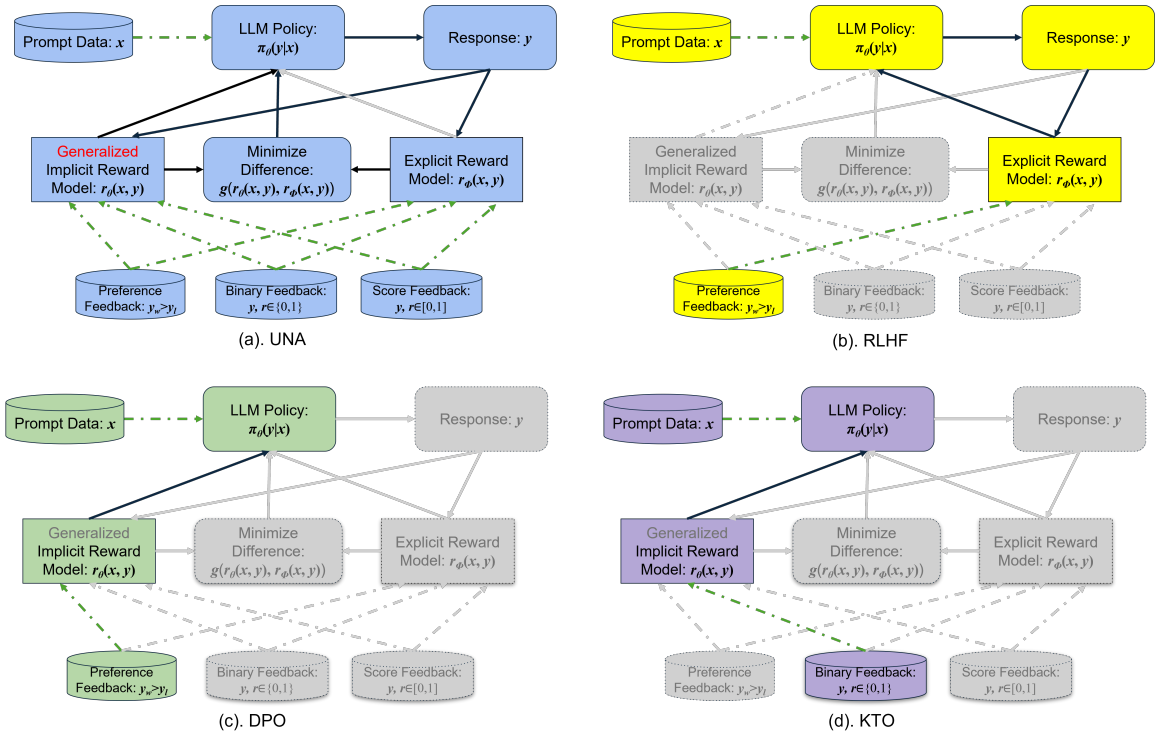

Figure 1: A figure comparison among (a). UNA, (b) RLHF, (c) DPO and (d) KTO. Each subfigure is composed of four types of data: “prompt data", “preference feedback", “binary feedback" and “score feedback", “LLM policy", “response", two reward models: “generalized implicit reward model" and “explicit reward model" and a module to minimize the difference between implicit and explicit rewards. The connection between data to other modules are utilizing green dash arrow, while others are connected by black solid arrow. All unused modules are grayed out. In part (b), RLHF firstly utilizes preference feedback to train the explicit reward model, and then use the evaluation provided by the explicit reward model to continuous optimize the policy in a online mode. In comparison, in part (c) and (d), DPO and KTO utilize preference feedback and binary feedback respectively to generate implicit reward to align LLM policy. However, in part (a), UNA can utilize **different types of data** to get the generalized implicit and explicit rewards and minimize their differences to align LLM policy in **both online and offline modes**.

using the difference in implicit rewards calculated for desired and undesired responses. Kahneman-Tversky Optimization (KTO) (Ethayarajh et al., 2024) extends DPO to use binary feedback for desired and undesired responses (Figure 1(d)). Despite their efficiency, both DPO and KTO require labeled training data for direct supervised optimization, whereas RLHF/PPO relies on a learned RM to guide policy optimization.

We propose UNA, a method capable of training with different data types, addressing the limitations of RLHF/PPO, DPO, and KTO. We design an implicit reward model in UNA and prove that the optimal policy $\pi^*_\theta(y|x)$ for the RLHF objective is achieved when the implicit reward model is satisfied. Based on this generalized implicit reward, UNA unifies RLHF/PPO, DPO, and KTO into a supervised learning framework that minimizes the difference between an implicit reward and an explicit reward (Figure 1(a)). The explicit reward can be derived from human labelers, reward functions, or LLMs. UNA reformulates alignment by replacing PPO-based reinforcement learning in conventional RLHF pipelines with a supervised learning objective that minimizes the differences between an implicit reward derived from the policy and an explicit reward signal, enabling more stable and computationally efficient training.

This paper has **the following contributions**:

(1) We propose UNA, a unified alignment framework that reformulates RLHF/PPO, DPO, and KTO within a supervised learning paradigm, which can flexibly accommodate diverse feedback types, including pairwise, binary, and scalar rewards, under both online and offline training settings.

(2) We mathematically prove that the optimal policy derived from the RLHF objective function is induced by the reward function $r_\theta(x,y) = \beta \log \left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right)$.

(3) We conduct comprehensive experiments

across multiple benchmark datasets and show that UNA surpasses DPO and KTO across different tasks. It achieves this while simplifying the RL fine-tuning process in RLHF/PPO, improving stability, and significantly reducing memory overhead.

2 Related Work

The LLM field has been transformed by large-scale pretraining with billions of parameters and trillions of tokens (OpenAI et al., 2024; Anthropic, 2024; Team et al., 2023). After pretraining, SFT is applied to enhance performance on downstream tasks. However, pretraining and SFT alone cannot resolve bias and ethical issues inherent in the training data (OpenAI et al., 2024).

To address these challenges, RLHF with PPO has been widely adopted to align LLMs, including GPT and Claude (Ouyang et al., 2022; Bai et al., 2022a). Despite its effectiveness, RLHF/PPO suffers from high memory usage, instability, and multi-stage training complexity, including separate RM training and RL fine-tuning (Rafailov et al., 2023). To reduce the cost of human labeling, AI feedback can replace human feedback in a method called RLAIF (Bai et al., 2022b; Lee et al., 2023). RLOO, in contrast, considers PPO overkill for pretrained LLMs and provides a simpler alignment alternative (Ahmadian et al., 2024).

RLHF and RLAIF remain complex, unstable, and memory-intensive; GRPO addresses these issues by removing the PPO value model and using the average reward as the advantage baseline (Shao et al., 2024). In DAPO (Yu et al., 2025), the authors utilized (i) a higher ceiling clip, (ii) dynamic sampling, (iii) token-level policy gradient loss, and (iv) overlong reward shaping to improve the stability of GRPO. GSPO replaces (i) the token-level importance ratio with a sequence-level importance ratio and (ii) routing replay to stabilize the expert choice in GRPO (Zheng et al., 2025).

Simplifying RLHF: DPO and Variants DPO simplifies RLHF by mapping the optimal policy to the reward model in a single step, transforming unstable RL training into a binary classification problem (Rafailov et al., 2023). DPOP (Pal et al., 2024) mathematically demonstrates that the reward of desired responses may decrease during DPO and introduces a maximum term to prevent this. IPO identifies that under nearly deterministic conditions, the KL divergence constraint imposed by β may become ineffective, potentially leading to overfitting, and proposes a new loss term to mitigate this (Azar et al., 2023). Sequential DPO (sDPO) divides the dataset into splits and aligns the model sequentially, achieving better performance than using the entire dataset at once (Kim et al., 2024). Iterative DPO leverages the LLM as both response generator and evaluator to iteratively improve itself (Yuan et al., 2024; Xu et al., 2024). TDPO provides token-level rewards to refine generation (Rafailov et al., 2024; Zeng et al., 2024).

Merging SFT with Alignment Several works integrate SFT with alignment. For example, ORPO introduces a loss function that increases the ratio of desired over undesired responses, achieving the goals of both SFT and alignment (Hong et al., 2024). PAFT conducts SFT and alignment in parallel and merges the results afterward (Pentyala et al., 2024). R-DPO (Park et al., 2024) and SimPO (Meng et al., 2024) address the verbosity problem in outputs, using length-control methods to reduce response length while maintaining performance.

Feedback Types: Pairwise, Binary, and Ranking Early work focused on pairwise datasets, which are costly to collect. Binary feedback, such as "thumbs up" or "thumbs down," is easier to obtain. KTO leverages human preference between desired and undesired responses for effective binary feedback alignment (Ethayarajh et al., 2024), while DRO optimizes binary feedback by estimating policy and value functions sequentially (Richemond et al., 2024). Nash learning models LLM improvement as a min-max problem, addressing intransitivity in human preferences through iterative optimization (Munos et al., 2024), though at the cost of increased training time. SPPO uses a single model to simulate both sides of a competitive setup (Wu et al., 2024).

Ranking-based approaches, such as LiPO (Liu et al., 2024), RRHF (Yuan et al., 2023), and PRO (Song et al., 2024), utilize the ranking of response lists and relative scores. RPO minimizes KL divergence between predicted and labeled rewards, aligning closely with the method proposed in this work (Nvidia et al., 2024).

Open Challenges Despite these advances, several challenges remain: (i) No unified method currently combines RLHF and DPO; (ii) There is no unified framework that integrates diverse forms of feedback, including pairwise, binary, and listwise ranking. These limitations motivate the development of UNA, which aims to unify and simplify LLM alignment methods such as PPO, DPO, and KTO, while improving training stability and efficiency

compared to conventional RLHF pipelines. Unlike RLHF, which relies on reinforcement learning with an explicit reward optimization loop, UNA formulates alignment as a supervised learning problem. This design enables more stable training dynamics and higher computational efficiency while retaining alignment effectiveness.

3 UNified Alignment (UNA) Framework

We introduce UNA, which derives a general loss function that reformulates RLHF/PPO, DPO, and KTO under a unified supervised learning framework, allowing UNA to leverage different types of feedback data. We further compare UNA with existing techniques and show the relationship of UNA to DPO, KTO, and RLHF/PPO.

3.1 UNA via Implicit Reward Modeling

Inspired by the idea of DPO, we propose a new relationship between the implicit reward model and the optimal policy for a unified alignment framework, including RLHF/PPO, DPO, and KTO on different types of data. By adhering to the same objective outlined in RLHF (Equation 1), we formulate a novel connection between the implicit reward function and the optimal policy, as shown in Equation 2. The derivation can be found in Section 4 through log-sum inequality. A more general derivation of UNA, which arrives at the same result as DPO, is presented in Section D.

$$\pi_\theta^*(y|x) = \arg\max_{\pi_\theta} \mathbb{E}_{x\sim D}\Big[\mathbb{E}_{y\sim\pi_\theta(\cdot|x)}[r_\phi(x,y)] - \beta\, D_{\mathrm{KL}}(\pi_\theta(\cdot \mid x) \,\|\, \pi_{\mathrm{ref}}(\cdot \mid x))\Big] \tag{1}$$

$$r_\theta(x,y) = \beta \log\left(\frac{\pi_\theta(y|x)}{\pi_{\mathrm{ref}}(y|x)}\right) \tag{2}$$

The optimal implicit reward formulation in Equation 2 implies that we can transform the original unstable, memory-expensive RL training process into a reward function optimization problem, i.e., a stable and memory-efficient supervised learning process. The explicit rewards in the original RL training process can be derived from multiple methods in both online and offline modes, including human labeling, LLM-as-a-Judge, and a reward model. Eventually, the RL fine-tuning process is transformed into a general minimization problem between explicit reward $r_\phi(x,y)$ and implicit reward $r_\theta(x,y)$ as shown in Equation 3 where $g(x_1,x_2)$ refers to a general function that measure the difference between x_1 and x_2 like MSE.

The implicit and explicit rewards may be defined on different numerical scales. For example, the explicit reward can be provided on a scale of $[0,1,2,3,4,5]$, while the implicit reward is $[-1,1]$. To ensure meaningful comparison and stable optimization, rewards are normalized before training so that implicit and explicit rewards are aligned on a common scale before the difference is calculated.

$$L_{\text{UNA-reward}}(\pi_\theta) = \mathbb{E}_{(x,y)\sim D}[g(r_\phi(x,y), r_\theta(x,y))] \tag{3}$$

Leveraging this general implicit reward function, UNA can be applied in both online and offline modes. Figure 2 illustrates UNA's applications to various data types and its simplification of RLHF.

3.1.1 Offline UNA

In Offline UNA, prompts, responses, and their corresponding explicit rewards (x,y,r) are gathered before training. These explicit rewards can encompass pairwise feedback, binary feedback, and score-based feedback, all of which the UNA framework is designed to handle seamlessly. Offline UNA encompasses: (i) equivalence to DPO for pairwise preference dataset; (ii) compatibility with binary feedback, and (iii) accommodation of score-based feedback.

Equivalence to DPO for Pairwise Datasets For pairwise datasets, the implicit rewards of desired and undesired responses can be derived as shown in Figure 2(a). Then, the LLM policy is aligned by maximizing the difference of implicit rewards between desired and undesired responses. It is equivalent to DPO as the loss function is the same as long as $g(x) = \log[\sigma(x)]$ is applied to the difference of implicit rewards of desired and undesired responses in Equation 3.

Compatibility with Binary Feedback For binary feedback, the positive and negative feedback can be transformed into explicit scores. Positive or 'thumbs up' data can be assigned an explicit reward score of 1, i.e., $r_\phi(x,y_w)=1$. In contrast, negative or 'thumbs down' data can be assigned an explicit reward score of 0, i.e., $r_\phi(x,y_l)=0$. Afterward, the implicit reward is first estimated, and then its difference from the explicit reward model is minimized on a pointwise basis, which contrasts with preference-based feedback. Because the explicit feedback is binary, a normalization function

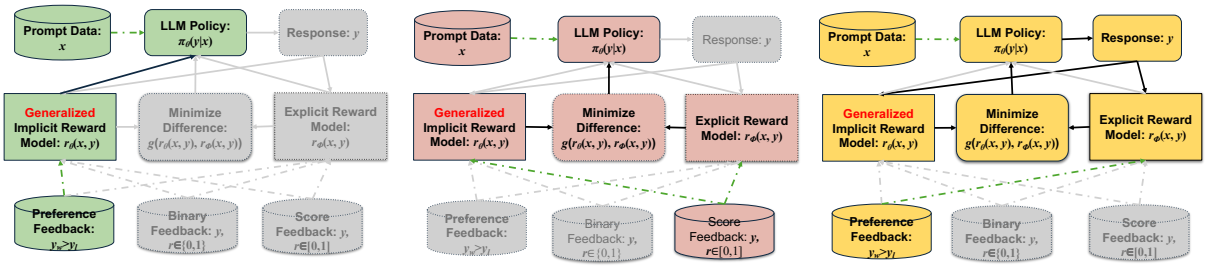

Figure 2: The two applications of UNA: Offline UNA and Online UNA. Offline UNA includes (a). equivalent to DPO for pairwise data, (b). RM/LLM-as-a-Judge distillation for score-based data. Online UNA includes (c). simplification of RLHF for online training. The same modules are utilized as in Figure 1, and unused modules are grayed out. For part (a), the same steps as DPO will be utilized. For part (b), (c), from the different types of data, including pairwise, binary, and score-based feedback, implicit and explicit rewards are firstly gathered. Then, the difference between implicit and explicit rewards is minimized to align the LLM policy.

should be utilized on the implicit reward function beforehand. Considering the implicit and explicit rewards, multiple loss functions can be formulated, like MSE and BCE.

Accommodation of Score-based Feedback Researchers have utilized LLM-as-a-Judge and RM to evaluate responses by outputting score-based rewards according to predefined standards. If the score-based evaluations are accurate enough, they can be an extra information to utilize for alignment, compared with binary or preference feedback. When the tuple type of data (prompt, response, explicit reward) is provided, the prompt and response are utilized to calculate implicit reward as shown in Equation 2, and the model is aligned by minimizing the difference between implicit and explicit rewards as shown in Figure 2(b). Because the explicit rewards from RM and LLM are not binary, usually a score in the interval [0, 1]. As a result, only MSE can be used as the loss function, excluding BCE. In particular, when LLM-as-a-Judge is utilized for evaluation, it can be regarded as an offline version of RLAIF.

3.1.2 Online UNA

Online UNA generates responses y on-the-fly from the current policy given sampled prompts x, and computes rewards r based on the resulting (x, y) pairs via a learned RM. This framework aligns closely with RLHF but offers a more streamlined and stable approach to the RLHF process. Online UNA features improvement over RLHF in the RL fine-tuning stage by replacing PPO with a supervised learning process.

Simplification of RLHF When utilizing a reward model for online evaluation, UNA greatly simplifies the RL fine-tuning stage of RLHF/PPO as shown in Figure 2(c). Assuming the reward model has already been trained, the focus shifts exclusively to the RL fine-tuning stage. Prompts are first sent to the current policy for online response generation and implicit reward estimation. Then, the prompt and response are sent to the reward model for explicit reward estimation. The last step minimize the differences between implicit and explicit rewards to align the LLM policy. Eventually, the original RL objective in Equation 1 can be transformed to difference minimization, like the MSE of implicit reward and explicit reward.

UNA has several benefits over PPO in the RL fine-tuning stage. First, it transforms the original unstable RL problem into a stable supervised learning problem by minimizing the difference between implicit and explicit rewards. Second, UNA removes the necessity of a value model in PPO, and partially reduces memory cost. Finally, the computation cost of MSE is much lower compared with the multiple terms in PPO to maintain performance. As a result, UNA will speed up the training process.

3.2 Theoretical Relationship of UNA to DPO and RLHF

UNA and DPO The implicit rewards of UNA is presented in Equation 2, DPO in Equation 4. The key difference between them lies in the presence of the $\beta \log Z(x)$ term. Specifically, the implicit reward used in UNA can be viewed as a special case of DPO, where the partition function $Z(x) = \sum_y \pi_{\text{ref}}(y|x) \exp\left(\frac{1}{\beta} r_\theta(x, y)\right)$ is equal to 1. This condition is exactly satisfied when the reward func-

tion takes the form $r_\theta(x, y) = \beta \log \left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right)$. Therefore, the optimal implicit reward function in UNA can be interpreted as a special case—and a strict subset—of the optimal implicit reward function used in DPO.

$$r_\theta(x, y) = \beta \log \left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right) + \beta \log Z(x) \quad (4)$$

Since $Z(x)$ is generally intractable, DPO sidesteps this issue by employing pairwise preference comparisons, which allow the partition term to cancel out. However, this approach precludes the use of pointwise preference data, which often carries richer information. In contrast, UNA avoids the need for $Z(x)$ altogether, enabling effective utilization of pointwise preference signals. RLHF typically utilizes pointwise preference signals from a pretrained reward model. From this perspective, UNA unifies RLHF and DPO by bridging their underlying data types—pointwise and pairwise—within a common theoretical framework.

UNA and RLHF Both UNA and RLHF leverage pointwise rewards for aligning language models. RLHF follows a reinforcement learning paradigm, aiming to directly maximize the total explicit reward through policy optimization. In contrast, UNA treats the pointwise reward as an explicit target and aligns the model via supervised learning by minimizing the discrepancy between the implicit reward (induced by the policy) and the explicit reward function. Notably, DPO also employs supervised learning but relies on pairwise preference data for alignment. From this perspective, UNA unifies RLHF and DPO by bridging their training paradigms: It adopts a supervised learning similar to DPO, while retaining the ability to leverage pointwise reward signals as in RLHF.

In summary, DPO aligns LLMs using pairwise preferences via supervised learning, while RLHF aligns LLMs using pointwise preferences through RL. UNA serves as a unifying framework that employs supervised learning to integrate both pairwise and pointwise preference data.

4 Mathematical Proof of UNA

Here we rigorously prove that $r(x, y) = \beta \log \left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right)$ will maximize the objective in RL in Equation 1. The proof deriving the mapping between the optimal policy and the reward model in DPO is provided in the appendix B.

Proposition 1. Log Sum Inequality Let $a_1, \ldots, a_n$ and $b_1, \ldots, b_n$ be non-negative numbers. Denote the sum of all a_i by a, i.e., $\sum_{i=1}^n a_i = a$ and the sum of all b_i by b, i.e., $\sum_{i=1}^n b_i = b$. The log sum inequality states Equation 5 with equality if and only if $\frac{a_i}{b_i}$ are equal for all i, in other words $a_i = \lambda \times b_i$ for all i. The proof could be found in appendix C.

$$\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geq a \log \frac{a}{b} \quad (5)$$

Starting from the same objective in Equation 1, it can be simplified as shown in Equation 6.

$$\begin{aligned}
\pi_\theta^*(y|x) &= \max_{\pi_\theta} \mathbb{E}_{x\sim D} \Big[\mathbb{E}_{y\sim\pi_\theta(y|x)} r_\theta(x, y) \\
&\qquad - \beta D_{\text{KL}} \left(\pi_\theta(y|x) \| \pi_{\text{ref}}(y|x) \right) \Big] \\
&= \max_{\pi_\theta} \mathbb{E}_{x\sim D} \left[\mathbb{E}_{y\sim\pi_\theta(y|x)} \left(r(x, y) - \beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right) \right] \\
&= \max_{\pi_\theta} \mathbb{E}_{x\sim D} \left[\mathbb{E}_{y\sim\pi_\theta(y|x)} \left(\frac{1}{\beta} r(x, y) - \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right) \right] \\
&= \max_{\pi_\theta} \mathbb{E}_{x\sim D} \left[\mathbb{E}_{y\sim\pi_\theta(y|x)} \left(- \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x) e^{\frac{1}{\beta} r(x,y)}} \right) \right]
\end{aligned} \quad (6)$$

Based on the log-sum inequality in Equation 5, the term can be further simplified as shown in Equation 7 because both $\pi_\theta(y|x)$ and $\pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y))}$ are non-negative.

$$\begin{aligned}
&\mathbb{E}_{x\sim D} \left[\mathbb{E}_{y\sim\pi_\theta(y|x)} \left(- \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y))}} \right) \right] \\
&= -\mathbb{E}_{x\sim D} \left[\sum_y \left(\pi_\theta(y|x) \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y))}} \right) \right] \\
&\leq -\mathbb{E}_{x\sim D} \left[\left(\sum_y \pi_\theta(y|x) \right) \log \frac{\sum_y \pi_\theta(y|x)}{\sum_y \pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y))}} \right] \\
&= -\mathbb{E}_{x\sim D} \left(1 \log \frac{1}{\sum_y \pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y))}} \right) \\
&= \mathbb{E}_{x\sim D} \left[\log \left(\sum_y \pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y))} \right) \right] \\
&= \mathbb{E}_{x\sim D} \left[\log \left(\mathbb{E}_{y\sim\pi_{\text{ref}}(y|x)} e^{\frac{1}{\beta}(r(x,y))} \right) \right]
\end{aligned} \quad (7)$$

As a result, the maximum value of the objective function in Equation 1 is $\beta \mathbb{E}_{x\sim D} \left\{ \log \left(\mathbb{E}_{y\sim\pi_{\text{ref}}(y|x)} e^{\frac{1}{\beta}(r(x,y))} \right) \right\}$ in Equation 7, and this inequality reaches the equality

Table 1: Comparison of UNA with DPO, KTO considering pairwise, binary, and score-based data on different benchmarks. Best performance values are in bold.

Method	TruthfulQA	IFEval	HellaSwag	ARC	WinoGrande	MMLU-pro	Math-Hard
Mistral 7B	42.58	23.22	83.44	61.43	77.58	30.11	2.92
+ DPO	44.75	26.30	84.42	62.88	79.16	30.41	2.25
+ KTO	47.72	24.18	84.21	62.29	78.14	30.43	2.34
+ UNA-pairwise	44.75	26.30	84.42	62.88	79.16	30.41	2.25
+ UNA-binary (BCE)	48.33	26.49	84.60	63.14	79.40	**30.73**	2.99
+ UNA-score (MSE)	55.09	37.25	84.52	63.23	**80.27**	29.72	2.77
+ UNA-score & binary	**64.62**	**51.28**	**86.86**	**66.47**	79.79	30.09	**3.25**
Llama 8B	45.16	13.19	81.78	58.11	76.87	32.73	5.66
+ DPO	53.47	19.63	83.01	59.22	78.22	33.05	6.42
+ KTO	55.07	23.24	83.15	59.13	77.66	32.86	**6.95**
+ UNA-pairwise	53.47	19.63	83.01	59.22	78.22	33.05	6.42
+ UNA-binary (BCE)	54.75	22.96	83.00	59.04	78.45	33.01	6.57
+ UNA-score (MSE)	60.46	**35.13**	84.14	61.69	**79.40**	**34.42**	4.98
+ UNA-score & binary	**61.24**	27.61	**85.40**	**62.88**	78.93	34.25	5.66
Gemma 4B	39.73	27.47	77.48	58.02	72.69	27.92	6.72
+ DPO	40.37	28.27	77.83	58.45	72.93	28.03	6.57
+ KTO	39.71	27.66	77.56	58.28	72.61	27.98	6.57
+ UNA-pairwise	40.37	28.27	77.83	58.45	72.93	28.03	6.57
+ UNA-binary (BCE)	40.79	26.19	77.95	58.36	72.61	27.95	6.87
+ UNA-score (MSE)	45.74	**29.80**	79.71	**61.26**	**73.48**	**28.72**	5.29
+ UNA-score & binary	**46.69**	27.54	**81.21**	59.30	72.14	28.49	**7.40**
Qwen 8B	52.19	42.27	79.71	67.92	77.03	47.21	29.46
+ DPO	51.78	40.21	79.81	67.41	76.87	47.48	28.32
+ KTO	52.07	43.19	79.58	67.66	76.56	47.18	28.25
+ UNA-pairwise	51.78	40.21	79.81	67.41	76.87	47.48	28.32
+ UNA-binary (BCE)	52.10	39.57	79.34	66.72	76.80	46.89	25.38
+ UNA-score (MSE)	64.44	53.46	81.00	**68.69**	**78.37**	**48.94**	34.59
+ UNA-score & binary	**66.92**	**64.14**	**82.92**	68.43	73.72	44.83	**42.60**

condition when Equation 8 is satisfied where λ is a constant.

$$\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}r(x,y)}} = \frac{1}{\lambda} \tag{8}$$

By rewriting this term, we can obtain the reward in term of the policy, i.e., $r(x,y) = \beta \log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right) + \beta \log(\lambda)$. In special case, $\lambda = 1$, it is simplified to $r(x,y) = \beta \log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right)$.

A more generalized UNA derivation is presented in Appendix D. This derivation confirms that the same fundamental relationship between the reward model and the policy model, as defined by DPO, is maintained (Equation 4). It also includes a discussion detailing the conditions $\lambda = 1$.

5 Experiments and Results

We evaluate UNA under two experimental settings. **Offline Setting** `mistralai/Mistral-7B-v0.1` (Jiang et al., 2023), `meta-llama/Llama-3.1-8B` (Grattafiori et al., 2024), `Qwen/Qwen3-8B-Base` (Yang et al., 2025), and `google/gemma-3-4b-pt` (Team et al., 2025) are utilized as the policy model, and the `HelpSteer2` dataset (Nvidia et al., 2024) is used as the alignment data, which have a prompt, chosen and rejected responses with corresponding scores that are labeled by humans from the perspectives of *helpfulness*, *correctness*, *coherence*, *complexity*, and *verbosity*. The combined score is computed as: $0.65 \times$ helpfulness $+ 0.8 \times$ correctness $+ 0.45 \times$ coherence, following (Wang et al., 2024c). For binary feedback, the chosen responses are regarded as desired responses with reward "+1" and rejected responses are regarded as undesired responses with reward "0". The score-based feedback includes a rating of 0 to 4 for each metric in `HelpSteer2`. The rewards are weighted and normalized and used as explicit feedback to align the LLM.

Low rank adaptation (LoRA) (Hu et al., 2021) is employed during the fine-tuning process with $r = 32$, where r denotes the ranks used in LoRA. Beam search is used to identify the optimal combination of β and learning rate. The selected configurations are listed. UNA-binary uses $\beta = 0.01$, while DPO, KTO, and UNA-score utilize $\beta = 0.03$. Furthermore, UNA-score employs a learning rate of 3×10^{-5}, whereas the other methods use a learning rate of 5×10^{-6}.

Online Setting We use policy model `Qwen/Qwen2-1.5B-Instruct` (Yang et al., 2024a) and `mistralai/Mistral-7B-Instruct` (Jiang et al., 2023), and reward model

Table 2: The comparison of UNA with RLHF using `HelpSteer2` prompts on different benchmarks.

Method	TruthfulQA	IFEval	HellaSwag	ARC	WinoGrande	MMLU-Pro	Math-Hard
Qwen2-1.5B-INST	45.93	22.20	66.72	43.94	66.06	**25.56**	5.40
+ RLHF	46.93	22.37	66.56	42.83	64.88	25.17	**5.48**
+ UNA	**47.08**	**24.78**	**66.98**	**44.28**	**65.27**	25.30	5.40
Mistral-7B-INST	**55.94**	38.46	75.99	**55.29**	73.72	24.53	**2.02**
+ RLHF	55.88	38.53	76.03	55.20	73.56	24.60	1.79
+ UNA	55.88	**39.17**	**76.61**	55.20	**74.03**	**24.87**	1.75

`Ray2333/GRM-Llama3.2-3B-rewardmodel-ft` (Yang et al., 2024b). We use prompts from `Helpsteer2`, excluding those longer than 512 tokens. In RLHF, prompts are used for response generation, reward estimation via a reward model, and policy updates through PPO. In contrast, online UNA uses the same prompts for response generation, implicit reward estimation by the policy, explicit reward estimation by the reward model, and policy updates via discrepancy minimization (e.g., MSE) between implicit and explicit rewards.

Similarly, we identify the optimal combination of parameters with beam search. For β, RLHF utilizes 10, while UNA uses 30, with both approaches employing the same learning rate of 3×10^{-6}.

After alignment, seven benchmark tasks are utilized to measure the performance, including TruthfulQA (Lin et al., 2022), IFEval (Zhou et al., 2023), HellaSwag (Zellers et al., 2019), ARC (Clark et al., 2018), WinoGrande (Sakaguchi et al., 2019), MMLU-pro (Wang et al., 2024b), and Math-Hard (Hendrycks et al., 2021). In addition to evaluating the model's selection capabilities from predefined candidate answers, Alpaca-eval is used to assess the model's ability to generate text responses.

5.1 Offline: Improvements over DPO & KTO

The results are presented in Table 1, along with the following insights. (1) UNA works with different forms of feedback. (2) On binary data, UNA outperforms DPO and KTO. (3) With score-based feedback, UNA outperforms UNA-binary, benefiting from the additional information provided by scalar scores. (4) Although the settings are not directly comparable due to differences in training data, UNA with both binary and score feedback outperforms the score-only variant on many tasks where binary data are from Helpsteer3 (Wang et al., 2025). We also conducted evaluations on AlpacaEval (Li et al., 2023), and UNA-score achieves the highest performance (seen Table 3 in the appendix E).

5.2 Online: Improvement and Simplification over RLHF

Table 2 shows that online UNA outperforms RLHF on most tasks. The problem of alignment tax (Ouyang et al., 2022) still exists on some tasks, as their performances decrease. Notably, by reformulating RLHF as a supervised learning problem and eliminating the value model, online UNA substantially reduces both memory consumption and training time. The training time for 20,000 steps with 8 80G A100 GPUs is around 8 hours for RLHF and 6.5 hours for online UNA with the same batch size. The comparison of RLHF with UNA on AlpacaEval (Table 4 in appendix E). further demonstrates the benefit of UNA over RLHF.

6 Conclusion

We introduce UNA, a unified alignment framework that supports training with diverse forms of feedback. By introducing an implicit reward model and showing that satisfying this condition yields the RLHF-optimal policy, UNA provides a unified foundation for alignment across feedback modalities. Through theoretical derivations and extensive empirical evaluation, we demonstrate that UNA effectively supports binary, pairwise, and score-based feedback. In particular, under score-based feedback, UNA can exploit pairwise difference information, leading to consistently better performance than DPO and KTO. Furthermore, we find that combining binary and score-based feedback yields additional performance gains over the score-only UNA variant across multiple tasks. Our experimental results also show that UNA outperforms RLHF in both effectiveness and training efficiency. Overall, our findings suggest that UNA provides a general and practical alignment framework that overcomes key limitations of RLHF/PPO, DPO, and KTO, while enabling robust and effective learning from heterogeneous feedback signals.

Limitations

There are some limitations for this work. In this work, the theoretical unification of GRPO, which greatly reduces the computational overhead of PPO, within the UNA framework remains unexplored. In addition, the datasets utilized are limited to English research datasets, while more experiments on multilingual industrial-level datasets should be conducted.

References

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. 2024. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms. *Preprint*, arXiv:2402.14740.

AI Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1.

Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. 2023. A general theoretical paradigm to understand learning from human preferences. *Preprint*, arXiv:2310.12036.

Yuntao Bai, Andy Jones, et al. 2022a. Training a helpful and harmless assistant with reinforcement learning from human feedback. *Preprint*, arXiv:2204.05862.

Yuntao Bai, Saurav Kadavath, et al. 2022b. Constitutional ai: Harmlessness from ai feedback. *Preprint*, arXiv:2212.08073.

Ju-Seung Byun, Jiyun Chun, Jihyung Kil, and Andrew Perrault. 2024. Ares: Alternating reinforcement learning and supervised fine-tuning for enhanced multi-modal chain-of-thought reasoning through diverse ai feedback. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4410–4430.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. 2023. Ultrafeedback: Boosting language models with high-quality feedback. *Preprint*, arXiv:2310.01377.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. Kto: Model alignment as prospect theoretic optimization. *Preprint*, arXiv:2402.01306.

Aaron Grattafiori, Abhimanyu Dubey, et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *Preprint*, arXiv:2103.03874.

Jiwoo Hong, Noah Lee, and James Thorne. 2024. Orpo: Monolithic preference optimization without reference model. *Preprint*, arXiv:2403.07691.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Chip Huyen. 2023. Rlhf: Reinforcement learning from human feedback. *Webseite der Autorin. URL: https://huyenchip. com/2023/05/02/rlhf. html [Zugriff: 17.10. 2023]*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Dahyun Kim, Yungi Kim, Wonho Song, Hyeonwoo Kim, Yunsu Kim, Sanghoon Kim, and Chanjun Park. 2024. sdpo: Don't use your data all at once. *Preprint*, arXiv:2403.19270.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. 2023. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *Preprint*, arXiv:2309.00267.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacaeval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. *Preprint*, arXiv:2109.07958.

Tianqi Liu, Zhen Qin, Junru Wu, Jiaming Shen, Misha Khalman, Rishabh Joshi, Yao Zhao, Mohammad Saleh, Simon Baumgartner, Jialu Liu, Peter J. Liu, and Xuanhui Wang. 2024. Lipo: Listwise preference optimization through learning-to-rank. *Preprint*, arXiv:2402.01878.

Hao Ma, Tianyi Hu, Zhiqiang Pu, Liu Boyin, Xiaolin Ai, Yanyan Liang, and Min Chen. 2024. Coevolving with the other you: Fine-tuning llm with sequential cooperative multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 37:15497–15525.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. *Preprint*, arXiv:2405.14734.

Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Andrea Michi, Marco Selvi, Sertan Girgin, Nikola Momchev, Olivier Bachem, Daniel J. Mankowitz, Doina Precup, and Bilal Piot. 2024. Nash learning from human feedback. *Preprint*, arXiv:2312.00886.

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

Nvidia, :, Bo Adler, et al. 2024. Nemotron-4 340b technical report. *Preprint*, arXiv:2406.11704.

OpenAI, Josh Achiam, et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

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

Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. 2024. Disentangling length from quality in direct preference optimization. *Preprint*, arXiv:2403.19159.

Shiva Kumar Pentyala, Zhichao Wang, Bin Bi, Kiran Ramnath, Xiang-Bo Mao, Regunathan Radhakrishnan, Sitaram Asur, Na, and Cheng. 2024. Paft: A parallel training paradigm for effective llm fine-tuning. *Preprint*, arXiv:2406.17923.

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

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Preprint*, arXiv:2305.18290.

Pierre Harvey Richemond, Yunhao Tang, Daniel Guo, Daniele Calandriello, Mohammad Gheshlaghi Azar, Rafael Rafailov, Bernardo Avila Pires, Eugene Tarassov, Lucas Spangher, Will Ellsworth, Aliaksei Severyn, Jonathan Mallinson, Lior Shani, Gil Shamir, Rishabh Joshi, Tianqi Liu, Remi Munos, and Bilal Piot. 2024. Offline regularised reinforcement learning for large language models alignment. *Preprint*, arXiv:2405.19107.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande: An adversarial winograd schema challenge at scale. *Preprint*, arXiv:1907.10641.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. 2024. Preference ranking optimization for human alignment. *Preprint*, arXiv:2306.17492.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Gemma Team, Aishwarya Kamath, et al. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Shuhe Wang, Shengyu Zhang, Jie Zhang, Runyi Hu, Xiaoya Li, Tianwei Zhang, Jiwei Li, Fei Wu, Guoyin Wang, and Eduard Hovy. 2024a. Reinforcement learning enhanced llms: A survey. *arXiv preprint arXiv:2412.10400*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024b. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Preprint*, arXiv:2406.01574.

Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. 2024c. Helpsteer2: Open-source dataset for training top-performing reward models. *ArXiv*, abs/2406.08673.

Zhilin Wang, Yi Dong, Jiaqi Zeng, Virginia Adams, Makesh Narsimhan Sreedhar, Daniel Egert, Olivier Delalleau, Jane Scowcroft, Neel Kant, Aidan Swope, et al. 2024d. Helpsteer: Multi-attribute helpfulness dataset for steerlm. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3371–3384.

Zhilin Wang, Jiaqi Zeng, Olivier Delalleau, Hoo-Chang Shin, Felipe Soares, Alexander Bukharin, Ellie Evans, Yi Dong, and Oleksii Kuchaiev. 2025. Helpsteer3-preference: Open human-annotated preference data across diverse tasks and languages. *Preprint*, arXiv:2505.11475.

Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. 2024. Self-play preference optimization for language model alignment. *Preprint*, arXiv:2405.00675.

Jing Xu, Andrew Lee, Sainbayar Sukhbaatar, and Jason Weston. 2024. Some things are more cringe than others: Iterative preference optimization with the pairwise cringe loss. *Preprint*, arXiv:2312.16682.

An Yang, Anfeng Li, et al. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

An Yang, Baosong Yang, et al. 2024a. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. 2024b. Regularizing hidden states enables learning generalizable reward model for llms. In *Advances in Neural Information Processing Systems*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. 2025. Dapo: An open-source llm reinforcement learning system at scale. *Preprint*, arXiv:2503.14476.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. *Preprint*, arXiv:2401.10020.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. Rrhf: Rank responses to align language models with human feedback without tears. *Preprint*, arXiv:2304.05302.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

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

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. 2025. Group sequence policy optimization. *Preprint*, arXiv:2507.18071.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *Preprint*, arXiv:2311.07911.

Banghua Zhu, Michael I Jordan, and Jiantao Jiao. 2024. Iterative data smoothing: Mitigating reward overfitting and overoptimization in rlhf. *arXiv preprint arXiv:2401.16335*.

A Default Notation

x**:** prompt to LLM

y_w : desired response

y_l : undesired response

$P(y_w > y_l|x)$: the probability of desired response over undesired response

$r_\phi(x, y)$: the explicit reward

$r_\theta(x, y)$: the implicit reward

$s_\phi(x, y)$: the explicit score: normalized explicit reward

$s_\theta(x, y)$: the implicit score: normalized implicit reward

D_{KL} : KL divergence

π_θ : LLM policy to be aligned

π_{ref} : reference policy for LLM alignment

$g(\cdot)$: any function that measures the difference between implicit and explicit reward functions

B DPO: Relationship between optimal policy and reward function

The objective of RLHF / DPO is shown in Equation 1. From the objective, the relationship between optimal reward and optimal policy can be derived in Equation 4 where $Z(x) = \sum_y \pi_{\text{ref}}(y|x)e^{\left(\frac{1}{\beta}r_\theta(x,y)\right)}$. The illustration for deriving DPO is shown in Equation 9.

$$
\begin{aligned}
\pi^*_\theta(y|x) &= \max_{\pi_\theta} \mathbb{E}_{x\sim D}\left[\mathbb{E}_{y\sim\pi_\theta(y|x)} r_\theta(x,y) - \beta D_{\text{KL}}\left(\pi_\theta(y|x)\|\pi_{\text{ref}}(y|x)\right)\right] \\
&= \max_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[r(x,y) - \beta\log\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right]\right\} \\
&= \min_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[\log\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} - \frac{1}{\beta}r(x,y)\right]\right\} \\
&= \min_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[\log\left(\frac{\pi_\theta(y|x)}{\frac{1}{Z(x)}\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}r(x,y)}}\right) - \log\left(Z(x)\right)\right]\right\} \\
&= \min_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[\log\left(\frac{\pi_\theta(y|x)}{\frac{1}{Z(x)}\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}r(x,y)}}\right)\right] - \log\left(Z(x)\right)\right\} \\
&= \min_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{D_{KL}\left(\pi_\theta(y|x)\|\frac{1}{Z(x)}\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}r(x,y)}\right) - \log\left(Z(x)\right)\right\}
\end{aligned}
\tag{9}
$$

The objective function is minimized when $D_{KL}\left(\pi_\theta(y|x)||\frac{1}{Z(x)}\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}r(x,y)}\right) = 0$, and this is equivalent to $\pi_\theta(y|x) = \frac{1}{Z(x)}\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}r(x,y)}$. By rewriting, the reward model can be expressed in term of the current policy as shown in Equation 4.

However, the term $Z(x)$ cannot be computed as it needed to be computed by summing all candidate responses y. DPO avoids this problem by subtracting the rewards of desired and undesired responses $r(x,y_w) - r(x,y_l) = \beta\left[\log\left(\frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)}\right) - \log\left(\frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}\right)\right]$. In addition, the authors argue "We say that two reward functions $r(x,y)$ and $r'(x,y)$ are equivalent iff $r(x,y) - r'(x,y) = f(x)$ for some function f". However, rigorous proof cannot be provided and it is only provided that $r(x,y)$ and $r'(x,y)$ induce the same optimal policy. For Lipo, $r(x,y) = \beta\log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right)$ is directly utilized as rewards for listwise responses and KTO estimates $Z(x)$ by averaging over multiple samples.

C Derivation of log-sum inequality

Jensen inequality. For a real convex function φ, numbers $x_1, x_2, \ldots, x_n$ in its domain, and positive weights a_i, Jensen's inequality can be stated as in Equation 10:

$$\frac{\sum_{i=1}^{n} a_i \varphi(x_i)}{\sum_{i=1}^{n} a_i} \geq \varphi\left(\frac{\sum_{i=1}^{n} a_i x_i}{\sum_{i=1}^{n} a_i}\right) \tag{10}$$

Proof of log-sum inequality. Firstly, define $f(x) = x\log(x)$. Then, $f'(x) = 1 + \log(x)$ and $f''(x) = \frac{1}{x}$. For the domain $x > 0$, $f''(x) > 0$. As a result, $f(x) = x\log(x)$ is a convex function and satisfies Jensen's inequality. Then, the log-sum inequality could be derived in Equation 11.

$$\begin{aligned}
\sum_{i=1}^{n} a_i \log\left(\frac{a_i}{b_i}\right) &= \sum_{i=1}^{n} b_i f\left(\frac{a_i}{b_i}\right) \\
&= b\sum_{i=1}^{n} \frac{b_i}{b} f\left(\frac{a_i}{b_i}\right) \\
&= b\frac{\sum_{i=1}^{n} b_i f\left(\frac{a_i}{b_i}\right)}{\sum_{i=1}^{n} b_i} \\
&\geq bf\left[\frac{\sum_{i=1}^{n} b_i \frac{a_i}{b_i}}{\sum_{i=1}^{n} b_i}\right] \\
&= bf\left(\frac{a}{b}\right)
\end{aligned} \tag{11}$$

D Mathematical Proof of the Generalized UNA and Its Relationship with DPO

Starting from the same objective in Equation 1, it can be simplified as shown in Equation 12.

$$
\begin{aligned}
\pi_\theta^*(y|x) &= \max_{\pi_\theta} \mathbb{E}_{x\sim D}\left[\mathbb{E}_{y\sim\pi_\theta(y|x)} r_\theta(x,y) - \beta D_{\text{KL}}\left(\pi_\theta(y|x)\|\pi_{\text{ref}}(y|x)\right)\right] \\
&= \max_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[r(x,y) - \beta\log\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right]\right\} \\
&= \beta\max_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[\frac{1}{\beta}r(x,y) - \log\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right]\right\} \\
&= \beta\max_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[-\log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}r(x,y)}}\right)\right]\right\} \\
&= \beta\max_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[-\log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}(r(x,y)-f(x))}}\right) + \frac{1}{\beta}f(x)\right]\right\} \\
&= \beta\max_{\pi_\theta} \mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[-\log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}(r(x,y)-f(x))}}\right)\right] + \frac{1}{\beta}f(x)\right\}
\end{aligned}
\tag{12}
$$

Based on the log-sum inequality in Equation 5, the term can be further simplified as shown in Equation 13 because both $\pi_\theta(y|x)$ and $\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}(r(x,y)-f(x))}$ are non-negative.

$$
\begin{aligned}
&\beta\mathbb{E}_{x\sim D}\left\{\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[-\log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}(r(x,y)-f(x))}}\right)\right] + \frac{1}{\beta}f(x)\right\} \\
&= \beta\mathbb{E}_{x\sim D}\left\{-\sum_y\left[\pi_\theta(y|x)\log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}(r(x,y)-f(x))}}\right)\right] + \frac{1}{\beta}f(x)\right\} \\
&\le \beta\mathbb{E}_{x\sim D}\left\{\left[-\left(\sum_y\pi_\theta(y|x)\right)\log\left(\frac{\sum_y\pi_\theta(y|x)}{\sum_y\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}(r(x,y)-f(x))}}\right)\right] + \frac{1}{\beta}f(x)\right\} \\
&= \beta\mathbb{E}_{x\sim D}\left\{\left[-1\log\left(\frac{1}{\sum_y\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}(r(x,y)-f(x))}}\right)\right] + \frac{1}{\beta}f(x)\right\} \\
&= \beta\mathbb{E}_{x\sim D}\left\{\log\left(\mathbb{E}_{y\sim\pi_{\text{ref}}(y|x)}e^{\frac{1}{\beta}(r(x,y)-f(x))}\right) + \frac{1}{\beta}f(x)\right\}
\end{aligned}
\tag{13}
$$

As a result, the maximum value of the objective function $\max_{\pi_\theta}\mathbb{E}_{x\sim D}\left[\mathbb{E}_{y\sim\pi_\theta(y|x)}r_\theta(x,y) - \beta D_{\text{KL}}\left(\pi_\theta(y|x)\|\pi_{\text{ref}}(y|x)\right)\right]$ in Equation 12 is $\beta\mathbb{E}_{x\sim D}\left\{\log\left(\mathbb{E}_{y\sim\pi_{\text{ref}}(y|x)}e^{\frac{1}{\beta}(r(x,y)-f(x))}\right) + \frac{1}{\beta}f(x)\right\}$ in Equation 13, and this inequality reaches the equality condition when Equation 14 is satisfied where λ is a constant.

$$
\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)e^{\frac{1}{\beta}(r(x,y)-f(x))}} = \frac{1}{\lambda}
\tag{14}
$$

By rewriting this term, we can obtain the reward in term of the policy as shown in Equation 15. In special case, $f(x) = \beta\log(\lambda) = 0$, it is simplified to $r(x,y) = \beta\log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right)$. The condition $f(x) = \beta\log(\lambda) = 0$ refers that implicit and explicit reward models are exactly the same.

$$
\begin{aligned}
r(x,y) &= \beta\log\left(\frac{\lambda\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right) + f(x) \\
&= \beta\log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right) + f(x) + \beta\log(\lambda)
\end{aligned}
\tag{15}
$$

When plugging Equation 14 in Equation 13, the upper bound can be simplified into a constant $\beta \log(\lambda) + \mathbb{E}_{x\sim D}(f(x))$ as shown in Equation 16.

$$
\begin{aligned}
&\beta \mathbb{E}_{x\sim D} \left\{ \log \left(\mathbb{E}_{y\sim \pi_{\text{ref}}(y|x)} e^{\frac{1}{\beta}(r(x,y)-f(x))} \right) + \frac{1}{\beta} f(x) \right\} \\
&= \beta \mathbb{E}_{x\sim D} \left\{ \log \left(\mathbb{E}_{y\sim \pi_{\text{ref}}(y|x)} \frac{\lambda \pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right) + \frac{1}{\beta} f(x) \right\} \\
&= \beta \mathbb{E}_{x\sim D} \left\{ \log \left(\mathbb{E}_{y\sim \pi_\theta(y|x)} \lambda \right) + \frac{1}{\beta} f(x) \right\} \\
&= \beta \mathbb{E}_{x\sim D} \left\{ \log \left(\lambda \right) + \frac{1}{\beta} f(x) \right\} \\
&= \beta \log(\lambda) + \mathbb{E}_{x\sim D}(f(x))
\end{aligned}
\tag{16}
$$

When desired to generalize this into "infinite dimension", another constraint needs to be added, i.e., $\sum_y \pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y)-f(x))}$ should be finite. Then, $f(x)$ is further restricted to $f(x) > \max[r(x,y)]$ with normalization on $r(x,y)$ in advance. Eventually, $\sum_y \pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y)-f(x))} < \sum_y \pi_{\text{ref}}(y|x) = 1$, which will be finite.

Lastly, the relationship between UNA and DPO will be established. Under the optimal condition of UNA, as defined in Eq. 14, the probability $\pi_\theta(y|x)$ can be expressed as $\pi_\theta(y|x) = \frac{1}{\lambda} \pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y)-f(x))}$. Since $\pi_\theta(y|x)$ represents a valid probability distribution, it must satisfy the normalization condition $\sum_y \pi_\theta(y|x) = 1$. Consequently, this can be rewritten as shown in Eq. 17.

$$
\begin{aligned}
1 &= \sum_y \pi_\theta(y|x) \\
&= \sum_y \frac{1}{\lambda} \pi_{\text{ref}}(y|x) e^{\frac{1}{\beta}(r(x,y)-f(x))} \\
&= \sum_y \frac{\pi_{\text{ref}}(y|x) e^{\frac{1}{\beta} r(x,y)}}{\lambda e^{\frac{1}{\beta} f(x)}} \\
&= \frac{\sum_y \pi_{\text{ref}}(y|x) e^{\frac{1}{\beta} r(x,y)}}{\lambda e^{\frac{1}{\beta} f(x)}} \\
&= \frac{Z(x)}{\lambda e^{\frac{1}{\beta} f(x)}}
\end{aligned}
\tag{17}
$$

From Eq. 17, we can derive $Z(x) = \lambda e^{\frac{1}{\beta} f(x)}$. When apply $\log$ on both sides, $\beta \log(Z(x)) = \beta \log(\lambda e^{\frac{1}{\beta} f(x)}) = f(x) + \beta \log(\lambda)$. The implicit reward function of DPO and UNA is unified: $r(x,y) = \beta \log \left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right) + f(x) + \beta \log(\lambda) = \beta \log \left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right) + \beta \log(Z(x))$.

From Eq. 17, we derive the expression for $Z(x)$ as $Z(x) = \lambda e^{\frac{1}{\beta} f(x)}$. Taking the natural logarithm on both sides yields $\beta \log(Z(x)) = \beta \log \left(\lambda e^{\frac{1}{\beta} f(x)} \right) = f(x) + \beta \log(\lambda)$. Thus, the implicit reward function for DPO and UNA can be unified as $r(x,y) = \beta \log \left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right) + f(x) + \beta \log(\lambda) = \beta \log \left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right) + \beta \log(Z(x))$.

E Result of UNA on Alpacaeval

Table 3: The comparison of UNA with DPO, KTO, considering pairwise, binary, and score-based data on AlpacaEval using `HelpSteer2` as fine-tuning data

Method	Alpacaeval LC WR	Method	Alpacaeval LC WR
Mistral	0.31	Llama	0.25
+ DPO	3.67	+ DPO	2.09
+ KTO	4.46	+ KTO	4.17
+ UNA-pairwise	3.67	+ UNA-pairwise	2.09
+ UNA-binary (BCE)	7.41	+ UNA-binary (BCE)	3.96
+ UNA-score (MSE)	**8.78**	+ UNA-score (MSE)	**7.87**

Table 4: The comparison of UNA with RLHF using `HelpSteer2` prompts on AlpacaEval

Method	AlpacaEval LC WR	Method	AlpacaEval LC WR
Qwen2-1.5B-INST	1.06	Mistral-7B-INST	10.31
+ RLHF	0.66	+ RLHF	10.15
+ UNA	**1.63**	+ UNA	**10.54**