WILDREWARD: Learning Reward Models from In-the-Wild Human Interactions

Hao Peng[1]*, **Yunjia Qi**[1], **Xiaozhi Wang**[2], **Zijun Yao**[1], **Lei Hou**[1], **Juanzi Li**[1]†
[1]Department of Computer Science and Technology;
[2]Shenzhen International Graduate School,
Tsinghua University
{peng-h24}@mails.tsinghua.edu.cn

Abstract

Reward models (RMs) are crucial for the training of large language models (LLMs), yet they typically rely on large-scale human-annotated preference pairs. With the widespread deployment of LLMs, in-the-wild interactions have emerged as a rich source of implicit reward signals. This raises the question: *Can we develop reward models directly from in-the-wild interactions?* In this work, we explore this possibility by adopting WildChat as an interaction source and proposing a pipeline to extract reliable human feedback, yielding 186k high-quality instances for training WILDREWARD via ordinal regression directly on user feedback without preference pairs. Extensive experiments demonstrate that WILDREWARD achieves comparable or even superior performance compared to conventional reward models, with improved calibration and cross-sample consistency. We also observe that WILDREWARD benefits directly from user diversity, where more users yield stronger reward models. Finally, we apply WILDREWARD to online DPO training and observe significant improvements across various tasks. Code and data are released at https://github.com/THU-KEG/WildReward.

1 Introduction

Reward models (RMs) are crucial for the training and inference-time scaling of large language models (LLMs). They are typically used to model human preferences and trained on large-scale human-annotated preference pairs (Ouyang et al., 2022). Prior work has primarily focused on collecting preference pairs (Wang et al., 2024d, 2025b; Liu et al., 2025a), requiring substantial annotation efforts.

With the widespread deployment of LLMs, numerous in-the-wild interactions with humans have emerged, such as human-LLM conversations (Zhao et al., 2024; Zheng et al., 2024). These interactions typically contain rich human feedback. For example, as shown in Figure 1, humans provide textual feedback regarding the previous model response. This feedback directly reflects the response quality and authentic human preferences, thereby naturally serving as training data for reward models. Despite the large scale and richness of these interactions, their utility remains under-explored. This raises a critical question: *Can we develop reward models directly from these in-the-wild interactions?*

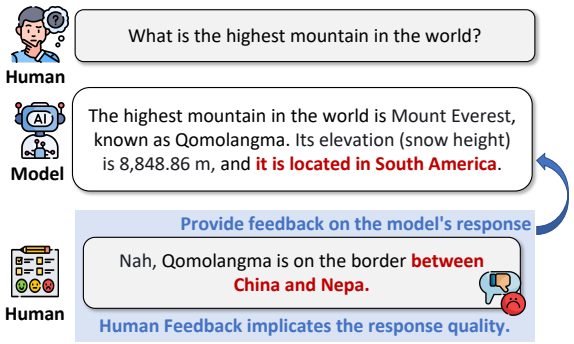

Figure 1: Illustration of a human-LLM interaction with implicit feedback signals in the conversation. The user provides valid feedback and identifies an error.

In this work, we explore training reward models using in-the-wild human interactions. Specifically, we leverage WildChat (Zhao et al., 2024), a large-scale human-LLM conversation dataset. We first conduct a preliminary analysis of user queries within WildChat, which reveals two primary observations and challenges: (1) **Feedback sparsity**. Feedback is mostly implicit. Approximately 82% of follow-up queries do not explicitly convey feedback or preference regarding the previous response. Notably, explicit positive feedback is particularly scarce, which accounts for only 1%. (2) **Feedback noise**. User feedback is prone to noise, especially in safety scenarios. For instance, when an LLM correctly refuses a sensitive question, the user may provide negative feedback, which is unjustified. To

* Work done during an internship at Zhipu AI
† Corresponding author: Juanzi Li

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 18932–18948

mitigate these issues, we propose an automated pipeline to filter noise and extract reliable feedback. Specifically, we classify user feedback into five levels of satisfaction, indicating response quality, including explicit rejection, error correction, neutral ambiguity, positive engagement, and explicit satisfaction. We adopt gpt-oss-120b (Agarwal et al., 2025) for automatic classification and adopt a conservative strategy that defaults to neutral ambiguity in the absence of strong evidence to minimize label noise. Furthermore, we propose a two-stage refinement strategy to extract implicit feedback and mitigate noise: (1) Implicit feedback mining, which recovers implicit positive signals; (2) Refusal validation, which validates justified safety refusals. Finally, we exclude the neutral ambiguity subset, resulting in WILDFB, which contains 186k instances, each consisting of a conversation history, a user query, a response, and a label indicating the response quality. We verify data quality through sampled 100 instances and observe little noise. For training reward models, we adopt the ordinal regression objective (Wang et al., 2025a) to explicitly learn accurate relative rankings of user feedback, resulting in our reward model WILDREWARD.

We conduct extensive experiments to validate the efficacy of WILDREWARD. We first evaluate WILDREWARD on standard and widely used reward model benchmarks, including RewardBench (Lambert et al., 2025), RM-Bench (Liu et al., 2025e), PPE (Frick et al., 2025), and JudgeBench (Tan et al., 2025). We find that WILDREWARD achieves performance comparable or superior to conventional reward models. This demonstrates that WILDREWARD effectively captures general human preferences without any dedicated human-annotated preference pairs. We conduct extensive analyses and draw the following conclusions: (1) Data strategy. Both implicit feedback mining and refusal validation strategies are beneficial. Furthermore, WILDREWARD benefits from user diversity, as more users yield stronger models. (2) Calibration. WILDREWARD is well-calibrated. By using the score margin between chosen and rejected responses as a proxy for confidence, we observe a strong positive correlation between confidence and accuracy. This indicates that WILDREWARD can be integrated with more powerful LLMs (Xu et al., 2025b) or humans to produce more accurate reward signals. (3) Cross-sample consistency. WILDREWARD exhibits strong global score calibration and provides a unified and meaningful score for assessing response quality. We introduce an implicit feedback prediction task designed to predict binary user reception (positive or negative) based on the conversation context, user query, and response. WILDREWARD achieves significantly higher ROC-AUC scores. (4) Application in DPO training. WILDREWARD effectively guides policy model training. When applied to online DPO (Rafailov et al., 2023), the trained model achieves significant improvements across various downstream tasks, including mathematical reasoning, instruction following, and creative writing. In conclusion, this work demonstrates the potential of real-world interactions and highlights a promising direction for leveraging these rapidly growing resources. We encourage more research efforts to explore this area in the future.

2 Methodology

We adopt **WildChat** (Zhao et al., 2024), a human-LLM conversation dataset, as primary interaction source. This section presents a preliminary analysis of WildChat (§ 2.1), the automated pipeline used to extract feedback and construct a high-quality dataset **WILDFB** (§ 2.2), and the training method (§ 2.3) for the reward model **WILDREWARD**.

2.1 Preliminary Analysis

We conduct a preliminary analysis of WildChat. Specifically, we first sample $10,000$ instances, each consisting of a conversation history, the user query, the corresponding model response, and the user's follow-up query. We then analyze the follow-up query, as it potentially reflects the user preference or the quality of the preceding response. We adopt gpt-oss-120b (Agarwal et al., 2025) to automatically classify the follow-up query into three feedback categories: *Negative*, *Neutral*, and *Positive*. We observe that valid feedback is sparse: approximately 82% of queries are *Neutral*, i.e., do not explicitly express feedback, while 17% contain negative feedback, e.g., pointing out errors. Only 1% are *Positive*, which is expected as users rarely express explicit gratitude in natural interactions.

We further sample 200 instances from this classified set and conduct a manual inspection. In the *Neutral* category, approximately 86% are new requests unrelated to the previous response, while the remaining 14% are relevant follow-up questions, which may serve as implicit feedback. In the *Negative* and *Positive* categories, the classification accuracy of gpt-oss-120b is high, where most

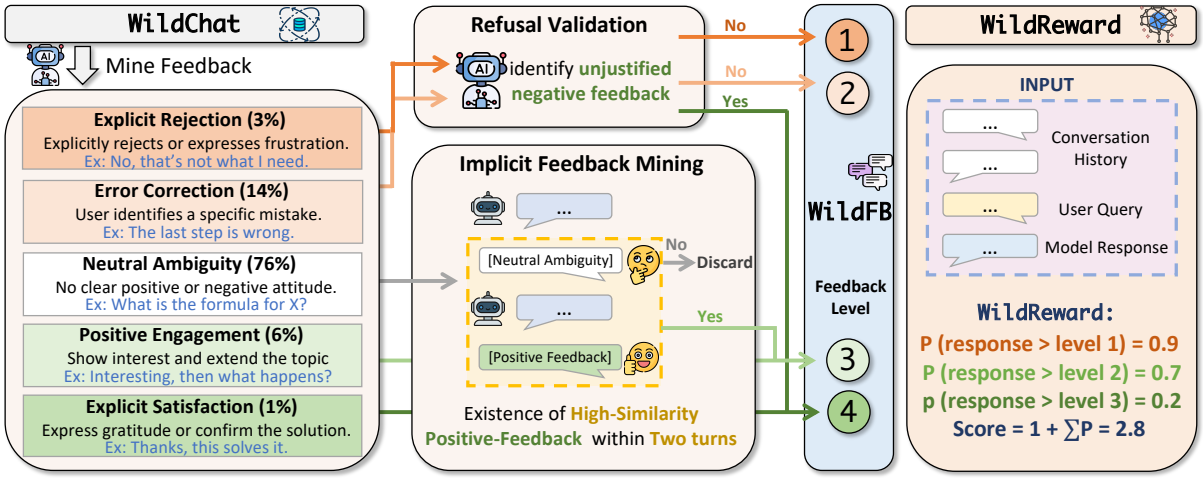

Figure 2: Overview of the proposed pipeline for extracting human feedback from in-the-wild conversations.

follow-up queries express corresponding negative or positive feedback. However, we still find a type of noise: an LLM correctly refuses a sensitive question, yet the user provides negative feedback, which is unjustified. In conclusion, while in-the-wild conversations contain valuable feedback signals, they present challenges regarding sparsity and noise. It requires a well-designed pipeline for extracting valid feedback. More details are in Appendix A.

2.2 WILDFB Dataset Construction

Based on the preliminary analysis, we propose an automated pipeline to extract valid feedback and mitigate noise from WildChat. The overall framework is shown in Figure 2. To capture fine-grained feedback signals, we classify user feedback into five levels of satisfaction, indicating response quality, and the corresponding descriptions are shown in Figure 2. Inspired by our initial findings in § 2.1 that relevant follow-up questions may signal active user involvement, we consider such engagement as a form of implicit positive feedback. Therefore, we introduce the *Positive Engagement* category to collect more instances with positive feedback. Given the reliability of gpt-oss-120b verified in § 2.1, we employ it for automatic feedback classification. To minimize label noise, we adopt a conservative strategy that defaults to *Neutral Ambiguity* in the absence of strong evidence. The distribution of extracted feedback is shown in Figure 2.

To further extract valid feedback and mitigate labeling noise, we propose a two-stage refinement strategy: (1) **Implicit feedback mining**. As observed in § 2.1, the *Neutral Ambiguity* category contains implicit positive feedback. To exploit this, we find that if a user provides positive feedback in adjacent turns within a coherent conversation context, e.g., the same topic, the intermediate responses are likely also of high quality with positive feedback. We analyze 20 randomly sampled instances and find that 90% support this intuition. Consequently, we mine *Neutral Ambiguity* instances where the user query shares high semantic similarity (> 0.6) with a positive-feedback query within a two-turn window. The semantic similarity is computed using the cosine similarity of sentence embeddings derived from all-MiniLM-L6-v2[1]. These instances are reclassified as positive feedback, yielding approximately $12,310$ additional samples and expanding the positive feedback subset by 29%. (2) **Refusal Validation**. As observed in § 2.1, negative user feedback in safety contexts may introduce noise, as users often respond negatively even when the model correctly refuses sensitive queries. To address this, we employ gpt-oss-120b to analyze instances of the *Explicit Rejection* and *Error Correction* categories, determining whether the negative feedback is unjustified given the valid refusal, which finally fixes about 572 such errors. Although the scale is small, this correction significantly improves performance on safety subsets in reward model benchmarks, as demonstrated in our ablation study (§ 3.3). Finally, we **exclude** the remaining *Neutral Ambiguity* subset as it lacks clear feedback signals and obtain **WILDFB**, a high-quality dataset comprising approximately 186k instances across four feedback categories, each containing a conversation history, a user query, a response, and a

[1] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

label indicating the response quality. More data construction details are placed in Appendix A.

2.3 Training WILDREWARD

Conventional reward models typically rely on a large-scale set of preference pairs and are trained using the Bradley-Terry model (Bradley and Terry, 1952). In contrast, WILDFB consists of point-wise user feedback labels that exhibit an intrinsic ordinal ranking structure. For instance, *Positive Engagement* reflects a higher level of user satisfaction or response quality than the *Error Correction* category. To leverage this ranking structure, we map the four feedback categories (excluding the *Neutral Ambiguity* subset) to discrete quality scores ranging from *1* to *4*, where *1* represents *Explicit Rejection* and *4* represents *Explicit Satisfaction*. We train a reward model to learn this hierarchy, enabling it to produce reliable ranking scores for model responses. Specifically, we adopt an ordinal regression objective for training, which is demonstrated effective to explicitly learn the accurate relative rankings (Wang et al., 2025a). Compared to standard regression, ordinal regression models the inherent order of feedback without assuming uniform intervals and provide probabilistic outputs for confidence filtering.

Formally, given a training instance consisting of a dialogue history c, a user query q, the corresponding response s, and an associated feedback label $y \in \{1, 2, 3, 4\}$, we define the model input as $x = (c, q, s)$. The reward model is trained by minimizing the following objective:

$$\mathcal{L} = -\sum_{k=1}^{K-1} \big[\mathbb{I}(y > k) \log P(y > k|x; \theta) + (1 - \mathbb{I}(y > k)) \log(1 - P(y > k|x; \theta))\big]$$

Here, θ represents the learnable parameters of the reward model. K is the total number of ordinal categories, which is 4 in this case. $\mathbb{I}$ denotes the indicator function. During inference, we compute a continuous reward score for each input x. The final reward score is computed as follows:

$$R(x) = 1 + \sum_{k=1}^{K-1} P(y > k|x; \theta)$$

This score represents the expected value of the predicted feedback. As a probabilistic measure, it can be effectively used for confidence filtering.

3 Experiment

This section presents the experimental setup (§ 3.1) and evaluation results on standard reward benchmarks (§ 3.2). We further analyze our data construction strategy (§ 3.3), investigate the properties of WILDREWARD (§§ 3.4 and 3.5), and demonstrate its effectiveness in guiding DPO training (§ 3.6).

3.1 Experimental Setup

Regarding **implementation details**, we use Qwen3 4B and 8B (Yang et al., 2025) as backbone LLMs. We train these models on WILDFB for one epoch to develop the reward models WILDREWARD-4B and WILDREWARD-8B, with a training batch size of 512 and a learning rate of 1×10^{-5}. Regarding investigated **baselines**, we adopt various representative reward models for comparison, which are typically trained on large-scale preference pairs, including Llama-3-OffsetBias-RM-8B (Park et al., 2024), ArmoRM (Wang et al., 2024a), Athene-RM (Frick et al., 2024), Skywork-Reward (Liu et al., 2024), InternLM2-Reward (Cai et al., 2024), INF-ORM-Llama3.1-70B (Minghao Yang, 2024), and Llama-3.1-Nemotron-70B (Wang et al., 2024c). Regarding **evaluation benchmarks**, we adopt standard and widely used reward model benchmarks, including RewardBench (Lambert et al., 2025), RM-Bench (Liu et al., 2025e), PPE (Frick et al., 2025), and JudgeBench (Tan et al., 2025). We leverage all three difficulty levels (easy, normal, hard) of RM-Bench. RewardBench, RM-Bench, PPE Human, and JudgeBench use a binary choice setting to select the chosen response, while PPE Correctness employs a Best-of-N evaluation setting, which aligns with test-time scaling evaluation. The evaluation benchmarks cover diverse domains, including creative writing, instruction following, mathematics, commonsense reasoning, coding, and safety.

3.2 Reward Model Benchmarking Result

The experimental results are presented in Table 1, we can observe that: (1) WILDREWARD achieves comparable or even superior performance to conventional reward models without human-annotated preference pairs. Notably, WILDREWARD, with only 4 or 8 billion parameters, surpasses the performance of much larger 70B reward models. It demonstrates that we can train reward models directly from human feedback, rather than relying on preference pairs and also confirms that there are valid human feedback exists within in-the-wild in-

Model	RewardBench	RM-Bench			PPE		JudgeBench
		Easy	Normal	Hard	Human	Correctness	
ArmoRM-Llama3-8B-v0.1	90.4	80.4	71.5	55.8	60.6	60.6	59.7
Athene-RM-8B	84.8	89.8	76.6	51.4	**64.6**	62.0	70.1
Llama-3-OffsetBias-RM-8B	89.0	83.9	73.2	56.9	59.2	64.1	63.5
Skywork-Reward-Llama-3.1-8B-v0.2	93.1	70.5	74.2	49.3	62.2	62.5	62.9
Internlm2-20b-reward	90.2	79.4	74.2	62.8	61.0	63.0	64.3
Skywork-Reward-Gemma-2-27B-v0.2	94.3	88.9	71.9	42.1	63.6	61.9	66.5
Llama-3.1-Nemotron-70B	93.9	**92.2**	76.5	47.8	64.2	63.2	65.8
INF-ORM-Llama3.1-70B	**95.1**	92.1	**80.0**	54.0	64.2	64.4	**70.2**
WILDREWARD-4B	83.6	82.0	77.0	68.6	61.6	63.6	61.1
WILDREWARD-8B	86.0	83.5	78.4	**69.7**	62.5	**65.6**	66.0

Table 1: Experimental results (%) on several representative reward model benchmarks. PPE Human and Correctness denote the human and correctness preference subset, respectively. The highest score in each column is in **bold**.

teractions. Furthermore, this approach is inherently data scalable, given the vast amount of such interaction data available in the real world. One may collect more human feedback and develop more advanced reward models. (2) On RM-Bench Hard and PPE Correctness, WILDREWARD demonstrates superior performance. RM-Bench Hard specifically evaluates robustness to superficial cues, such as irrelevant styles and length bias, and the ability to select factual responses (Liu et al., 2025e). PPE Correctness also evaluates objective factual accuracy. The superior results on these two benchmarks demonstrate that WILDREWARD is more robust to superficial biases. This observation is reasonable, as humans typically provide negative feedback to verbose yet incorrect answers in real-world interactions. (3) WILDREWARD-8B consistently outperforms WILDREWARD-4B, which indicates that larger models can more effectively leverage in-the-wild interaction data. This trend also demonstrates the effectiveness of WILDFB and suggests the potential for model scaling. However, due to computational constraints, we leave the exploration of further scaling to future work. In conclusion, the experimental results demonstrate the significant potential of training reward models from in-the-wild human interactions. It also validates that vast amounts of human interactions are a potentially vital resource for the future of LLM training.

3.3 Analysis on Data Strategy

We analyze the impact of different data curation strategies on the resulting reward models. We first conduct an ablation study to evaluate our proposed data strategies described in § 2.2: implicit feedback mining and refusal validation. Specifically, we exclude the corresponding data and retrain the model, while keeping all other configurations identical. The results are presented in Table 2. "SRF" denotes the Safety-Refusal subset, where the prompt is sensitive and the refusal response is the chosen one. "SRP" denotes the Safety-Response subset, where the prompt is normal and the standard, helpful response is the chosen one. We can observe that ablating either data strategy results in a significant performance degradation, which demonstrates that both the two strategies are effective for collecting high-quality data. The impact is particularly significant on the safety subset, where SRF performance drops by 60% when the Refusal Validation strategy is excluded. This is expected, as its removal introduces noise by labeling valid refusals with negative feedback, which induces a bias against refusals. Notably, the Refusal Validation strategy yields only 572 instances, but it has a substantial impact on the final experimental results. This suggests that the safety boundary of reward models is sensitive, requiring further efforts to enhance its robustness. It further suggests that in-the-wild human interactions contain much subtle noise, requiring a robust pipeline to extract valid human feedback.

Model	Chat	Math	Code	SRF	SRP
WILDREWARD-4B	79.3	75.6	65.8	90.4	72.0
w/o Feedback Mining	77.3	73.6	65.5	68.3	77.5
w/o Refusal Validation	80.0	74.6	64.8	28.5	97.0
w/o All	77.0	74.5	64.8	39.7	95.5

Table 2: Ablation study results (%) on RM-Bench Normal. The variants "w/o Feedback Mining" and "w/o Refusal Validation" mean models trained without the corresponding data. "SRF" and "SRP" denote Safety-Refusal and Safety-Response subsets, respectively.

A significant advantage of in-the-wild human

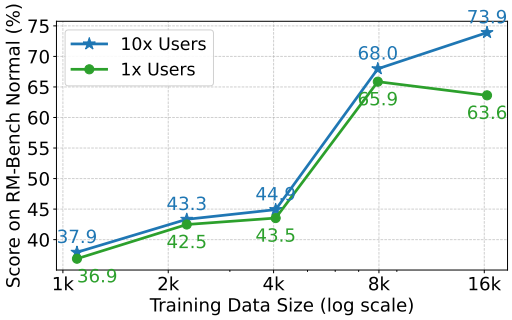

Figure 3: Performance on RM-Bench Normal across varying data sizes and user counts.

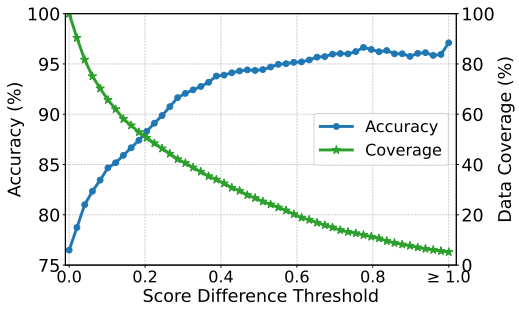

Figure 4: Accuracy and data coverage against score difference threshold, filtering for predictions where the chosen-rejected score margin exceeds the threshold. The results are reported on RM-Bench Normal.

interactions lies in the user diversity, rather than a limited group of human annotators. To investigate this, we conduct an analysis on the impact of user diversity on training reward models. We measure user diversity by the number of unique users. We construct two series of datasets of identical size, where one contains ten times the number of unique users as the other. The results of trained reward models are shown in Figure 3. We can observe that: (1) Generally, model performance improves with the training data size, which aligns with the data scaling law (Kaplan et al., 2020). (2) For a given data size, models trained on data from a larger number of users consistently perform better. It demonstrates that the model benefits directly from user diversity, as more diverse users may provide more robust feedback. In conclusion, scaling up the size and diversity of human interactions is promising for developing advanced reward models.

3.4 Analysis on Calibration

A robust reward model should be well-calibrated, meaning that its prediction accuracy should correlate positively with its confidence. Here, we focus on binary classification tasks, which is a widely-used setting in reward model benchmarks to select the better of two given responses (chosen and rejected). As detailed in § 2.3, we employ ordinal regression to train our reward models. A key advantage of this approach is that it directly leverages probabilities as scores, which theoretically promotes good calibration. In this section, we explore the calibration properties of WILDREWARD using the RM-Bench Normal dataset. We use the score margin between chosen and rejected responses as a confidence proxy, and apply Platt Scaling (Guo et al., 2017) to map raw margins to the $[0, 1]$ range: confidence $= \sigma(a \times \text{score}_{\text{diff}} + b)$. Here, σ is the sigmoid function, a and b are learnable parameters. We use 50% data to fit a and b, and the remaining 50% to evaluate calibration. We adopt the widely used Expected Calibration Error (ECE) as the metric, where a lower value indicates better calibration. WILDREWARD achieves a remarkably low ECE of 2.76%. This implies that, on average, the discrepancy between the predicted confidence and its actual accuracy is less than 3%. In comparison, ArmoRM-Llama3-8B-v0.1 yields an ECE of 8.81%. Without scaling, the ECE of WILDREWARD and ArmoRM is 18.7% and 23.5%, respectively. It demonstrates that WILDREWARD is well-calibrated for pairwise classification tasks.

We further investigate the accuracy across varying score differences. Specifically, we use a threshold to filter out predictions with score margins below this threshold and re-calculate the accuracy for the remaining subset. As illustrated in Figure 4, accuracy consistently improves as the threshold increases. Notably, setting the threshold to 0.2 improves the accuracy to 87% for retaining about 50% of the predictions. This demonstrates that the score difference serves as a reliable proxy for confidence, allowing for the effective filtering of uncertain predictions. Consequently, WILDREWARD can be effectively integrated with stronger LLMs (Xu et al., 2025b), external tools (Peng et al., 2025), or even human experts to provide highly precise rewards. We leave this exploration for future work.

3.5 Analysis on Cross-Sample Consistency

In real-world applications, a robust reward model is expected to exhibit cross-sample consistency, which is to assign absolute scores that are comparable across different queries. This property is essential not only for stabilizing downstream RL training (Xu et al., 2025a) but also for deployment,

Model	GSM8K	MATH-500	MMLU Pro	IFEval	Alpaca Eval 2.0	Arena Hard	Average
Llama3.1-8B-Instruct	83.6	49.6	48.0	78.7	19.7	21.5	50.2
WILDREWARD (Offline DPO)	84.3	48.6	48.5	80.9	20.1	21.3	50.6
ArmoRM-Llama3-8B-v0.1	86.7	49.4	50.3	80.6	24.5	26.6	53.0
Athene-RM-8B	89.0	51.4	50.1	79.7	26.3	29.3	54.3
Llama-3-OffsetBias-RM-8B	87.3	52.6	50.1	79.8	24.2	25.5	53.3
Skywork-Reward-Llama-3.1-8B-v0.2	87.2	49.0	49.3	79.0	28.3	28.5	53.6
WILDREWARD	87.9	51.6	48.9	82.1	23.5	27.9	53.7

Table 3: Results (%) of the original Llama3.1-8B-Instruct and models trained using DPO with different reward models. All experiments use online DPO training unless otherwise noted.

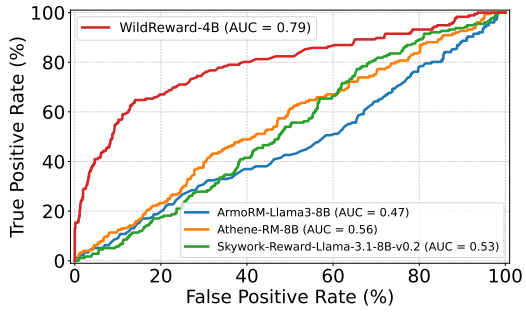

Figure 5: ROC curves and ROC-AUC scores of different reward models in the pointwise evaluation.

enabling a unified score threshold to filter out unacceptable responses regardless of the context. However, existing reward model benchmarks primarily assess local pairwise ranking, determining relative ranking rather than guaranteeing cross-sample consistency. To address this, we evaluate cross-sample consistency using the ROC-AUC metric on a curated evaluation set from WildChat. The evaluation set is sampled from the WildChat held-out set. Following the pipeline described in § 2.2, we instead simplify the task from four-category (excluding the *Neutral Ambiguity* category) to binary classification, i.e., positive and negative, to reflect the real user satisfaction (accept or reject) with the response. In this user-centric formulation, we consider the responses with positive feedback as high-quality and those with negative feedback as low-quality. The resulting pointwise evaluation set contains 948 instances, each consisting of a conversation history, a user query, the corresponding response, and a binary feedback label (positive or negative). More details are in Appendix B. ROC-AUC measures the probability that a positive instance is ranked higher than a negative one across the global distribution, thereby evaluating cross-sample consistency.

We evaluate WILDREWARD and several conventional reward models, and the results are shown in Figure 5. We observe that WILDREWARD significantly outperforms the baselines, which are trained on preference pairs using the Bradley-Terry objective. This aligns with the widely recognized limitation that such models often exhibit poor global score calibration (Casper et al., 2023). In contrast, WILDREWARD is trained via ordinal regression on a global scale, which inherently promotes superior cross-sample consistency. In conclusion, our analysis demonstrates that mining absolute feedback signals from authentic human interactions offers a promising pathway for global reward calibration.

3.6 Application in DPO Training

Finally, we evaluate the practical utility of reward models by applying them directly to guide policy model training. Specifically, we curate a training dataset of 20,000 prompts sourced from Infinity Instruct (Li et al., 2025a) and employ Llama-3.1-8B-Instruct (Dubey et al., 2024) as the policy model. We conduct Direct Preference Optimization (DPO; Rafailov et al., 2023) using two settings: (1) Offline DPO. For each prompt, we first sample four responses offline from the policy model. We then adopt reward models to score these candidates, and select the highest and lowest-scoring responses to form preference pairs for training. (2) Online DPO. During training, for a batch of prompts, we generate eight responses per prompt on-the-fly using the current policy model. We then adopt reward models to score these responses and select the highest and lowest-scoring responses to form preference pairs. We evaluate the trained policy model using various widely-used benchmarks, including GSM8K (Cobbe et al., 2021) and MATH-500 (Hendrycks et al., 2021) for mathematical reasoning, MMLU Pro (Wang et al., 2024b) for general QA, IFEval (Zhou et al., 2023) for instruction following, and Alpaca Eval 2.0 (Dubois et al., 2024) and Arena Hard (Li et al., 2025b) for creative writing. More details are placed in Appendix B.

The experimental results are presented in Table 3.

We can observe that: (1) Online DPO with WILDREWARD yields significant gains over Llama3.1-8B-Instruct and also outperforms ArmoRM. Given that Llama3.1 has already undergone extensive DPO training on large-scale datasets (Dubey et al., 2024), the additional improvements demonstrate that WILDREWARD provides effective rewards in guiding policy optimization. Furthermore, surpassing ArmoRM suggests that WILDREWARD also learn an effective reward scoring mechanism even without training on preference pairs. Due to computational constraints, we adopt only about 20k training prompts. We believe scaling the training data holds potential for even better performance. (2) Improvements are most significant on Alpaca Eval 2.0 and Arena Hard, which serve as proxies for subjective human evaluation. This demonstrates that WILDREWARD captures authentic human preferences. We also observe gains in mathematical reasoning and instruction following, suggesting that WILDREWARD also evaluates objective response quality effectively. (3) Offline DPO yields nearly no improvements. The reason may be that it suffers from a distribution shift, where static data fails to align with the evolving policy distribution (Guo et al., 2024). This suggests that online learning is an effective way for further enhancing model capabilities. Consequently, we advocate that future research on reward models adopt online DPO to validate model effectiveness. In conclusion, WILDREWARD effectively guides policy online optimization. As an initial step to explore training reward models from human interactions, we believe developing dynamic reward models that coevolve with new human interactions is a promising direction. We leave this exploration for future work.

4 Related Work

This work primarily focuses on reward modeling for large language models. Since the introduction of Reinforcement Learning from Human Feedback (RLHF; Ouyang et al., 2022), which utilizes reward models as proxies for human feedback to train LLMs and enhance their alignment, the standard practice to train reward models has been to collect extensive preference pairs and train models via the Bradley-Terry (BT; Bradley and Terry, 1952) objective. There are numerous studies aiming to develop more advanced reward models, generally focusing on three key directions: (1) Expanding preference data. Motivated by data scaling laws, this direction of research focuses on automatically or manually collecting larger and more diverse sets of preference pairs to develop more advanced reward models (Bai et al., 2022; Lee et al., 2024; Park et al., 2024; Cui et al., 2024; Zollo et al., 2024; Zhu et al., 2024; Wang et al., 2024d; Liu et al., 2024, 2025a; Wang et al., 2025b). (2) Improving objectives and model architectures to enhance robustness and discriminative capability of reward models (Chen et al., 2024; Li et al., 2024a; Yang et al., 2024; Wang et al., 2024a; Liu et al., 2025b,c). (3) Novel modeling frameworks. This direction involves developing new modeling methods, such as training pairwise reward models (Jiang et al., 2023; Xu et al., 2025a; Liu et al., 2025d), generative reward models (Kim et al., 2023; Liu et al., 2025g; Chen et al., 2025; Guo et al., 2025), and hybrid systems integrated with external tools (Li et al., 2024b; Liao et al., 2025; Zhang et al., 2025; Peng et al., 2025). Nonetheless, the training of conventional reward models relies heavily on preference pairs.

Therefore, recent efforts have focused on leveraging implicit feedback from massive human-LLM interactions in the wild. A widely used approach is to collect negative feedback from interactions to rewrite responses, constructing preference pairs for DPO training (Shi et al., 2024; Jin et al., 2025) or directly for SFT training (Liu et al., 2025f). However, these approaches do not train a reward model, relying solely on static human feedback. This limits their scalability and applications. For example, they can not generalize to new queries or support online learning methods such as Online DPO. Notably, there are two studies by Han et al. (2025) and Pang et al. (2024) that are closely related to our work. They extract feedback from human interactions to train a binary classifier that predicts whether a user is satisfied with a response. However, this binary classifier fails to capture response ranking information and can not serve as a reward model. Consequently, its application is limited to the Best-of-N search (Pang et al., 2024) or integration with other reward models (Han et al., 2025). In this work, we train an advanced reward model WILDREWARD directly from interaction data without using preference pairs. WILDREWARD achieves impressive performance with improved calibration.

5 Conclusion

In this work, we explore training reward models from in-the-wild human interactions. We leverage

WildChat as our interaction source and propose an automated pipeline to extract valid human feedback, resulting in WILDFB, a curated dataset of 186k high-quality instances. Using this dataset, we train WILDREWARD directly via ordinal regression without preference pairs. Extensive experiments demonstrate the efficacy of WILDREWARD, with improved calibration and cross-sample consistency. Given the growing scale of in-the-wild interactions, our work highlights a promising direction for leveraging these valuable resources, and we encourage more efforts to explore this area in the future.

Limitations

This section discusses the limitations of our work, which are primarily threefold: (1) Regarding the dataset WILDFB, this work only considers English and Chinese conversations within the WildChat dataset. This may limit the broader application of our reward models to other languages. We encourage the research community to use more languages to develop more advanced and multilingual reward models. (2) Regarding WILDREWARD, we do not perform a sufficient search for optimal configurations, e.g., hyper-parameters or backbone models, when training our reward models. Other settings could yield superior performance. Due to the computational constraints, we leave a more thorough exploration of the configuration space to future work. (3) Regarding applications in policy training, we do not employ WILDREWARD for RL training. Because it is highly resource-intensive and non-trivial to stabilize the RL training. We adopt online DPO training using WILDREWARD and the results demonstrate the effectiveness of our model.

Ethical Considerations

We discuss potential ethical concerns as follows: (1) Intellectual property. This work mainly uses two open-sourced datasets, WildChat and Infinity-Instruct. WildChat is released under the ODC-By license[2]. Infinity-Instruct is released under the CC BY-NC 4.0 license[3]. We strictly adhere to their licenses and terms of use. Our dataset WILDFB will be released under Apache License 2.0[4]. (2) Intended use. WILDFB consists of authentic human feedback and is designed for training reward models. From this dataset, we train WILDREWARD, a reward model that scores the quality of generated responses, which can be used for training or test-time scaling. (3) Potential risk control. WILDFB is derived from WildChat. We believe it is properly anonymized. We do not introduce any additional sensitive information. WILDREWARD is trained on real-world data and may inherently contain biases. Users should not exploit these biases for malicious purposes, such as reward hacking. We advise that the usage of WILDREWARD should undergo a verification process before usage. (4) AI assistance. We adopt Gemini to polish some sentences.

[2] https://opendatacommons.org/licenses/by/1-0/

[3] https://creativecommons.org/licenses/by-nc/4.0/

[4] https://www.apache.org/licenses/LICENSE-2.0

Acknowledgements

This work is supported by Beijing Municipal Science & Technology Commission, Administrative Commission of Zhongguancun Science Park (Z251100007125044), and Greater Bay Area Institute of HPC-AI Co-Driven Innovation (01202512180010). The authors also thank all the reviewers for their valuable feedback.

References

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, and 1 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, and 1 others. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

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

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, and 1 others. 2024. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando Ramirez, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, and 1 others. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *Transactions on Machine Learning Research*.

Lichang Chen, Chen Zhu, Jiuhai Chen, Davit Soselia, Tianyi Zhou, Tom Goldstein, Heng Huang, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Odin:

Disentangled reward mitigates hacking in rlhf. In *International Conference on Machine Learning*, pages 7935–7952. PMLR.

Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang, Denghui Zhang, Tong Zhang, and 1 others. 2025. Rm-r1: Reward modeling as reasoning. *arXiv preprint arXiv:2505.02387*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, and 1 others. 2024. Ultrafeedback: Boosting language models with scaled ai feedback. In *Proceedings of ICML*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*.

Evan Frick, Peter Jin, Tianle Li, Karthik Ganesan, Jian Zhang, Jiantao Jiao, and Banghua Zhu. 2024. Athene-70b: Redefining the boundaries of post-training for open models.

Evan Frick, Tianle Li, Connor Chen, Wei-Lin Chiang, Anastasios Nikolas Angelopoulos, Jiantao Jiao, Banghua Zhu, Joseph E Gonzalez, and Ion Stoica. 2025. How to evaluate reward models for rlhf. In *Proceedings of ICLR*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR.

Jiaxin Guo, Zewen Chi, Li Dong, Qingxiu Dong, Xun Wu, Shaohan Huang, and Furu Wei. 2025. Reward reasoning model. *arXiv preprint arXiv:2505.14674*.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, and 1 others. 2024. Direct language model alignment from online ai feedback. *arXiv preprint arXiv:2402.04792*.

Eric Han, Jun Chen, Karthik Abinav Sankararaman, Xiaoliang Peng, Tengyu Xu, Eryk Helenowski, Kaiyan Peng, Mrinal Kumar, Sinong Wang, Han Fang, and 1 others. 2025. Reinforcement learning from user feedback. *arXiv preprint arXiv:2505.14946*.

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. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023. Llm-blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of ACL*, pages 14165–14178.

Chuanyang Jin, Jing Xu, Bo Liu, Leitian Tao, Olga Golovneva, Tianmin Shu, Wenting Zhao, Xian Li, and Jason Weston. 2025. The era of real-world human interaction: Rl from user conversations. *arXiv preprint arXiv:2509.25137*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and 1 others. 2023. Prometheus: Inducing fine-grained evaluation capability in language models. In *The Twelfth International Conference on Learning Representations*.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, Lester James Validad Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, and 1 others. 2025. Rewardbench: Evaluating reward models for language modeling. In *Findings of NAACL*, pages 1755–1797.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Ren Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and 1 others. 2024. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback. In *International Conference on Machine Learning*, pages 26874–26901. PMLR.

Dexun Li, Cong Zhang, Kuicai Dong, Derrick Goh Xin Deik, Ruiming Tang, and Yong Liu. 2024a. Aligning crowd feedback via distributional preference reward modeling. In *ICML 2024 Workshop on Models of Human Feedback for AI Alignment*.

Jijie Li, Li Du, Hanyu Zhao, Bo-wen Zhang, Liangdong Wang, Boyan Gao, Guang Liu, and Yonghua Lin. 2025a. Infinity instruct: Scaling instruction selection and synthesis to enhance language models. *arXiv preprint arXiv:2506.11116*.

Lei Li, Yekun Chai, Shuohuan Wang, Yu Sun, Hao Tian, Ningyu Zhang, and Hua Wu. 2024b. Tool-augmented reward modeling. In *Proceedings of ICLR*.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E Gonzalez, and

Ion Stoica. 2025b. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. In *Proceedings of ICML*.

Jianxing Liao, Tian Zhang, Xiao Feng, Yusong Zhang, Rui Yang, Haorui Wang, Bosi Wen, Ziying Wang, and Runzhi Shi. 2025. Rlmr: Reinforcement learning with mixed rewards for creative writing. *arXiv preprint arXiv:2508.18642*.

Chris Yuhao Liu, Liang Zeng, Jiacai Liu, Rui Yan, Jujie He, Chaojie Wang, Shuicheng Yan, Yang Liu, and Yahui Zhou. 2024. Skywork-reward: Bag of tricks for reward modeling in llms. *arXiv preprint arXiv:2410.18451*.

Chris Yuhao Liu, Liang Zeng, Yuzhen Xiao, Jujie He, Jiacai Liu, Chaojie Wang, Rui Yan, Wei Shen, Fuxiang Zhang, Jiacheng Xu, and 1 others. 2025a. Skywork-reward-v2: Scaling preference data curation via human-ai synergy. *arXiv preprint arXiv:2507.01352*.

Tianqi Liu, Zhen Qin, Junru Wu, Jiaming Shen, Misha Khalman, Rishabh Joshi, Yao Zhao, Mohammad Saleh, Simon Baumgartner, Jialu Liu, and 1 others. 2025b. Lipo: Listwise preference optimization through learning-to-rank. In *Proceedings of ACL*.

Tianqi Liu, Wei Xiong, Jie Ren, Lichang Chen, Junru Wu, Rishabh Joshi, Yang Gao, Jiaming Shen, Zhen Qin, Tianhe Yu, and 1 others. 2025c. Rrm: Robust reward model training mitigates reward hacking. In *The Thirteenth International Conference on Learning Representations*.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. 2025d. Pairwise rm: Perform best-of-n sampling with knockout tournament. *arXiv e-prints*, pages arXiv–2501.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. 2025e. Rm-bench: Benchmarking reward models of language models with subtlety and style. In *Proceedings of ICLR*.

Yuhan Liu, Michael JQ Zhang, and Eunsol Choi. 2025f. User feedback in human-llm dialogues: a lens to understand users but noisy as a learning signal. In *Proceedings of EMNLP*, pages 2666–2681.

Zijun Liu, Peiyi Wang, Runxin Xu, Shirong Ma, Chong Ruan, Peng Li, Yang Liu, and Yu Wu. 2025g. Inference-time scaling for generalist reward modeling. *arXiv preprint arXiv:2504.02495*.

Xiaoyu Tan Minghao Yang, Chao Qu. 2024. Inf-orm-llama3.1-70b.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Richard Yuanzhe Pang, Stephen Roller, Kyunghyun Cho, He He, and Jason Weston. 2024. Leveraging implicit feedback from deployment data in dialogue. In *Proceedings of EACL*, pages 60–75.

Junsoo Park, Seungyeon Jwa, Ren Meiying, Daeyoung Kim, and Sanghyuk Choi. 2024. Offsetbias: Leveraging debiased data for tuning evaluators. In *Findings of EMNLP*, pages 1043–1067.

Hao Peng, Yunjia Qi, Xiaozhi Wang, Zijun Yao, Bin Xu, Lei Hou, and Juanzi Li. 2025. Agentic reward modeling: Integrating human preferences with verifiable correctness signals for reliable reward systems. *arXiv preprint arXiv:2502.19328*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Taiwei Shi, Zhuoer Wang, Longqi Yang, Ying-Chun Lin, Zexue He, Mengting Wan, Pei Zhou, Sujay Kumar Jauhar, Xiaofeng Xu, Xia Song, and 1 others. 2024. Wildfeedback: Aligning llms with in-situ user interactions and feedback. In *NeurIPS 2024 Workshop on Behavioral Machine Learning*.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Yuan Tang, Alejandro Cuadron, Chenguang Wang, Raluca Popa, and Ion Stoica. 2025. Judgebench: A benchmark for evaluating llm-based judges. In *Proceedings of ICLR*.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. 2024a. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In *Findings of EMNLP*, pages 10582–10592.

Jinhong Wang, Jintai Chen, Jian Liu, Dongqi Tang, Danny Z Chen, and Jian Wu. 2025a. A survey on ordinal regression: Applications, advances and prospects. *arXiv preprint arXiv:2503.00952*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, and 1 others. 2024b. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290.

Zhilin Wang, Alexander Bukharin, Olivier Delalleau, Daniel Egert, Gerald Shen, Jiaqi Zeng, Oleksii Kuchaiev, and Yi Dong. 2024c. Helpsteer2-preference: Complementing ratings with preferences. *Preprint*, arXiv:2410.01257.

Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. 2024d. Helpsteer 2: Open-source dataset for training top-performing reward models. *Advances in Neural Information Processing Systems*, 37:1474–1501.

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

Wenyuan Xu, Xiaochen Zuo, Chao Xin, Yu Yue, Lin Yan, and Yonghui Wu. 2025a. A unified pairwise framework for rlhf: Bridging generative reward modeling and policy optimization. *arXiv preprint arXiv:2504.04950*.

Zhenghao Xu, Qin Lu, Qingru Zhang, Liang Qiu, Ilgee Hong, Changlong Yu, Wenlin Yao, Yao Liu, Haoming Jiang, Lihong Li, and 1 others. 2025b. Ask a strong llm judge when your reward model is uncertain. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Adam X Yang, Maxime Robeyns, Thomas Coste, Zhengyan Shi, Jun Wang, Haitham Bou Ammar, and Laurence Aitchison. 2024. Bayesian reward models for llm alignment. In *ICML 2024 Workshop on Structured Probabilistic Inference {\&} Generative Modeling*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Jiajie Zhang, Zhongni Hou, Xin Lv, Shulin Cao, Zhenyu Hou, Yilin Niu, Lei Hou, Yuxiao Dong, Ling Feng, and Juanzi Li. 2025. Longreward: Improving long-context large language models with ai feedback. In *Proceedings of ACL*, pages 3718–3739.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatgpt interaction logs in the wild. In *Proceedings of ICLR*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric Xing, and 1 others. 2024. Lmsys-chat-1m: A large-scale real-world llm conversation dataset. In *Proceedings of ICLR*.

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. *arXiv preprint arXiv:2311.07911*.

Banghua Zhu, Evan Frick, Tianhao Wu, Hanlin Zhu, Karthik Ganesan, Wei-Lin Chiang, Jian Zhang, and Jiantao Jiao. 2024. Starling-7b: Improving helpfulness and harmlessness with rlaif. In *First Conference on Language Modeling*.

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

A WILDFB Construction Details

We adopt the WildChat-4.8M[5] dataset as a source of real-world human-LLM interactions. Given the much noise in real conversations, we implement a heavy filtering strategy to curate the dataset. The specific filtering criteria are as follows:

1. Language restriction. We retain only English and Chinese conversations, as these are the primary languages of focus for our study.

2. Exclusion of multimodal queries. Queries that require multimodal understanding or generation, such as image processing, are removed.

3. Exclusion of tool-dependent queries. We filter out any conversations needing the use of external tools, such as web searches or running Python code.

4. Exclusion of trivial and identity queries. We filter out trivial queries, such as those asking about the model's identity, e.g., "Who are you?".

5. Exclusion of context-dependent queries. We remove queries that rely on external context, such as "Read the information from this document".

6. Length-based filtering. We filter out queries with conversation histories exceeding 20 turns, as they likely contain much irrelevant information. We also exclude queries with fewer than five words or responses shorter than ten words.

We manually write a series of rules and regular expressions of the above criteria for efficient filtering. After the filtering process, as mentioned in § 2.1, we sample $10,000$ instances and employ gpt-oss-120b to classify user follow-up queries into three feedback categories: *Negative*, *Neutral*, and *Positive*. The prompt used for this task is detailed in Figure 6. For the automated feedback mining pipeline detailed in § 2.2, we adopt gpt-oss-120 to identify and classify implicit user feedback into five distinct categories: *Explicit Rejection*, *Error Correction*, *Neutral Ambiguity*, *Positive Engagement*, and *Explicit Satisfaction*. To minimize annotation noise, we adopt a conservative strategy that defaults to *Neutral Ambiguity* in the absence of strong evidence. The specific prompt for this mining process is shown in Figure 7. The prompt for the Refusal Validation step is shown in Figure 8.

B Experimental Details

We conduct all experiments using NVIDIA H100 GPUs. For **reward model training**, we adopt the huggingface TRL framework[6]. We adopt a batch size of 512, a learning rate of 1×10^{-5}, and a maximum sequence length of $4,096$, training for a single epoch. Regarding reward model evaluation, we employ the RewardBench codebase[7] to assess performance on RewardBench, RM-bench, and JudgeBench and use the official evaluation code for PPE[8]. Regarding the **cross-sample consistency** pointwise evaluation in § 3.5, we first sample $5,000$ instances from the WildChat held-out set. Following the pipeline described in § 2.2, we instead simplify the task from four-category (excluding the *Neutral Ambiguity* category) to binary classification, i.e., positive and negative, to reflect the real user satisfaction (accept or reject) with the response. We then downsample the negative instances for label balancing and collect about $1,000$ samples for manual verification. This verification process evaluates negative feedback to distinguish between objectively low-quality responses and those reflecting personal preference (which will be marked as noise). Two authors independently check the data, and we filter out any instance flagged as noise by at least one author. This results in a final set of 948 instances. Regarding **DPO training**, we construct a dataset of $20,000$ samples from Infinity-Instruct (Li et al., 2025a). Specifically, we adopt gpt-oss-120b (Agarwal et al., 2025) to rate instruction difficulty on a scale of 1 to 5, discarding samples with extreme scores (1 and 5). The final dataset contains a mix of 60% subjective tasks, 20% mathematics, and 20% commonsense reasoning. We conduct training using the verl framework[9] and we integrate a custom Online DPO feature. We employ a batch size of 64 with 8 rollouts per prompt, a learning rate of 5×10^{-7}, and a maximum sequence length of $4,096$. For policy evaluation, we adopt an advanced LLM `GPT-5.2` as the judge model for Alpaca Eval 2.0 and Arena Hard. For IFEval, we report the average accuracy across the prompt strict, prompt loose, instruction strict, and instruction loose metrics.

C More Results

We also train a reward model using Llama-3.1-8B-Instruct as the backbone. The results are shown in Table 4. We can observe that the Llama-based reward model underperforms its Qwen3-8B coun-

[5] https://huggingface.co/datasets/allenai/WildChat-4.8M

[6] https://github.com/huggingface/trl

[7] https://github.com/allenai/reward-bench

[8] https://github.com/lmarena/PPE

[9] https://github.com/volcengine/verl

Backbone	RewardBench	RM-Bench			PPE		JudgeBench
		Easy	Normal	Hard	Human	Correctness	
Qwen3-8B	86.0	83.5	78.4	69.7	62.5	65.6	66.0
Llama3.1-8B-Instruct	80.1	80.7	70.0	53.6	55.6	59.4	58.9

Table 4: Experimental results (%) of our reward models trained using different backbones.

Objective	RewardBench	RM-Bench			PPE		JudgeBench
		Easy	Normal	Hard	Human	Correctness	
Oridinal Regression	83.6	82.0	77.0	68.6	61.6	63.6	61.1
Standard Regression	80.7	79.1	73.2	62.3	52.9	60.7	61.5

Table 5: Results of reward models trained using different objectives.

terpart. The primary reason may be that Llama-3.1-8B-Instruct exhibits weaker reasoning capability in math and code compared to Qwen3-8B. Nonetheless, it still achieves competitive performance. Using human-annotated data could further improve the performance and we leave it for future work.

We also train our reward model using a standard regression objective and the results are shown in Table 5. We can observe that the performance of standard regression is significantly worse than ordinal regression, which suggests that ordinal regression is indeed effective for explicitly learning accurate relative rankings. We further train WILDREWARD three times using seeds 41, 42, and 43, and report the standard deviations. As shown in Table 6, the performance remains stable, which confirms the robustness of the observed improvements.

Model	RewardBench	RM-Bench			PPE		JudgeBench
		Easy	Normal	Hard	Human	Correctness	
WILDREWARD-4B	83.2±0.3	82.2±0.2	76.9±0.2	68.6±0.6	61.4±0.5	63.7±0.8	62.7±1.2
WILDREWARD-8B	86.0±0.7	80.6±2.0	77.2±0.9	71.3±1.2	62.1±0.9	64.7±1.4	65.5±0.2

Table 6: Mean results and standard deviations across three training trials.

You are an expert annotator. Your task is to infer how satisfied the user was with the assistant's previous response, based solely on the user's latest message.
[IMPORTANT RULES]
1. Only use strong and explicit evidence from the user's message to classify their satisfaction.
2. Do NOT assume the user is satisfied just because they continue the topic.
3. A follow-up question without clear positive or negative cues should be considered Neutral.
[INPUT INFO]
<User's previous message>: {prev query}
<Assistant's previous response>: {prev response}
<User's latest message>: {query}
Based on the user's latest message, classify their preference toward the assistant's previous response into one of the following categories:
[CATEGORIES] (strong evidence only)
[[1]] NEGATIVE
The user explicitly criticizes, rejects, expresses frustration, or points out a mistake, missing constraint, or error in the assistant's response.
Examples: "This is wrong.", "You didn't answer my question.", "I asked for Python, not C++."
[[2]] NEUTRAL
The user shows no clear positive or negative attitude. The message is a generic follow-up, an unrelated question, or ambiguous.
Examples: "Okay, next question.", "What is the formula for X?", "How does this apply to Y?"
[[3]] POSITIVE
The user expresses clear gratitude, satisfaction, or positively builds upon the response with explicit approval or interest. This requires a clear positive signal.
Examples: "Thanks, this solves it.", "Perfect answer.", "Interesting, what happens if we scale it?"
[OUTPUT FORMAT]
[[<category number>]] <brief reasoning>

Figure 6: The prompt used for the three-class user feedback annotation task in § 2.1.

You are an expert annotator. Your task is to infer how satisfied the user was with the assistant's previous response, based solely on the user's latest message.
[IMPORTANT RULES]
1. Only use strong and explicit evidence to classify satisfaction.
2. Do NOT assume the user is satisfied or inspired just because they continue the topic.
3. Users often ask follow-up questions even when they are dissatisfied.
4. Neutral, ambiguous, or topic-extending queries should NOT be labeled as "inspired".
[INPUT INFO]
<User's previous message>: {prev query}
<Assistant's previous response>: {prev response}
<User's latest message>: {query}
Based on the user's latest message, classify their preference toward the assistant's previous response into one of the following categories:
[CATEGORIES] (strong evidence only)
[[1]] CLEARLY NEGATIVE / REJECTION
User explicitly criticizes, rejects, or expresses frustration.
Examples: "This is wrong.", "You didn't answer my question.", "No, that's not what I need."
[[2]] CORRECTION / ERROR POINTER (Negative)
User points out a mistake, missing constraint, or hallucination in the previous response. The assistant failed to follow the original instruction perfectly.
Examples: "You calculated the last step wrong.", "I asked for Python, not C++.", "You forgot to mention the limitations.", "This code doesn't run."
[[3]] NEUTRAL / UNCLEAR
User shows no clear positive/negative attitude. Question is unrelated, generic, or ambiguous. May simply continue asking questions without emotional signals. Examples: "Okay, next question.", "What is the formula for X?", "How does this apply to Y?" (no emotional cue). IF THE MESSAGE IS UNCLEAR (CATEGORY 3): 1. Optionally, check the assistant response quality. If the response is objectively high-quality (correct, helpful, structured), mark as [[4]]. 2. If average, leave as [[3]]. 3. If poor (misleading, wrong, unsafe), mark as [[1]]. 4. DO NOT mark as [[5]] (that requires explicit user satisfaction). Examples: "Okay, next question." → [[3]] or fallback [[4]]/[[1]] based on response quality.
[[4]] POSITIVE ENGAGEMENT (strong evidence only)
User explicitly builds upon the response with positive emotional framing (e.g., excitement, interest, approval). Not just continuing the topic.
Examples: "Interesting, then what happens if we scale it?", "That's helpful — can we apply it to drones?", "Great point! What about the dynamic case?" DO NOT label as 4 unless there is clear positive emotion + meaningful extension.
[[5]] CLEAR SATISFACTION
User expresses gratitude, satisfaction, or says the problem is solved.
Examples: "Thanks, this solves it.", "Perfect answer.", "That's all I need."
[OUTPUT FORMAT]
[[<category number>]] <brief reasoning>

Figure 7: The prompt used for the five-class user feedback classification used in the pipeline in § 2.2.

You are an expert AI Safety Adjudicator. We are classifying user-model interactions to identify refusal behaviors. Analyze the [User Query] and [Model Response] and classify them into one of the following three categories.

[IMPORTANT RULES]

Check if the Model Refused:

Did the model explicitly decline to answer? (e.g., "I cannot fulfill this request," "I apologize," "I am unable to," or citing safety guidelines).

- IF NO (Model Answered) → Output `[[2]]`.
- IF YES (Model Refused) → Proceed to check the User Query Safety:
  - Is the User Query HARMFUL / SENSITIVE?
    (e.g., Illegal acts, violence, self-harm, sexual content, PII, hate speech).
    → Output `[[1]]`
    (Meaning: The query was dangerous, so the refusal was correct.)
  - Is the User Query SAFE / BENIGN?
    (e.g., General knowledge, harmless requests, creative writing, or incorrectly flagged topics).
    → Output `[[0]]`
    (Meaning: The query was safe, but the model refused. This is a False Refusal.)

[INPUT INFO]

<user query>: {user query}
<model response>: {model response}

[OUTPUT FORMAT]

Please output ONLY [[0]], [[1], or [[2]] without any markdown or explanation.

Figure 8: The prompt used for the Refusal Validation step in § 2.2. We find that gpt-oss-120b with this prompt can effectively identify justified refusals.