Towards Data-Centric RLHF: Simple Metrics for Preference Dataset Comparison

Judy Hanwen Shen*
Stanford University
jhshen@stanford.edu

Archit Sharma
Stanford University
architsh@stanford.edu

Jun Qin
Apple
jqin22@apple.com

Abstract

The goal of aligning language models to human preferences requires data that reveal these preferences. Ideally, time and money can be spent carefully collecting and tailoring bespoke preference data to each downstream application. However, in practice, a select few publicly available preference datasets are often used to train reward models for reinforcement learning from human feedback (RLHF). While new preference datasets are being introduced with increasing frequency, there are currently no existing efforts to measure and compare these datasets. In this paper, we systematically study preference datasets through three perspectives: scale, label noise, and information content. We propose specific metrics for each of these perspectives and uncover different axes of comparison for a better understanding of preference datasets. Our work is a first step towards a data-centric approach to alignment by providing perspectives that aid in training efficiency and iterative data collection for RLHF.

1 Introduction

Reinforcement learning from human feedback (RLHF) is typically the final stage of the modern large language model (LLM) training pipeline Achiam et al. [2023], Touvron et al. [2023], Groeneveld et al. [2024]. The reward models necessary for RLHF algorithms are predominantly trained from datasets of pairwise preferences Bai et al. [2022], Ouyang et al. [2022]. While a substantial number of works have focused on new algorithms for learning from preference data to better train reward models Moskovitz et al. [2023], Zheng et al. [2023], Dong et al. [2024], Xiong et al. [2024], relatively few works have examined qualities of these datasets themselves. At the very minimum, all of these pairwise datasets of human preferences contain examples with 1) a prompt, 2) two responses, and 3) an annotation of which response is preferred. Beyond this basic structure, preference datasets vary widely in domain (e.g. code, chat, QA, etc.), generation process (e.g. synthetic vs human), collection procedure (e.g. annotation, prompt generation), and even size (e.g. 10k - 300k examples Zheng et al. [2023], Cui et al. [2023]).

Ideally, a custom preference dataset for each specific application can be collected, and carefully labeled by multiple annotators for reward model training. New technical reports that accompany state-of-the-art language models highlight the importance of preference data quality yet give little to no details about the preference datasets used DeepSeek-AI et al. [2024], Dubey et al. [2024]. Among

*Work done during internship at Apple

2nd Workshop on Attributing Model Behavior at Scale – 38th Conference on Neural Information Processing Systems (NeurIPS 2024).

publicly available preference datasets, there is folk wisdom that more carefully curated datasets are better, yet no rigorous study or methodology for comparing these datasets exists beyond summary statistics, such as token count Dong et al. [2024]. Today, little is known about when and why one preference dataset may be better than another, nor what "better" can mean in the context of these datasets.

In this paper, we initiate the study of measuring properties of preference datasets for the purpose of reward model training. A useful measurement should be robust to different base model choices and applicable to any dataset containing pairwise preferences. To this end, we propose three data-centric approaches for comparing preference datasets: effective sample size, noise invariance, and information content. We evaluate both in-distribution performance and domain generalization (i.e. through a standard reward modeling benchmark) on the induced reward model trained on these datasets. We validate our results through ablations across different model sizes to demonstrate the connection between these measurements and subsequent reward model performance. Together, our work gives three simple but intuitive perspectives for understanding preference datasets that are broadly applicable to the development of new datasets across domains.

2 Related Work

Data-Centric Methods Scaling laws introduced to describe the relationship between parameters, data, and compute for pre-training have been widely accepted as the explanation for why larger models and more data are better for language model training Kaplan et al. [2020], Hoffmann et al. [2022]. Different approaches for improving data quality and composition have been proposed as efficient alternatives for indiscriminately training on all available data Penedo et al. [2024], Xie et al. [2024]. However, the scale of pre-training data vastly eclipses the scale of data used in the fine-tuning and RLHF stages. Data quality and data selection for reward model training may be more similar to supervised learning settings than language modeling. In supervised learning and supervised fine-tuning, careful data selection and pruning have been shown to lower the number of samples required Paul et al. [2021], Sorscher et al. [2022], Xia et al. [2024]. However, reward models do differ from the supervised learning setting since they are adapted from these pre-trained base models. Recent work has studied data scaling for fine-tuning LLMs to find that LLM performance benefits more from pre-training data scaling than fine-tuning data scaling and the optimal fine-tuning method is task and data-dependent Zhang et al. [2024a].

Publicly Available Preference Datasets For RLHF preference datasets in particular, early works collected datasets on the order of tens of thousands of examples for reward model training. For example, for a summarization task Stienon et al., Stiennon et al. [2020] collected 64k preference pairs based on Reddit prompts, while the WebGPT Nakano et al. [2021] reward model was trained with 16k preference pairs based on prompts from existing QA datasets. Subsequent datasets follow a more general human-assistant format while being much larger (e.g. OpenAssistant Köpf et al. [2024], HH-RLHF Bai et al. [2022], Stanford Human Preferences Ethayarajh et al. [2022]). However, these datasets vary drastically in collection procedure. For example, for InstructGPT and HH-RLHF humans were asked to rank model-generated responses while for OpenAssistant and Stanford Human Preferences preferences for different human-generated responses were gathered. More recently, preference datasets where both responses and rankings are synthetically generated have gained popularity Cui et al. [2023], Daniele and Suphavadeeprasit [2023]. These synthetically constructed datasets offers more training samples and more diversity in terms of the topics generated. There is also a movement back to creating smaller but carefully annotated preferences, often with multiple annotators Wang et al. [2024]. Despite the large variation in practices for generating these different datasets, there has been little comparison and characterization of how different datasets affect reward model training.

Challenges of Reward Modeling and Learning from Human Preferences Defining data quality is complex for preference data since many different tasks may use the same reward model for RLHF. There are concerns with the representativeness of preferences as well as the alignment between collected data and the intended objective Lambert and Calandra [2023], Kirk et al. [2024], Chen et al. [2024]. One suggestion for measuring the effectiveness of reward models is standardized benchmarks on reward model performance on a variety of common tasks Lambert et al. [2024]. This approach measures the generalization of a single reward model on different tasks by testing how well each

reward model performs on scoring the chosen response higher. The top-performing models on this benchmark leaderboard include models of a variety of sizes from 8B to 340B parameters and a variety of preference data sizes from 10k to more than 700k examples. Given this mishmash of different approaches, it is important to understand how to measure preference data quality for the reward modeling step of RLHF. This work aims to characterize the elements of preference data quality that inform practical decisions around data generation, annotation, and usage in this setting.

3 Model Agnostic Data Metrics

3.1 Preliminaries

Let x be the prompt, y_w be the winning (chosen) response, and y_l be the losing (rejected) response. Let $D = \{(x, y_w, y_l)_i\}_{i=1}^n \sim \mathcal{D}$ be the dataset of preferences that we will study. Let $r : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ be the reward model that maps a (x, y) prompt response pair to a score. In reward modeling, we want to compare the rewards of two given generations. The Bradley-Terry model defines Y_{ij} as a Bernoulli random variable representing the outcome of whether the completion y_i is preferred or wins over the completion y_j. Under this model, $Y_{ij} \sim$ Bernoulli(p_{ij}) and the log ratio of the probability that y_i wins over y_j is:

$$\log \frac{p_{ij}}{1 - p_{ij}} = r(x, y_i) - r(x, y_j).$$

If we let y_i be the winning completion y_w and y_j be the losing completion y_l, we can then write the probability of the reward model preferring y_w as:

$$P(y_w \succ y_l) = \frac{\exp(r(x, y_w))}{\exp(r(x, y_l)) + \exp(r(x, y_w))}.$$

Following prior work Ouyang et al. [2022], Bai et al. [2022], the probability of the reward model giving a higher score to the chosen response can then be maximized directly through the following objective function:

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

3.2 Datasets and Models

We examine four publicly available preference datasets in our study: Anthropic Helpful-Harmless (HH-RLHF) Bai et al. [2022], Ultrafeedback (ULTRAFEEDBACK) Cui et al. [2023], LMSYS Arena Preferences (LMSYS) Chiang et al. [2024], and PKU-SafeRLHF (SAFERLHF) Ji et al. [2024]. These datasets are selected based on their frequent use in prior works Bai et al. [2022], Dong et al. [2024][2]. For each dataset, we examine their behavior on reward models trained from pre-trained models of different sizes: 350 million (Opt-350m Zhang et al. [2022]), 1 billion (TinyLlama-1B-3T Zhang et al. [2024b]), and 7 billion parameters (Llama2-7B and Llama2-7B-Chat Touvron et al. [2023]). We focus predominately on reward models trained from base models but also include ablations with fine-tuned versions since the practice around reward model training varies. For example, some papers train reward models from checkpoints already fine-tuned with instructions and human feedback (e.g. Llama3-8B-Instruct) Dong et al. [2024]) and other works train reward models directly from based models Wang et al. [2024], Zheng et al. [2023]. Notably, Ouyang et. al. Ouyang et al. [2022] remark that similar reward model quality was observed between training on a base 6B model and an instruction-tuned 6B model. We evaluate both in-domain and generalization performance through evaluation set accuracy and Rewardbench Lambert et al. [2024] respectively.

4 Experiments

4.1 Scaling: Are larger preference datasets better?

The first perspective we examine is the role of dataset size for different preference datasets. Unlike scaling laws for pre-training, there is no consensus about how large a preference dataset should be to train a good reward model. For summarization in particular, Stienon et. al. Stiennon et al. [2020] estimate that doubling their particular dataset size leads to a 1.1% increase in reward model

[2]We include dataset details in the supplementary materials

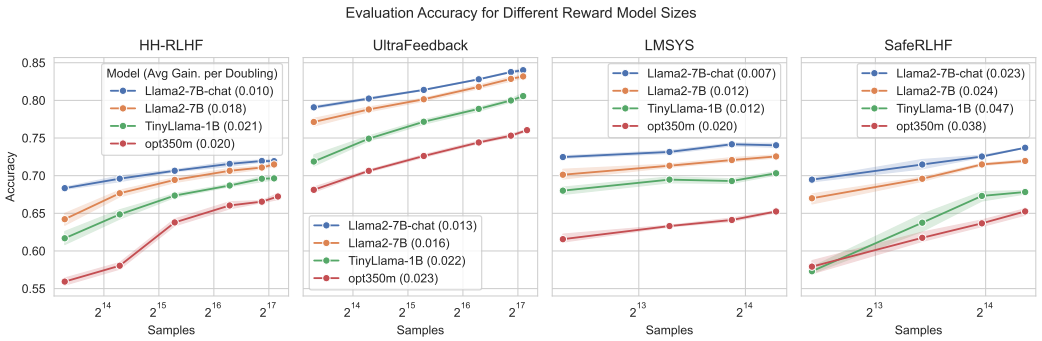

Figure 1: Scaling behavior when measuring evaluation set accuracy is dataset dependent.

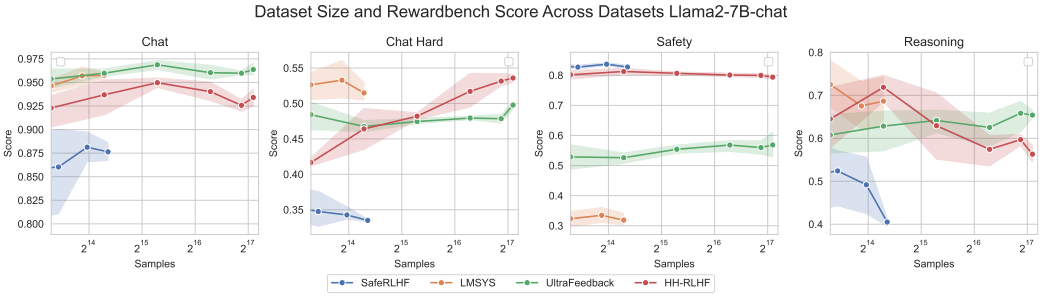

Figure 2: Comparing RewardBench performance across different datasets for Llama2-7B-chat model. Increasing the dataset size does not improve performance for most datasets on most tasks.

validation accuracy until 65k examples. In contrast, others have found that even when using 2.9 million examples, reward model accuracy continues to improve Touvron et al. [2023]. While these differences can be blamed on the dataset composition, the impact of increasing the training set size across different datasets has not been studied. We examine four datasets that range in size from 30k examples to 200k examples and observe how training dataset size impacts performance. Figure 1 illustrates the impact of scaling on evaluation set accuracy. For all datasets, the larger models (Llama2-7B, Llama2-7B-chat), gain less from doubling the dataset size. While Llama2-7B-chat is fine-tuned with RLHF from part of HH-RLHF, this pattern remains even for other datasets that were released after Llama2-7B-chat. Among datasets, SAFERLHF has the highest average gain per doubling of the training dataset (2.4-4.7%) for all models.

We also investigate the effect of increasing dataset size on a more general suite of reward model tasks that might be outside the training distribution using RewardBench Lambert et al. [2024] (Figure 2). Unlike evaluation accuracy, increasing dataset size does not always improve, and sometimes harms, performance on this benchmark. Some datasets dominate a task across all sample sizes (e.g. ULTRAFEEDBACK on Chat, HH-RLHF on Reasoning, and SAFERLHF on Safety). This shows that a small subset of samples (e.g. 10K examples or 10% of a dataset) is already sufficient and that dataset composition may be more important in achieving good performance than scale. For example, 10k examples from SAFERLHF outperforms 140k examples from HH-RLHF. These results are model invariant across different reward model sizes. For example, ULTRAFEEDBACK remains the best dataset for the Chat category across both the 350M and 1B model[3].

4.2 Noise Invariance: How robust are reward models to label noise?

Prior works have reported the human agreement with the collected preferences to be 76% for summarization Stiennon et al. [2020] and 73% inter-annotator agreement for response quality for general instruction tasks Ouyang et al. [2022]. Ideally, annotator disagreement serves as a filter for low-quality preference data, however, even if the collection process is unknown, it is still useful to

[3]See Appendix D.1 for details

Base Model	HH-RLHF	ULTRAFEEDBACK	LMSYS	SAFERLHF
Opt-350m	88.6%	95.0%	94.9%	92.6%
TinyLlama-1B	90.1%	95.4%	95.4%	94.4%
Llama2-7B	78.9%	93.3%	94.6%	84.9%
Llama2-7B-chat	92.7%	93.6%	92.4%	90.7%

Table 1: Percentage of total evaluation accuracy achieved with 30% of labels flipped for each dataset for different sized reward models. ULTRAFEEDBACK and LMSYS are particularly noise invariant but all datasets are fairly robust to label flipping.

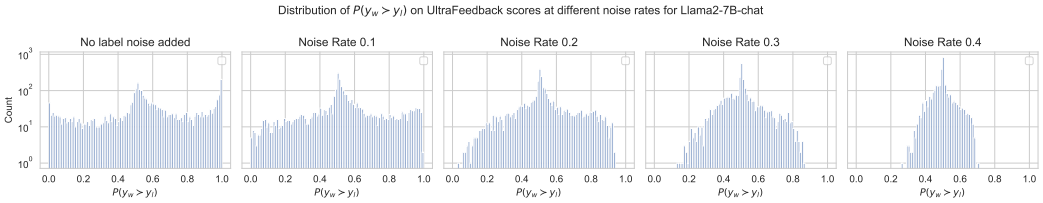

Figure 3: The impact of noise on reward model confidence $P(y_w \succ y_l)$ on ULTRAFEEDBACK for RewardBench. We see that as the noise rate (% of flipped labels) increases, the probability of the winning response being chosen concentrates around 0.5. This phenomenon is similar across all models and datasets to different extents.

understand how much noise there might be in the preference dataset. In image classification tasks, neural networks are robust to label noise Rolnick et al. [2017]. In these settings, a random label is used instead of the true label in multiclass classification. In the context of preference data, we can model label noise as the flipping of the chosen response with the rejected response. We can define: p as the noise rate and add random label noise by constructing a dataset:

$$(x, y_w, y_l) = \begin{cases} (x, y_l, y_w) & \text{w.p.} \quad p \\ (x, y_w, y_l) & \text{w.p.} \quad 1-p. \end{cases}$$

Table 1 shows the percentage of the peak evaluation set accuracy achieved when 30% of labels are flipped. Overall, we find that reward model performance remains unaffected by label flipping until 30-40% of labels are flipped. The same pattern is observed on RewardBench tasks across all models[4].

Explaining Noise Invariance: The Role of Noise in Reward Model Confidence We can look at the underlying prediction probabilities to further understand why introducing label noise does not significantly affect performance both on the evaluation set and the RewardBench tasks. Since accuracy for both sets of metrics is calculated through expected binary outcomes (i.e. $\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\mathbb{1}[\hat{y} = y_c]\right]$ where $\hat{y} = \arg\max_{y\in\{y_w,y_l\}} r(x, y)$), we can use the Bradley-Terry model to calculate $P(y_w \succ y_l)$ and investigate how these distributions change. As the noise rate increases, the distribution of probabilities (e.g. $P(y_w \succ y_l)$) becomes more concentrated around 0.5 (Figure 3). This pattern is consistent across different reward model sizes and datasets. Across different datasets, Figure 5 shows that when label noise is introduced, HH-RLHF and LMSYS collapses quicker to $P(y_w \succ y_l) \approx 0.5$ than other datasets. This suggests that there might be a higher level of baseline noise in the HH-RLHF labels that results in more uncertain predictions. This pattern is again consistent across different reward model sizes.

To precisely characterize model confidence, we can measure the expected calibration error (ECE) of reward model outputs Guo et al. [2017]. However, in the Bradly-Terry model, using $P(y_w \succ y_l)$ directly as model confidence results in perfect accuracy when $P(y_w \succ y_l) > 0.5$. The only prior work we could find that measures calibration in reward models uses $\max\{P(y_w \succ y_l), P(y_l \succ y_w)\}$ as the confidence of the model Pikus et al. [2023]. To properly measure calibration, we can write each evaluation pair as (x, y_1, y_2, z) and split it into $(x, y_w, y_l, z = 1)$ and $(x, y_l, y_w, z = 0)$. Then to calculate the calibration error we can use $P(z = 1) := P(y_1 \succ y_2)$ as model confidence and plot the count of $z = 1$ as the accuracy (see Figure 6). The overall ECE is equivalent to the max method from prior work but now we have confidence values in the entire interval of $[0, 1]$ instead of just

[4]Full plots and more details can be found in Appendix D.2

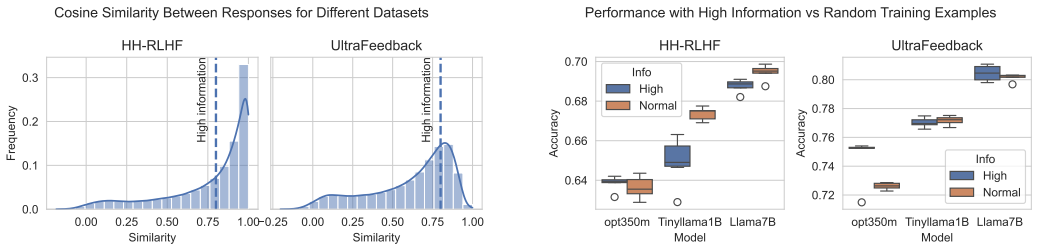

Figure 4: (Left) Distribution of cosine similarity of response pairs for different datasets. The HH-RLHF dataset contains much more similar response pairs (e.g. (y_w, y_l)) than the ULTRAFEEDBACK dataset. (Right) The evaluation set accuracy for training different models with "high information" or low response similarity data compared to a random sample. The benefits of "high information" are most salient in the smallest model.

$[0.5, 1]$. As label noise increases, we observe lower calibration error (e.g. ECE=0.183 no noise to ECE=0.086 30% label noise for ULTRAFEEDBACK) (see Section A for more details).

4.3 Information Content: Are high contrast responses necessary for reward model learning?

A major dichotomy in how preference datasets are generated is whether the responses are human-written or sampled from large language models. For example, the Anthropic Helpful-Harmless (HH-RLHF) dataset contains response pairs generated from responses from LLMs of the same family Bai et al. [2022]. In contrast, the Stanford Human Preference Dataset (SHP) dataset is gathered from pairs of (presumably human) Reddit responses Ethayarajh et al. [2022]. As responses are more similar in quality, prior work has found that human annotation agreement reduces these responses Touvron et al. [2023]. While the relative informativeness of an example for training a reward model is likely model-dependent, since the models used for reward model training vary in training data, a minimal level of contrast between the chosen and rejected response is likely a prerequisite for valuable examples in preference datasets. Given the differences in response generation, we can compare and contrast different datasets by computing the cosine similarity between embeddings of responses (i.e. $1 - d_{cos}(y_w, y_l)$)[5]. Figure 4 shows that the HH-RLHF dataset has many more similar response pairs than ULTRAFEEDBACK. To understand the impact of training with high-information examples, we created a threshold of 0.8 in cosine similarity and designated the examples with a smaller similar as "high information". Fixing the training set size, we compared the performance of training the high-information examples to a random sample. Surprisingly, the results vary by model and dataset. For the larger models (i.e. 1B+ parameters), there is little difference between the high information and random training sets of the same size. However, for the smaller 350 million parameter model, we see that the high information examples often resulted in a better evaluation accuracy (Figure 4).

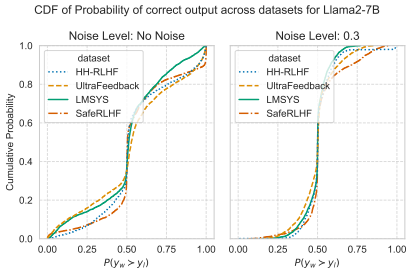

Figure 5: Empirical CDF of $P(y_w \succ y_l)$ for different datasets at different noise levels for Llama 7B on RewardBench. When there is no noise, some datasets induce a more confident distribution even with the same number of training examples. As more noise is added, all probabilities shift towards 0.5 and the datasets become indistinguishable

5 Discussion

Our work investigates three aspects of preference datasets to identify dataset differences and connect these differences to downstream performance on both in- and out-domain tasks. Firstly, we find

[5]We use all-MiniLM-L6-v2 from Sentence Transformer to generate embeddings. We investigated a suite of different sentence embeddings and found them to be highly correlated.

that while preference datasets vary in size, a larger dataset is not better than a smaller dataset that is more relevant to the task. Furthermore, increasing dataset size gives only marginal gains for in-domain evaluation accuracy and may even hinder performance on out-of-domain tasks. Future work introducing new preference datasets should report the marginal gain of using the entire dataset on different models compared to using just 10-25% of the dataset.

Secondly, we find all four of the preference datasets we examine to be extremely noise invariant. We attribute this observation to label noise introducing more uncertainty in reward model predictions rather than prediction reversal. This suggests that better preference datasets can tolerate a higher level of label noise. Future work introducing new preference datasets should report the noise invariance of a dataset and the calibration error induced in the downstream reward model.

Lastly, we find that preference datasets vary widely in the distribution of similarity of response pairs. The performance improvements of training from high information or dissimilar response pairs depends on the underlying reward model. An extreme case is if the underlying language model has undergone RLHF policy learning using a preference dataset, then the relative value or information of this dataset should be lower for reward modeling. Recent work has proposed that learning policies from on-policy data outperforms methods using out-of-distribution data Tajwar et al. [2024]. Future work should define and investigate on-policy data for reward model learning in the context of RLHF.

References

J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Y. Bai, A. Jones, K. Ndousse, A. Askell, A. Chen, N. DasSarma, D. Drain, S. Fort, D. Ganguli, T. Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

J. Błasiok, P. Gopalan, L. Hu, and P. Nakkiran. A unifying theory of distance from calibration. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 1727–1740, 2023.

A. Chen, S. Malladi, L. H. Zhang, X. Chen, Q. Zhang, R. Ranganath, and K. Cho. Preference learning algorithms do not learn preference rankings. *arXiv preprint arXiv:2405.19534*, 2024.

W.-L. Chiang, L. Zheng, Y. Sheng, A. N. Angelopoulos, T. Li, D. Li, H. Zhang, B. Zhu, M. Jordan, J. E. Gonzalez, and I. Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024.

G. Cui, L. Yuan, N. Ding, G. Yao, W. Zhu, Y. Ni, G. Xie, Z. Liu, and M. Sun. Ultrafeedback: Boosting language models with high-quality feedback, 2023.

L. Daniele and Suphavadeeprasit. Amplify-instruct: Synthetically generated diverse multi-turn conversations for efficient llm training. *arXiv preprint arXiv:(coming soon)*, 2023. URL `https://huggingface.co/datasets/LDJnr/Capybara`.

DeepSeek-AI, A. Liu, B. Feng, B. Wang, B. Wang, B. Liu, C. Zhao, C. Dengr, C. Ruan, D. Dai, D. Guo, D. Yang, D. Chen, D. Ji, E. Li, F. Lin, F. Luo, G. Hao, G. Chen, G. Li, H. Zhang, H. Xu, H. Yang, H. Zhang, H. Ding, H. Xin, H. Gao, H. Li, H. Qu, J. L. Cai, J. Liang, J. Guo, J. Ni, J. Li, J. Chen, J. Yuan, J. Qiu, J. Song, K. Dong, K. Gao, K. Guan, L. Wang, L. Zhang, L. Xu, L. Xia, L. Zhao, L. Zhang, M. Li, M. Wang, M. Zhang, M. Zhang, M. Tang, M. Li, N. Tian, P. Huang, P. Wang, P. Zhang, Q. Zhu, Q. Chen, Q. Du, R. J. Chen, R. L. Jin, R. Ge, R. Pan, R. Xu, R. Chen, S. S. Li, S. Lu, S. Zhou, S. Chen, S. Wu, S. Ye, S. Ma, S. Wang, S. Zhou, S. Yu, S. Zhou, S. Zheng, T. Wang, T. Pei, T. Yuan, T. Sun, W. L. Xiao, W. Zeng, W. An, W. Liu, W. Liang, W. Gao, W. Zhang, X. Q. Li, X. Jin, X. Wang, X. Bi, X. Liu, X. Wang, X. Shen, X. Chen, X. Chen, X. Nie, X. Sun, X. Wang, X. Liu, X. Xie, X. Yu, X. Song, X. Zhou, X. Yang, X. Lu, X. Su, Y. Wu, Y. K. Li, Y. X. Wei, Y. X. Zhu, Y. Xu, Y. Huang, Y. Li, Y. Zhao, Y. Sun, Y. Li, Y. Wang, Y. Zheng, Y. Zhang, Y. Xiong, Y. Zhao, Y. He, Y. Tang, Y. Piao, Y. Dong, Y. Tan, Y. Liu, Y. Wang, Y. Guo, Y. Zhu, Y. Wang, Y. Zou, Y. Zha, Y. Ma, Y. Yan, Y. You, Y. Liu, Z. Z. Ren, Z. Ren, Z. Sha, Z. Fu, Z. Huang, Z. Zhang, Z. Xie, Z. Hao, Z. Shao, Z. Wen, Z. Xu, Z. Zhang, Z. Li, Z. Wang, Z. Gu, Z. Li, and Z. Xie. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model, 2024.

H. Dong, W. Xiong, B. Pang, H. Wang, H. Zhao, Y. Zhou, N. Jiang, D. Sahoo, C. Xiong, and T. Zhang. Rlhf workflow: From reward modeling to online rlhf. *arXiv preprint arXiv:2405.07863*, 2024.

A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Sravankumar, A. Korenev, A. Hinsvark, A. Rao, A. Zhang, A. Rodriguez, A. Gregerson, A. Spataru, B. Roziere, B. Biron, B. Tang, B. Chern, C. Caucheteux, C. Nayak, C. Bi, C. Marra, C. McConnell, C. Keller, C. Touret, C. Wu, C. Wong, C. C. Ferrer, C. Nikolaidis, D. Allonsius, D. Song, D. Pintz, D. Livshits, D. Esiobu, D. Choudhary, D. Mahajan, D. Garcia-Olano, D. Perino, D. Hupkes, E. Lakomkin, E. AlBadawy, E. Lobanova, E. Dinan, E. M. Smith, F. Radenovic, F. Zhang, G. Synnaeve, G. Lee, G. L. Anderson, G. Nail, G. Mialon, G. Pang, G. Cucurell, H. Nguyen, H. Korevaar, H. Xu, H. Touvron, I. Zarov, I. A. Ibarra, I. Kloumann, I. Misra, I. Evtimov, J. Copet, J. Lee, J. Geffert, J. Vranes, J. Park, J. Mahadeokar, J. Shah, J. van der Linde, J. Billock, J. Hong, J. Lee, J. Fu, J. Chi, J. Huang, J. Liu, J. Wang, J. Yu, J. Bitton, J. Spisak, J. Park, J. Rocca, J. Johnstun, J. Saxe, J. Jia, K. V. Alwala, K. Upasani, K. Plawiak, K. Li, K. Heafield, K. Stone, K. El-Arini, K. Iyer, K. Malik, K. Chiu, K. Bhalla, L. Rantala-Yeary, L. van der Maaten, L. Chen, L. Tan, L. Jenkins, L. Martin, L. Madaan, L. Malo, L. Blecher, L. Landzaat, L. de Oliveira, M. Muzzi, M. Pasupuleti, M. Singh, M. Paluri, M. Kardas, M. Oldham, M. Rita, M. Pavlova, M. Kambadur, M. Lewis, M. Si, M. K. Singh, M. Hassan, N. Goyal, N. Torabi, N. Bashlykov, N. Bogoychev, N. Chatterji, O. Duchenne, O. Çelebi, P. Alrassy, P. Zhang, P. Li, P. Vasic, P. Weng, P. Bhargava, P. Dubal, P. Krishnan, P. S. Koura, P. Xu, Q. He, Q. Dong, R. Srinivasan, R. Ganapathy, R. Calderer, R. S. Cabral, R. Stojnic, R. Raileanu, R. Girdhar, R. Patel, R. Sauvestre, R. Polidoro, R. Sumbaly, R. Taylor, R. Silva, R. Hou, R. Wang, S. Hosseini, S. Chennabasappa, S. Singh, S. Bell, S. S. Kim, S. Edunov, S. Nie, S. Narang, S. Raparthy, S. Shen, S. Wan, S. Bhosale, S. Zhang, S. Vandenhende, S. Batra, S. Whitman, S. Sootla, S. Collot, S. Gururangan, S. Borodinsky, T. Herman, T. Fowler, T. Sheasha, T. Georgiou, T. Scialom, T. Speckbacher, T. Mihaylov, T. Xiao, U. Karn, V. Goswami, V. Gupta, V. Ramanathan, V. Kerkez, V. Gonguet, V. Do, V. Vogeti, V. Petrovic, W. Chu, W. Xiong, W. Fu, W. Meers, X. Martinet, X. Wang, X. E. Tan, X. Xie, X. Jia, X. Wang, Y. Goldschlag, Y. Gaur, Y. Babaei, Y. Wen, Y. Song, Y. Zhang, Y. Li, Y. Mao, Z. D. Coudert, Z. Yan, Z. Chen, Z. Papakipos, A. Singh, A. Grattafiori, A. Jain, A. Kelsey, A. Shajnfeld, A. Gangidi, A. Victoria, A. Goldstand, A. Menon, A. Sharma, A. Boesenberg, A. Vaughan, A. Baevski, A. Feinstein, A. Kallet, A. Sangani, A. Yunus, A. Lupu, A. Alvarado, A. Caples, A. Gu, A. Ho, A. Poulton, A. Ryan, A. Ramchandani, A. Franco, A. Saraf, A. Chowdhury, A. Gabriel, A. Bharambe, A. Eisenman, A. Yazdan, B. James, B. Maurer, B. Leonhardi, B. Huang, B. Loyd, B. D. Paola, B. Paranjape, B. Liu, B. Wu, B. Ni, B. Hancock, B. Wasti, B. Spence, B. Stojkovic, B. Gamido, B. Montalvo, C. Parker, C. Burton, C. Mejia, C. Wang, C. Kim, C. Zhou, C. Hu, C.-H. Chu, C. Cai, C. Tindal, C. Feichtenhofer, D. Civin, D. Beaty, D. Kreymer, D. Li, D. Wyatt, D. Adkins, D. Xu, D. Testuggine, D. David, D. Parikh, D. Liskovich, D. Foss, D. Wang, D. Le, D. Holland, E. Dowling, E. Jamil, E. Montgomery, E. Presani, E. Hahn, E. Wood, E. Brinkman, E. Arcaute, E. Dunbar, E. Smothers, F. Sun, F. Kreuk, F. Tian, F. Ozgenel, F. Caggioni, F. Guzmán, F. Kanayet, F. Seide, G. M. Florez, G. Schwarz, G. Badeer, G. Swee, G. Halpern, G. Thattai, G. Herman, G. Sizov, Guangyi, Zhang, G. Lakshminarayanan, H. Shojanazeri, H. Zou, H. Wang, H. Zha, H. Habeeb, H. Rudolph, H. Suk, H. Aspegren, H. Goldman, I. Damlaj, I. Molybog, I. Tufanov, I.-E. Veliche, I. Gat, J. Weissman, J. Geboski, J. Kohli, J. Asher, J.-B. Gaya, J. Marcus, J. Tang, J. Chan, J. Zhen, J. Reizenstein, J. Teboul, J. Zhong, J. Jin, J. Yang, J. Cummings, J. Carvill, J. Shepard, J. McPhie, J. Torres, J. Ginsburg, J. Wang, K. Wu, K. H. U, K. Saxena, K. Prasad, K. Khandelwal, K. Zand, K. Matosich, K. Veeraraghavan, K. Michelena, K. Li, K. Huang, K. Chawla, K. Lakhotia, K. Huang, L. Chen, L. Garg, L. A, L. Silva, L. Bell, L. Zhang, L. Guo, L. Yu, L. Moshkovich, L. Wehrstedt, M. Khabsa, M. Avalani, M. Bhatt, M. Tsimpoukelli, M. Mankus, M. Hasson, M. Lennie, M. Reso, M. Groshev, M. Naumov, M. Lathi, M. Keneally, M. L. Seltzer, M. Valko, M. Restrepo, M. Patel, M. Vyatskov, M. Samvelyan, M. Clark, M. Macey, M. Wang, M. J. Hermoso, M. Metanat, M. Rastegari, M. Bansal, N. Santhanam, N. Parks, N. White, N. Bawa, N. Singhal, N. Egebo, N. Usunier, N. P. Laptev, N. Dong, N. Zhang, N. Cheng, O. Chernoguz, O. Hart, O. Salpekar, O. Kalinli, P. Kent, P. Parekh, P. Saab, P. Balaji, P. Rittner, P. Bontrager, P. Roux, P. Dollar, P. Zvyagina, P. Ratanchandani, P. Yuvraj, Q. Liang, R. Alao, R. Rodriguez, R. Ayub, R. Murthy, R. Nayani, R. Mitra, R. Li, R. Hogan, R. Battey, R. Wang, R. Maheswari, R. Howes, R. Rinott, S. J. Bondu, S. Datta, S. Chugh, S. Hunt, S. Dhillon, S. Sidorov, S. Pan, S. Verma, S. Yamamoto, S. Ramaswamy, S. Lindsay, S. Lindsay, S. Feng, S. Lin, S. C. Zha, S. Shankar, S. Zhang, S. Zhang, S. Wang, S. Agarwal, S. Sajuyigbe, S. Chintala, S. Max,

S. Chen, S. Kehoe, S. Satterfield, S. Govindaprasad, S. Gupta, S. Cho, S. Virk, S. Subramanian, S. Choudhury, S. Goldman, T. Remez, T. Glaser, T. Best, T. Kohler, T. Robinson, T. Li, T. Zhang, T. Matthews, T. Chou, T. Shaked, V. Vontimitta, V. Ajayi, V. Montanez, V. Mohan, V. S. Kumar, V. Mangla, V. Albiero, V. Ionescu, V. Poenaru, V. T. Mihailescu, V. Ivanov, W. Li, W. Wang, W. Jiang, W. Bouaziz, W. Constable, X. Tang, X. Wang, X. Wu, X. Wang, X. Xia, X. Wu, X. Gao, Y. Chen, Y. Hu, Y. Jia, Y. Qi, Y. Li, Y. Zhang, Y. Zhang, Y. Adi, Y. Nam, Yu, Wang, Y. Hao, Y. Qian, Y. He, Z. Rait, Z. DeVito, Z. Rosnbrick, Z. Wen, Z. Yang, and Z. Zhao. The llama 3 herd of models, 2024.

K. Ethayarajh, Y. Choi, and S. Swayamdipta. Understanding dataset difficulty with $\mathcal{V}$-usable information. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5988–6008. PMLR, 17–23 Jul 2022.

D. Groeneveld, I. Beltagy, P. Walsh, A. Bhagia, R. Kinney, O. Tafjord, A. H. Jha, H. Ivison, I. Magnusson, Y. Wang, et al. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*, 2024.

C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

J. Ji, D. Hong, B. Zhang, B. Chen, J. Dai, B. Zheng, T. Qiu, B. Li, and Y. Yang. Pku-saferlhf: A safety alignment preference dataset for llama family models. *arXiv preprint arXiv:2406.15513*, 2024.

J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

H. R. Kirk, A. Whitefield, P. Röttger, A. Bean, K. Margatina, J. Ciro, R. Mosquera, M. Bartolo, A. Williams, H. He, et al. The prism alignment project: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models. *arXiv preprint arXiv:2404.16019*, 2024.

A. Köpf, Y. Kilcher, D. von Rütte, S. Anagnostidis, Z. R. Tam, K. Stevens, A. Barhoum, D. Nguyen, O. Stanley, R. Nagyfi, et al. Openassistant conversations-democratizing large language model alignment. *Advances in Neural Information Processing Systems*, 36, 2024.

N. Lambert and R. Calandra. The alignment ceiling: Objective mismatch in reinforcement learning from human feedback. *arXiv preprint arXiv:2311.00168*, 2023.

N. Lambert, V. Pyatkin, J. Morrison, L. Miranda, B. Y. Lin, K. Chandu, N. Dziri, S. Kumar, T. Zick, Y. Choi, et al. Rewardbench: Evaluating reward models for language modeling. *arXiv preprint arXiv:2403.13787*, 2024.

Z. Lan. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

T. Moskovitz, A. K. Singh, D. Strouse, T. Sandholm, R. Salakhutdinov, A. D. Dragan, and S. McAleer. Confronting reward model overoptimization with constrained rlhf. *arXiv preprint arXiv:2310.04373*, 2023.

R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

Z. Nussbaum, J. X. Morris, B. Duderstadt, and A. Mulyar. Nomic embed: Training a reproducible long context text embedder. *arXiv preprint arXiv:2402.01613*, 2024.

L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

M. Paul, S. Ganguli, and G. K. Dziugaite. Deep learning on a data diet: Finding important examples early in training. *Advances in neural information processing systems*, 34:20596–20607, 2021.

G. Penedo, H. Kydlíček, A. Lozhkov, M. Mitchell, C. Raffel, L. Von Werra, T. Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *arXiv preprint arXiv:2406.17557*, 2024.

B. Pikus, W. LeVine, T. Chen, and S. Hendryx. A baseline analysis of reward models' ability to accurately analyze foundation models under distribution shift. *arXiv preprint arXiv:2311.14743*, 2023.

N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL `https://arxiv.org/abs/1908.10084`.

D. Rolnick, A. Veit, S. Belongie, and N. Shavit. Deep learning is robust to massive label noise. *arXiv preprint arXiv:1705.10694*, 2017.

B. Sorscher, R. Geirhos, S. Shekhar, S. Ganguli, and A. Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35: 19523–19536, 2022.

N. Stiennon, L. Ouyang, J. Wu, D. M. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, and P. Christiano. Learning to summarize from human feedback. In *NeurIPS*, 2020.

H. Su, W. Shi, J. Kasai, Y. Wang, Y. Hu, M. Ostendorf, W.-t. Yih, N. A. Smith, L. Zettlemoyer, and T. Yu. One embedder, any task: Instruction-finetuned text embeddings. *arXiv preprint arXiv:2212.09741*, 2022.

F. Tajwar, A. Singh, A. Sharma, R. Rafailov, J. Schneider, T. Xie, S. Ermon, C. Finn, and A. Kumar. Preference fine-tuning of llms should leverage suboptimal, on-policy data. *arXiv preprint arXiv:2404.14367*, 2024.

H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Z. Wang, Y. Dong, O. Delalleau, J. Zeng, G. Shen, D. Egert, J. J. Zhang, M. N. Sreedhar, and O. Kuchaiev. Helpsteer2: Open-source dataset for training top-performing reward models. *arXiv preprint arXiv:2406.08673*, 2024.

M. Xia, S. Malladi, S. Gururangan, S. Arora, and D. Chen. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024.

S. M. Xie, H. Pham, X. Dong, N. Du, H. Liu, Y. Lu, P. S. Liang, Q. V. Le, T. Ma, and A. W. Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. *Advances in Neural Information Processing Systems*, 36, 2024.

W. Xiong, H. Dong, C. Ye, Z. Wang, H. Zhong, H. Ji, N. Jiang, and T. Zhang. Iterative preference learning from human feedback: Bridging theory and practice for rlhf under kl-constraint. In *Forty-first International Conference on Machine Learning*, 2024.

B. Zhang, Z. Liu, C. Cherry, and O. Firat. When scaling meets llm finetuning: The effect of data, model and finetuning method. *arXiv preprint arXiv:2402.17193*, 2024a.

P. Zhang, G. Zeng, T. Wang, and W. Lu. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*, 2024b.

S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P. S. Koura, A. Sridhar, T. Wang, and L. Zettlemoyer. Opt: Open pre-trained transformer language models, 2022.

R. Zheng, S. Dou, S. Gao, Y. Hua, W. Shen, B. Wang, Y. Liu, S. Jin, Q. Liu, Y. Zhou, et al. Secrets of rlhf in large language models part i: Ppo. *arXiv preprint arXiv:2307.04964*, 2023.

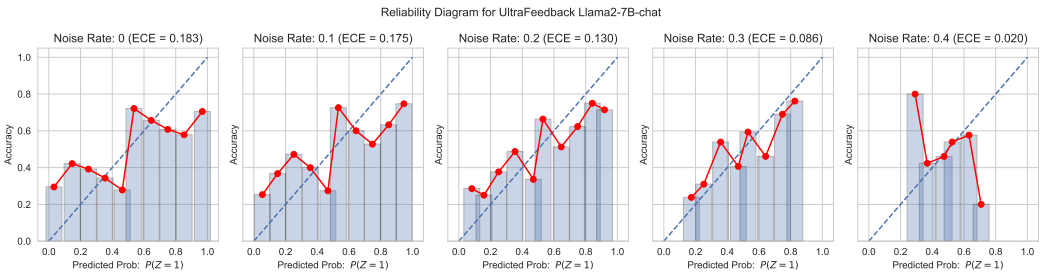

Figure 6: Reliability diagram illustrating expected calibration error (ECE) at different levels of noise for ULTREAFEEDBACK on RewardBench examples. More noise decreases calibration error.

A Noise: Calibration and Reward Modeling

Thus far, very few works have explored the notion of calibration in reward models. For pairwise preferences, we can think of a reward model as a binary predictor where the notion of calibration is rather natural. Expected calibration error, while suffering from real drawbacks Błasiok et al. [2023], is the most commonly used metric for measuring miscalibration Guo et al. [2017]. To compute calibration error, bins can created such that for a bin B_m, the confidence of the bin is just averaged the predicted probability:

$$\text{conf}(B_m) = \frac{1}{B_m} \sum_{i \in B_m} \hat{p}_i,$$

and the accuracy of bin B_m is the average accuracy of samples in the confidence bin range:

$$\text{acc}(B_m) = \frac{1}{B_m} \sum_{i \in B_m} 1[\hat{y}_i = y_i],$$

where y_i is the predicted label. In this setup, a perfectly calibrated predictor would have matching confidence and accuracy for each bin. In other words, the expected calibration error is the difference between the accuracy and confidence in each bin:

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|$$

A problem arises when computing this quantity for reward models on preference data if $\hat{p}_i$ is naively taken to be $\hat{p}_i = P(y_{w_i} \succ y_{l_i})$. This is because by definition, if $P(y_{w_i} \succ y_{l_i}) > 0.5$, $\hat{y}_i = y_i$. This means that zero calibration error can only be achieved through $P(y_{w_i} \succ y_{l_i}) \in \{0, 1\}$ with a perfect predictor.

The only prior work that studies calibration in reward models suggests computing the model probability as Pikus et al. [2023]:

$$\hat{p}_i = \max\{P(y_{w_i} \succ y_{l_i}), P(y_{l_i} \succ y_{w_i})\}.$$

This gives the right intuition that if $\hat{p}_i \approx 0.5$: we should be very uncertain of the outcome. However, this approach restricts $\hat{p}_i \in [0.5, 1]$. Thus, we suggest an alternative approach in creating another random variable $z \in \{0, 1\}$ to randomize the label outcomes so that each example has the following format (x, y_1, y_2, z). Each example, (x, y_w, y_l) becomes the following two examples: $(x, y_w, y_l, z = 1)$ and $(x, y_l, y_w, z = 0)$. Now we have the confidence of a bin as:

$$\text{conf}(B_m) = \frac{1}{B_m} \sum_{i \in B_m} Pr[z_i = 1] = Pr(y_1 \succ y_2)$$

and the accuracy of a bin as:

$$\text{acc}(B_m) = \frac{1}{B_m} \sum_{i \in B_m} z_i$$

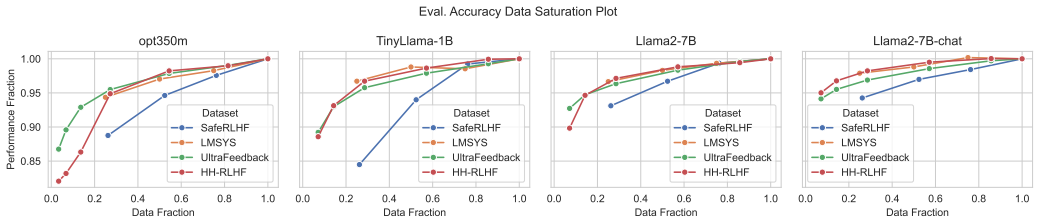

Figure 7: Comparing evaluation accuracy data fraction vs performance fraction, SAFERLHF is the slowest to achieve $> 95\%$ of total accuracy, requiring at least 50% of the dataset. In comparison, other datasets like HH-RLHF only require 10-25% of the dataset depending on the model.

This approach gives us the reliability diagram in Figure 6. We can see that as label noise increases, calibration error decreases. A trivial predictor can achieve zero ECE by always predicting the average of the labels. The figure shows that as a dataset approaches 50% noise, $Pr[z_i = 1]$ collapses to values near 0.5 for all examples. We encourage future work to continue investigating the calibration of reward models through our proposed method.

This transformation we suggest can be done in five simple lines of code:

```
p_chosen = sigmoid(w_rewards - l_rewards)   #P(yw > yl)
chosen_labels = np.ones(len(p_chosen))
p_rejected = 1-p_chosen
rejected_labels = 1-chosen_labels
# Some function that computes the ece given probabilities and true labels
compute_ece(y_pred=np.concatenate([p_chosen, p_rejected]),
y_true=np.concatenate([chosen_labels, rejected_labels]))
```

B Scale: Percentile Saturation

Our work compares preference datasets of vastly different sizes. In our main paper, we present two approaches, the *scaling law approach* of looking at how the performance of each dataset changes with increasing data (Figure 1) and the *benchmarking approach* where we plot the performance of different datasets for each task (Figure 2). Here we would like to present a third choice of *data saturation curves*. On the y-axis we plot the percentage of total performance achieved and on the x-axis we plot the percentage of total data used. This allows us to compare the data efficiency of datasets. In Figure 7, the first observation we can make is that while the shape of the slope of each line becomes flatter with large models, the ordering of datasets remains the same. This allows us to observe that across models of vastly different sizes, SAFERLHF is a dataset that is not very redundant. This is not an artifact of dataset size since LMSYS is approximately the same size.

C Dataset and Experiment Details

Our work looks at 4 different openly available preference datasets. We excluded preference datasets collected or derived from Reddit data due to recent restrictions with respect to terms of service. Specifically, the four datasets we used came from the following hugging face dataset URLs:

- HH-RLHF: `Anthropic/hh-rlhf`
- ULTRAFEEDBACK: `RLHFlow/UltraFeedback-preference-standard`
- LMSYS `lmsys/lmsys-arena-human-preference-55k`
- SAFERLHF `RLHFlow/PKU-SafeRLHF-30K-standard`

To ensure minimal data discrepancies between models, we filtered out examples longer than 512 tokens according to each model tokenizer. We also removed ties from the LMSYS dataset.

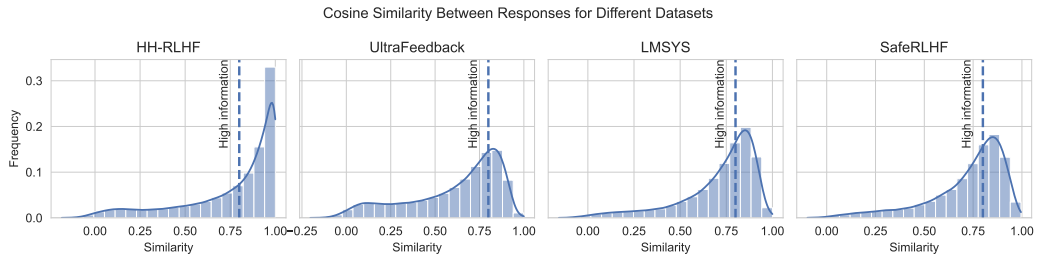

Figure 8: Distribution of Cosine Similarity of winning and losing responses across datasets. HH-RLHF contains many more similar pairs than other datasets

.

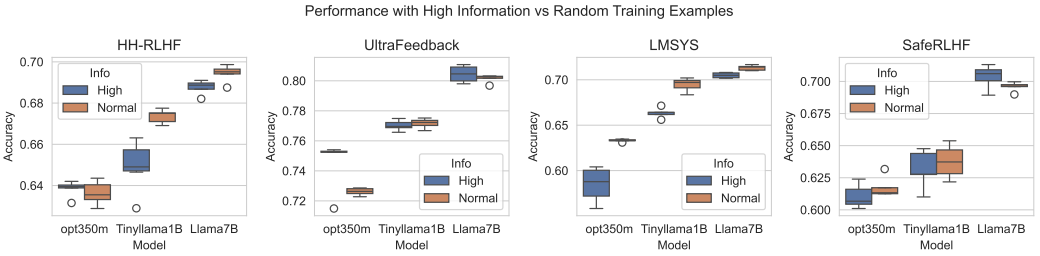

Figure 9: The evaluation set accuracy for training different models with "high information" or low response similarity data compared to a random sample. The benefits of "high information" are most salient in the smallest model.

C.1 Response Pair Distances

For computing distances between responses, we compared several different sentence embeddings. We compared instruction embeddings Su et al. [2022], retrieval embeddingsNussbaum et al. [2024], as well as general-purpose embeddingsLan [2019], Reimers and Gurevych [2019]. We found that cosine and Euclidian distances derived from all of them were highly correlated. Thus, we used a general-purpose pre-trained model: `all-MiniLM-L6-v2`. Using embeddings from this model, Figure 8 shows the contrast in response similarity between different datasets. We see that HH-RLHF contains many more similar winning-losing response pairs compared to other datasets. Furthermore, even though LMSYS responses are generated from a much more diverse set of models than the other three datasets, there are still more similar responses than dissimilar responses. We expect forum-based preference datasets such as Stanford Human Preferences to follow a vastly different distribution of response similarity.

Designating a threshold of 0.8 in similarity, we consider examples that are below 0.8 to be high information. Training on a subset of high-information examples, we compare the downstream performance with a random sample of the training set. While our initial hypothesis may be that training with high information examples would benefit downstream performance, we see that this is only true for small models such as opt350m (Figure 9). One explanation for this effect is that the embeddings used are trained with only 1B pairs on a $\leq$33M[6] parameter model. Once reward models are adapted from base modes with billions of parameters trained with trillions of tokens, these metrics of similarity might not be useful. An alternative explanation is that the value or information content of pairs of examples may depend on the base model itself. Future work should investigate model-dependent data valuation for preference data.

[6] `https://huggingface.co/microsoft/MiniLM-L12-H384-uncased`

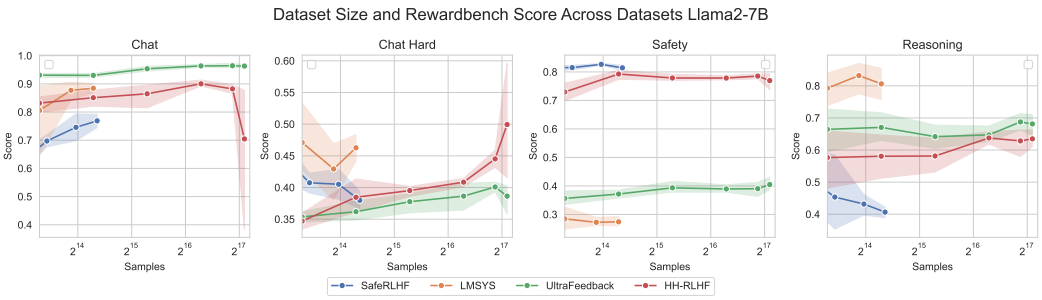

Figure 10: Comparing RewardBench performance across different datasets for Llama2-7B Model. Increasing the dataset size is insufficient to close the performance gap between datasets the best dataset depends on the evaluation task within RewardBench.

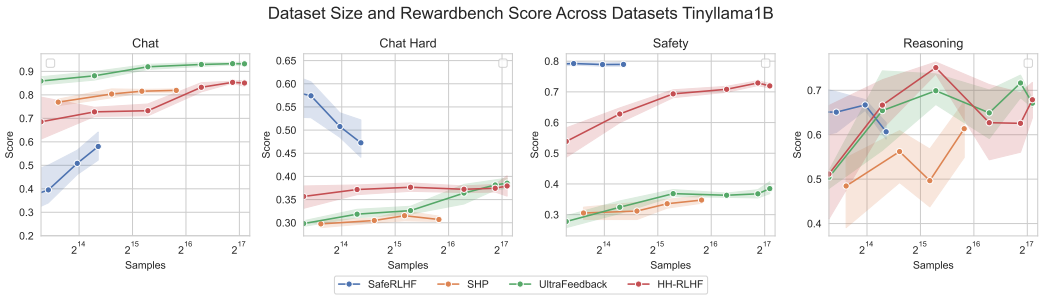

Figure 11: Comparing RewardBench performance across different datasets for TinyLlama-1B Model. Increasing the dataset size is insufficient to close the performance gap between datasets the best dataset depends on the evaluation task within RewardBench.

D Complete Results

D.1 Dataset Scaling

In the main text we show the OOD performance for the Llama2-7B-Chat model. We also include the Llama2-7B base model (Figure 10) where we see the same pattern of UltraFeedback dominating the chat category and SafeRLHF dominating the safety category of Rewardbench. For smaller models, Figure 11, shows a similar pattern for TinyLlama-1B and Figure 12. In these smaller models, the advantage of the SafeRLHF is even more stark.

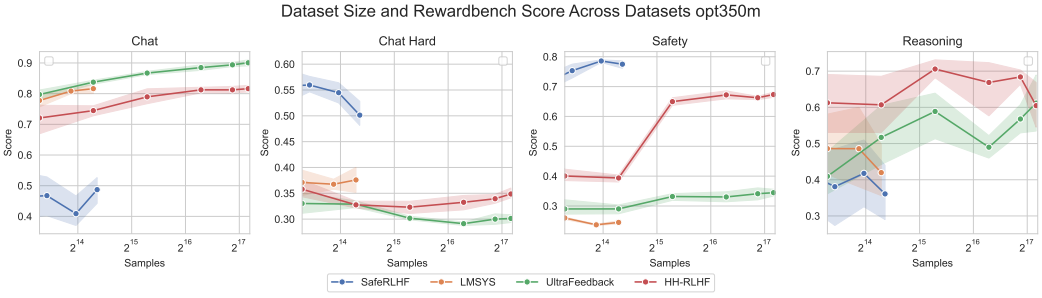

Figure 12: Comparing RewardBench performance across different datasets for OPT350M Model. Increasing the dataset size is insufficient to close the performance gap between datasets the best dataset depends on the evaluation task within RewardBench.

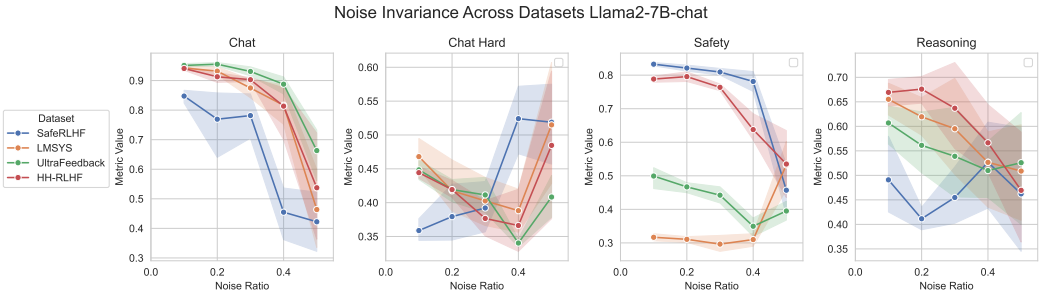

Figure 13: Comparing RewardBench performance across different datasets for Llama2-7B-chat reward model for different levels of label noise. Performance is relatively stable until 30% of labels have been flipped.

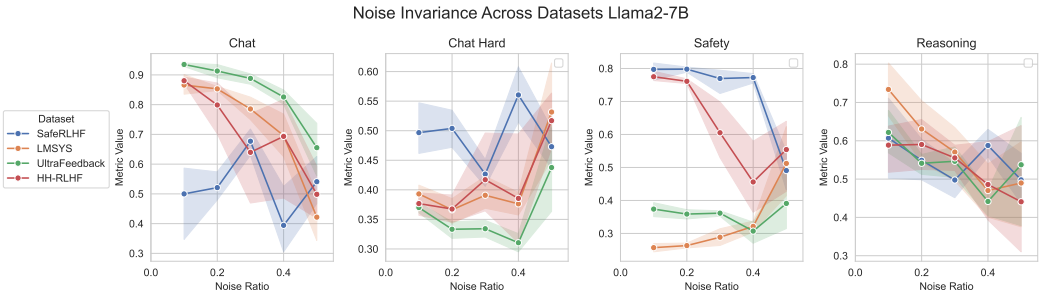

Figure 14: Comparing RewardBench performance across different datasets for Llama2-7B reward model for different levels of label noise. Performance is relatively stable until 30% of labels have been flipped.

D.2 Noise Invariance

We also include plots of the effect of dataset label noise on Rewardbench tasks. For all of the models, the Chat and Safety tasks are not significantly affected until 40% of the labels are flipped. For the Chat Hard and reasoning tasks, most of the models we train are not good enough to examine differences properly. It is also interesting that we do not observe cross-over behavior; no dataset starts with a worse performance and improves over a different dataset at a higher level of noise.

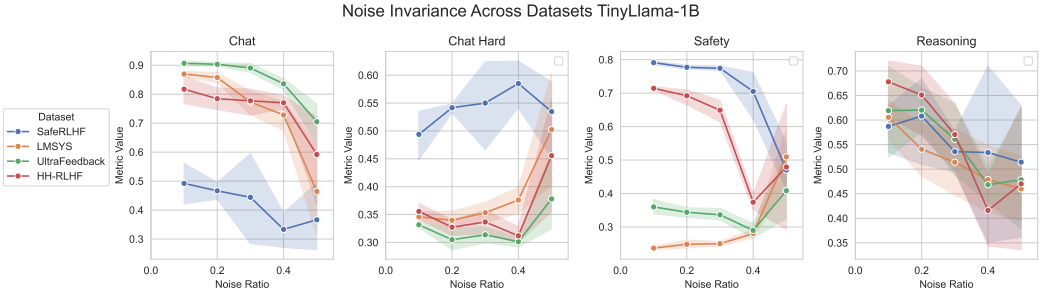

Figure 15: Comparing RewardBench performance across different datasets for TinyLlama-1B reward model for different levels of label noise. Performance is relatively stable until 30% of labels have been flipped.

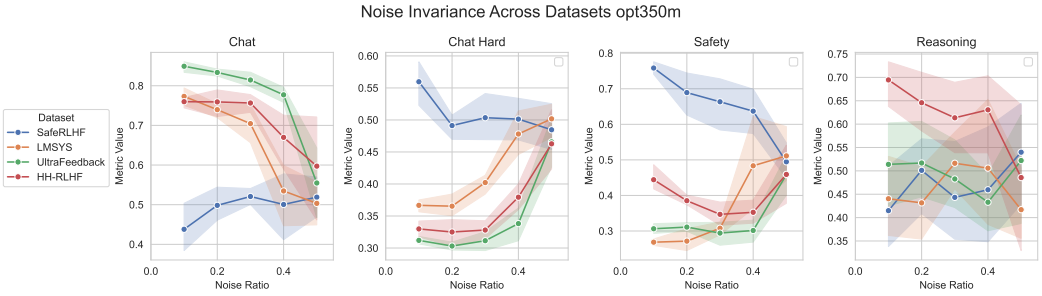

Figure 16: Comparing RewardBench performance across different datasets for Opt-350m reward model for different levels of label noise. Performance is relatively stable until 30% of labels have been flipped.