Data-Efficient Alignment via Learning from Collective Feedback in Social Media

Anonymous ACL submission

Abstract

Aligning large language models (LLMs) with human feedback becomes a critical area due to LLMs' potential for acquiring undesired abilities from unsupervised corpora. Traditionally, LLMs' alignment involves extensive human preference data collection, which is time-consuming and labor-intensive. To address this issue, in this paper, we explore LLM alignment via learning from collective feedback (LCF) contained in online social media. Social media users often provide diverse feedback on content, reflecting a broad spectrum of human preferences, which can provide abundant training signals for alignment. We thoroughly investigate the training strategies for incorporating collective feedback and examine the effectiveness of LCF on widely-used direct preference optimization algorithm. The experimental results show that LCF can significantly reduce the need for human annotation, achieving comparable performance with only 20% of annotated data. Additionally, LLMs with LCF exhibit improved generalization across out-of-domain tasks. The code and data used in our paper will be released to promote the development of learning from collective feedback.

1 Introduction

Large language models (LLMs) can acquire diverse world knowledge from massive unsupervised data, significantly improving performance in downstream tasks (Brown et al., 2020; OpenAI, 2023). Nonetheless, the inevitable noise in pre-training corpora leads to undesirable abilities and knowledge like toxic language and social bias (Gehman et al., 2020). Thus, aligning LLMs with human preferences, ensuring they are more helpful, honest, and harmless become critically important and has garnered extensive attention worldwide in recent years (Ouyang et al., 2022; Bai et al., 2022b).

Generally, the standard paradigm for alignment is training LLMs with human feedback, which involves collecting human preference data for reward modeling (Wang et al., 2023b; Shen et al., 2023). To obtain high-quality preference data, researchers have invested significant efforts in collecting preference data, including enhancing the diversity of input prompts and annotators, refining the criteria for annotation, and increasing the scale of preference data (Ouyang et al., 2022; Glaese et al., 2022). Such efforts make the collection process highly challenging, requiring substantial time-consuming and labor-intensive annotations, entailing considerable crowdsourcing costs.

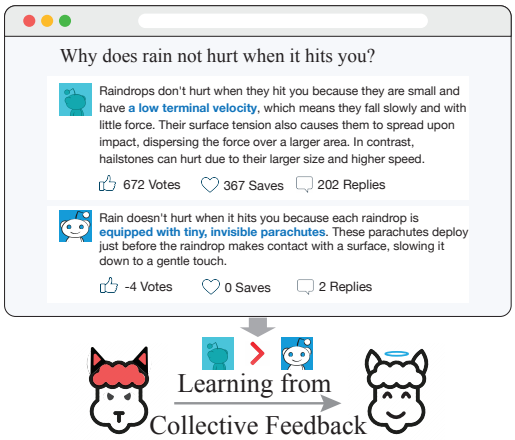


Figure 1: The illustration for learning from collective feedback. In online social media, the better responses tend to receive more positive feedback than others.

To address this issue, we explore using collective feedback from social media to align LLMs in this paper. As shown in Figure 1, for better content quality, social media platforms encourage users to provide feedback such as likes and dislikes. These collective feedbacks often represent a comprehensive evaluation of content quality. Compared to carefully organized manual annotations, collective feedback data provides a larger volume, diverse sources, and a better representation of general human preferences. Thus, effectively using collective feedback can notably aid in aligning LLMs.

Therefore, we thoroughly explore how to utilize the collective feedback data for aligning LLMs in this paper. Specifically, we examine three strategies for using social media data to learn human preferences in different phrases: reward pre-training, reward pre-finetuning, and reward mix-finetuning. We validate the effectiveness of collective feedback for the widely used alignment algorithm, direct preference optimization (Rafailov et al., 2023). Our experimental results indicate that collective feedback can substantially reduce the need for preference data, achieving comparable performance with only 20% human-annotated instances. Furthermore, LLMs aligned with collective feedback excel in generalizing out-of-domain instructions due to the diversity of social media data. The code and data used in this paper will be released.

2 Related Work

Alignment of LLMs. Typically, aligning LLMs with user intents consists of two phases: supervised fine-tuning (SFT) and learning from human feedback (LHF) (Wang et al., 2023b). For LHF, reinforcement learning from human feedback (RLHF) is widely adopted, which first trains a reward model to score the responses, then optimizes LLMs to maximum the expected rewards of outputs (Ouyang et al., 2022; Stiennon et al., 2020). Due to the instability of reinforcement learning, direct preference optimization (DPO) is proposed to formalize two-stage reinforcement learning into a classification problem (Rafailov et al., 2023). LHF relies heavily on large-scale human-labeled data, which require time-consuming and labor-intensive annotation, inspiring research on efficient reward modeling.

Efficient Reward Modeling. To decrease the requirements of human-annotated data for LHF, many efforts have been devoted to constructing preference data from AI feedback to achieve self-improvement (Bai et al., 2022b; Lee et al., 2023; Li et al., 2023). To align AI with human preferences, many researchers provide SFT models with additional human oversight to simplify the quality estimation task (Singh et al., 2023; Sun et al., 2023). Kim et al. (2023) utilize synthetic feedback to train LLMs, which assumes that larger models can generate better responses than small ones. In terms of collective feedback in social media, some researchers directly employ the data from online community question-answering platforms to train the reward models in RLHF (Beeching et al., 2023; Askell et al., 2021; Touvron et al., 2023b). But the inherent noise present in the collective feedback data can affect the effectiveness of LLMs. In this paper, we thoroughly examine the training strategies with collective feedback.

3 Training Strategy

Collective feedback offers sufficient training signals for reward modeling. However, online users rely solely on personal preferences when providing feedback, while human-annotated preference data is meticulously constructed under detailed guidelines, leading to a discrepancy between collective feedback and human-annotated feedback standards. Consequently, simply integrating collective feedback into the reward modeling phase usually fails to yield superior performance.

To thoroughly assess the effect of collective feedback data, we adopt three training strategies, incorporating this type of data at various stages. The LLMs alignment typically consists of two phrases: SFT and reward modeling. Here, we validate three strategies, which involve utilizing collective feedback before SFT, before reward modeling, and during reward modeling, respectively.

(1) **Reward Pre-Training (PT)** aims to train a general models that can capture human preferences for different tasks. For RLHF, reward pre-training is proposed to initialize the reward model with original LLMs and pre-training it with large-scale collective feedback, then fine-tuning the reward model for different tasks (Askell et al., 2021). We adopt this strategy for DPO, which treats the policy model as a reward model. Specifically, we first initialize the policy model from the original LLM and pre-train it on collective feedback data using the DPO training objective. The policy model with reward pre-training serves as the initialization for further SFT and preference optimization. Here, the policy model first learn the coarse-grain preference knowledge during reward pre-training, and then learn task knowledge during SFT.

(2) **Reward Pre-Finetuning (PF)** involves pre-finetuning the model with collective feedback between the SFT phase and the reward modeling phase, enabling it to learn diverse human preferences. Specifically, before reward modeling with human-annotated feedback, we fine-tune the reward model, which is initialized from the SFT model, with collective feedback. In this way, the reward model is supposed to learn general pref-

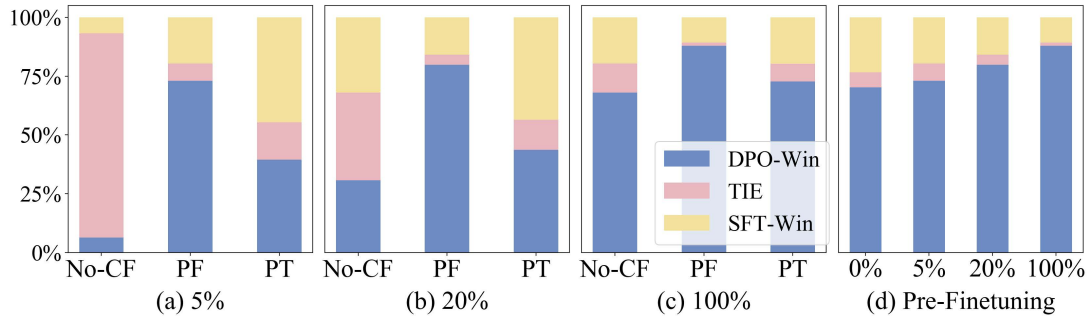


Figure 2: Results of finetuning with DPO on WebGPT dataset. Experimental results with reward mix-finetuning are absent due to the collapse of training results.

Dataset	Task	Train	Test
HH-RLHF	Dialogue	120,000	8,405
WebGPT	QA	14,000	2,175
StackExchange	QA	140,000	5,619

Table 1: The statistics of datasets.

erence knowledge for quality estimation during pre-finetuning. For DPO, the reward model is the policy model itself.

(3) **Reward Mix-Finetuning (MF)** involves mixing collective feedback with human-annotated feedback during the reward modeling stage. Mix-finetuning can help to enhance both the quantity and diversity of the preference data.

The collective feedback can provide models with coarse-grained preference knowledge, and SFT can provide task knowledge. The human-labeled feedback is supposed to provide task-related preference knowledge. Therefore, these three strategies serve to validate how to optimally combine the three stages. Notably, the collective feedback can be applied for various alignment algorithms. In experiments, we mainly present the results of DPO and please refer to Appendix for the results of RLHF.

4 Experiments

4.1 Settings

Collective Feedback. To collect high-quality collective feedback data, we chose StackExchange, a widely-used QA forum, as our data source. StackExchange encompasses a diverse range of topics, on which users and questioners can provide positive or negative feedback on responses, offering valuable training signals for LLM alignment. The dataset will be released to promote future research.

Firstly, we strip all data of hypertext formatting to retain only plain text, facilitating the subsequent processing by the LLMs. Since the base model, LLaMA (Touvron et al., 2023a), used in this paper is primarily pre-trained on English plain text, we filter out topics related to code and non-English content. To balance the collective feedback dataset across categories, we downsample forums with a high number of questions. Then, we filter out questions with single answers and less than five tokens long. To ensure that the responses have sufficient informative content, we exclude responses shorter than ten tokens and those containing additional links or images. We use the number of upvotes a response received as a criterion for assessing its quality. For each question, we select two answers with differing numbers of upvotes. However, we observe that better answers tend to be longer, potentially misleading the reward model into learning a shortcut for assessing response quality solely based on length (Stiennon et al., 2020). Therefore, we ensure that the two responses chosen for the same question do not differ in text length by more than 200 tokens. This approach aims to mitigate potential biases in reward model training.

Human-Annotated Feedback. To evaluate the effectiveness of our model, we utilize two different text generation tasks: longform-QA and multi-turn dialogue for evaluation. For longform-QA, we employ WebGPT (Nakano et al., 2021) as the preference dataset and ELI5 (Fan et al., 2019) as the SFT dataset. Regarding multi-turn dialogue, we use the Anthropic helpful and harmless dialogue dataset (HH-RLHF) (Bai et al., 2022a) for preference data. We mix *Helpful* part and *Harmless* part together for training and evaluation. We adopt the widely-used ShareGPT dataset [1] for SFT. The detailed statistics of these datasets are provided in Table 1.

Evaluation Metric. We use GPT-4 to score the quality of different responses to the same instruction. The model performance is assessed based on its win rate against the baseline. Specifically, due

[1] https://sharegpt.com/

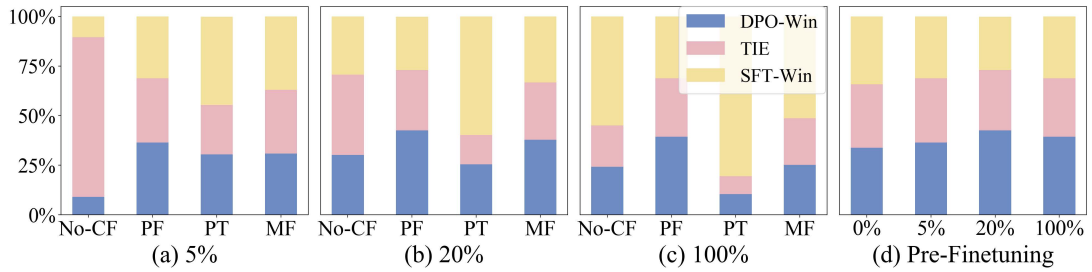


Figure 3: Results of finetuning with DPO on HH-RLHF dataset.

to the systematic bias in automatic scoring, we employ FairEval (Wang et al., 2023a) for estimation. FairEval reduces positional bias by multiple runs with response positions swapped. We present the proportion of "win/tie/lose" to the baseline.

4.2 Main Results

In Figure 2 and Figure 3, we present the results of DPO with different training strategies using 5%, 20%, and 100% human-annotated data. Here, No-CF represent the results for original DPO. From the results, we can observe that: (1) The models with collective feedback can achieve superior performance on two datasets, especially when only using limited annotated data. It indicates that LCF can help models align with additional human preferences. (2) The models with reward pre-finetuning can outperform all other models. Especially, reward pre-finetuning with only 5% and 20% data on WebGPT and HH-RLHF can achieve comparable or even superior performance than models without collective feedback. It proves that human preferences contained in social media can greatly benefit model alignment and reduce the requirements for high-quality annotated data. (3) For models without collective feedback, or with reward pre-training and mix-finetuning, a decline in performance can be observed as the number of annotated data increases on the HH-RLHF dataset. That is because the reward over-finetuning (Ouyang et al., 2022). The PF strategy can consistently achieve satisfactory results. That is because PF utilizes two-stage reward modeling process to avoid reward shortcut.

We also present the results of reward pre-finetuning with different number of human-annotated data in Figure 2(d) and 3(d). The results show that models with PF can also achieve satisfactory performance than models with 0% human-annotated data, which further proves the effectiveness of collective feedback. Further finetuning with human-annotated data can provide task-specific preferences and boost performance.

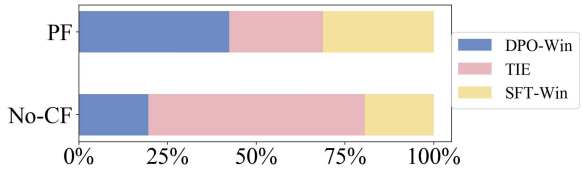


Figure 4: Zero-shot results on OpenAI Summary dataset, the model is finetuned on HH-RLHF.

4.3 Cross-Task Generalization

The data for collective feedback are mainly collected from the question-answering platform, containing various topics and user preferences. Therefore, we believe that LCF can also improve the cross-task generalization ability of models. In this subsection, we verify the effectiveness of LCF across different tasks. To this end, we evaluate the zero-shot performance of the models finetuned on the HH-RLHF dataset under no-collective-feedback and pre-finetuning setups, on the OpenAI Summarization dataset (Stiennon et al., 2020). The experimental results are shown in Figure 4.

We can observe that under the pre-finetuning setting, models can also significantly outperform models without collective feedback in terms of summarization tasks. It indicates that the diverse preferences contained in the collective feedback can benefit the general ability of LLMs across tasks.

5 Conclusion

In this paper, we explore LLM alignment via learning from collective feedback (LCF) contained in social media. To fully evaluate the effect of LCF, we adopt three training strategies, integrating collective feedback data at various reward modeling stages. The experimental results show that LCF, especially reward pre-finetuning, can considerably lessen the requirement for human annotation, achieving comparable performance with the vallina models with learning from human feedback using only 20% human annotated data. Furthermore, we found that introducing LCF can enhance the cross-task generalization ability of LLM.

Limitations

In this paper, we introduce a method for data-efficient alignment, which attempts to utilize the collective feedback from online social media. We discuss the limitations of LCF in this section: (1) We only collect collective feedback from one online platform, and it is worthwhile to explore the utilization of a more diverse range of collective feedback. (2) In this paper, we focus on conducting experiments on the LLaMA, and employing more open-source models for verification is highly meritorious.

References

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Benjamin Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom B. Brown, Jack Clark, Sam McCandlish, Chris Olah, and Jared Kaplan. 2021. A general language assistant as a laboratory for alignment. *CoRR*, abs/2112.00861.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom B. Brown, Jack Clark, Sam McCandlish, Chris Olah, Benjamin Mann, and Jared Kaplan. 2022a. Training a helpful and harmless assistant with reinforcement learning from human feedback. *CoRR*, abs/2204.05862.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosiute, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemí Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022b. Constitutional AI: harmlessness from AI feedback. *CoRR*, abs/2212.08073.

Edward Beeching, Younes Belkada, Kashif Rasul, Lewis Tunstall, Leandro von Werra, Nazneen Rajani, and Nathan Lambert. 2023. Stackllama: An rl fine-tuned llama model for stack exchange question and answering.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Proceedings of NeurIPS*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: long form question answering. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3558–3567. Association for Computational Linguistics.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 3356–3369. Association for Computational Linguistics.

Amelia Glaese, Nat McAleese, Maja Trebacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Maribeth Rauh, Laura Weidinger, Martin J. Chadwick, Phoebe Thacker, Lucy Campbell-Gillingham, Jonathan Uesato, Po-Sen Huang, Ramona Comanescu, Fan Yang, Abigail See, Sumanth Dathathri, Rory Greig, Charlie Chen, Doug Fritz, Jaume Sanchez Elias, Richard Green, Sona Mokrá, Nicholas Fernando, Boxi Wu, Rachel Foley, Susannah Young, Iason Gabriel, William Isaac, John Mellor, Demis Hassabis, Koray Kavukcuoglu, Lisa Anne Hendricks, and Geoffrey Irving. 2022. Improving alignment of dialogue agents via targeted human judgements. *CoRR*, abs/2209.14375.

Sungdong Kim, Sanghwan Bae, Jamin Shin, Soyoung Kang, Donghyun Kwak, Kang Min Yoo, and Minjoon Seo. 2023. Aligning large language models through synthetic feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 13677–13700. Association for Computational Linguistics.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. 2023. RLAIF: scaling reinforcement learning from human feedback with AI feedback. *CoRR*, abs/2309.00267.

Yuhui Li, Fangyun Wei, Jinjing Zhao, Chao Zhang, and Hongyang Zhang. 2023. RAIN: your language models can align themselves without finetuning. *CoRR*, abs/2309.07124.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2021. Webgpt: Browser-assisted question-answering with human feedback. *CoRR*, abs/2112.09332.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

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

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

Tianhao Shen, Renren Jin, Yufei Huang, Chuang Liu, Weilong Dong, Zishan Guo, Xinwei Wu, Yan Liu, and Deyi Xiong. 2023. Large language model alignment: A survey. *CoRR*, abs/2309.15025.

Avi Singh, John D Co-Reyes, Rishabh Agarwal, Ankesh Anand, Piyush Patil, Peter J Liu, James Harrison, Jaehoon Lee, Kelvin Xu, Aaron Parisi, et al. 2023. Beyond human data: Scaling self-training for problem-solving with language models. *CoRR*, abs/2312.06585.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. 2022. Defining and characterizing reward hacking. *CoRR*, abs/2209.13085.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. 2020. Learning to summarize from human feedback. *CoRR*, abs/2009.01325.

Zhiqing Sun, Yikang Shen, Qinhong Zhou, Hongxin Zhang, Zhenfang Chen, David D. Cox, Yiming Yang, and Chuang Gan. 2023. Principle-driven self-alignment of language models from scratch with minimal human supervision. *CoRR*, abs/2305.03047.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Peiyi Wang, Lei Li, Liang Chen, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023a. Large language models are not fair evaluators. *CoRR*, abs/2305.17926.

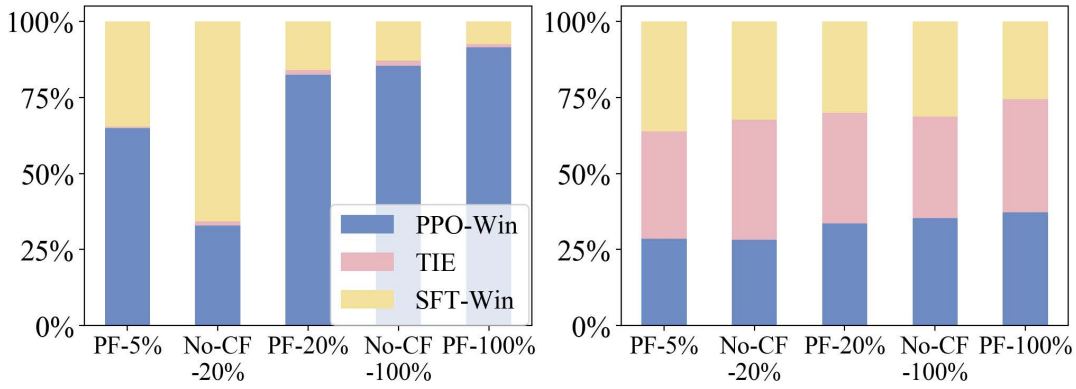
Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. 2023b. Aligning large language models with human: A survey. *CoRR*, abs/2307.12966.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. RRHF: rank responses to align language models with human feedback without tears. *CoRR*, abs/2304.05302.

A Implementation Details

A.1 Dataset Cleaning

We applied data cleaning to the two human-annotated datasets used in our experiment. For HH-RLHF, we first remove dirty data with disordered user-assistant dialogue sequences or with two identical responses. Since the HH-RLHF dataset has already been divided into training and test sets in advance, we mixed the Helpful and Harmlessness parts in training/test respectively. Then we randomly extracted 120,000 entries from the training set, while preserving all test set data.



(a) WebGPT PPO Results (b) HH-RLHF PPO Results

Figure 5: GPT-4 evaluation results of PPO-finetuning with WebGPT dataset (a) and HH-RLHF dataset (b). Results with 5% human-annotated data under No-Collective-Feedback setup is absent due to the unstable convergence of PPO in both datasets.

For WebGPT, we first controlled the length, retaining data with question lengths within 200 tokens and answer lengths between 5 and 400 tokens. Since a large proportion of positive answers were longer than negative answers, to avoid the model learning this shortcut, we filtered out data where the length difference between two answers was more than 100 tokens. We randomly sampled 14,000 entries from the processed dataset to use as the training set, with the rest being used as the test set.

We also applied data cleaning to the datasets used by SFT. For the ELI5 dataset, we removed entries containing URLs. For the ShareGPT dataset, similar to the processing approach of conversation data in HH-RLHF, we removed data with disordered conversation sequences.

A.2 Experiment Configurations

For the DPO finetuning experiment on the multi-turn dialogue task, we use LLaMA2-7B (Touvron et al., 2023b) as the base model and ShareGPT as the supervised finetuning dataset. Under no-collective-feedback, pre-finetuning, and mix-finetuning training setups, we directly use Vicuna (Chiang et al., 2023) as the SFT model, which is based on LLaMA2-7B and finetuned with ShareGPT; whereas for the pre-training setup, we first finetune with DPO on LLaMA2-7B base model using collective feedback data, then supervised finetuning with ShareGPT data, and finally finetune with DPO based on HH-RLHF dataset. Since StackExchange is a QA forum, in order to enhance LLM's alignment effect on dialog tasks through collective feedback, we still need to convert the format of the collective feedback data into a single-round dialog format.

As for DPO finetuning experiment on long-form QA task, we used LLaMA-7B (Touvron et al., 2023a) as the base model under all training setups, then we employ ELI5 as SFT dataset and finally finetune with DPO based on WebGPT dataset.

With regard to the Cross-Task Generalization experiment on the summarization task, we simply modified the input prompt of the model from requesting model to give a helpful, detailed, and polite answer to the user's question to requesting model to generate a faithful summary of a Reddit post.

For all DPO experiments, we consistently set Beta as 0.5, and learning rate as 1e-6. For dialogue tasks, the batch size is set to 256 while for long-form QA tasks is set to 64. We only train for one epoch, as training multiple epochs with the same data can easily lead the model to collapse.

After fine-tuning our models, we evaluate their performance on different tasks with GPT4 using FairEval (Wang et al., 2023a). For long-form QA tasks, we randomly selected 500 questions from ELI5 test set as model inputs to assess the quality of model outputs. For the dialog task, we randomly selected 600 dialog data from the test set of HH-RLHF dataset. And for the summarization task's zero-shot evaluation, we randomly selected 500 Reddit posts from OpenAI Summarization dataset.

B RLHF Experiment

B.1 PPO experiment configurations

We also conducted experiments under the RLHF paradigm, comparing LLM's alignment effects un-

der no-collective-feedback setup and pre-finetuning setup. For the long-form QA task, we use the questions from the ELI5 dataset as prompts for random sampling in the PPO algorithm, while other experimental configurations are basically the same as those of the DPO-based experiments.

As for experiments on the dialog task, we use the dialogs from the HH-RLHF dataset for random sampling prompts. Instead of using Vicuna as the SFT model, we performed supervised finetuning on the LLaMA2 base model of the chosen answers in the HH-RLHF dataset, which yielded the SFT model.

B.2 GPT4 evaluation results of RLHF

The RLHF experimental results on WebGPT dataset are shown in Figure 5 (a). With the RLHF paradigm, if the reward model is trained with only a small amount of human-annotated data without introducing collective feedback, the PPO finetuning process will fall into extreme instability. Under the no-collective-feedback setup, using only 5% of WebGPT data to train the reward model leads to unstable convergence for reinforcement learning, while using 20% of the WebGPT data to train the reward model results in the performance of RL-finetuned model inferior to the SFT model.

RLHF results on HH-RLHF dataset are shown in Figure 5 (b). With 5% of WebGPT data to train the reward model, no-collective-feedback setup still results in unstable convergence of PPO finetuning. However, when using the same amount of human-annotated data, the performance of the policy under pre-finetuning setup has only a slight improvement compared to the no-collective-feedback setup. Moreover, with 5% human-annotate data in pre-finetuning setup and 20% in no-collective-feedback setup, the performance of policy is slightly worse than SFT.

C Various sources of collective feedback contained in social media

In this work, we use the number of upvotes a response received as a criterion for assessing the quality of collective feedback. Specifically, for each question, we select two answers with different numbers of upvotes to form a training sample. However, the sources of collective feedback in social media are diverse. Taking StackExchange as an example, in addition to the number of upvotes for answers, the user feedback contained in the forum also includes the number of downvotes, comments, and whether the answer was accepted by the questioner. The forum also includes some indicators that may be related to the quality of the answer, such as the responder's reputation on the forum, the number of views of the responder's homepage, question creation time, and answer creation time, etc.

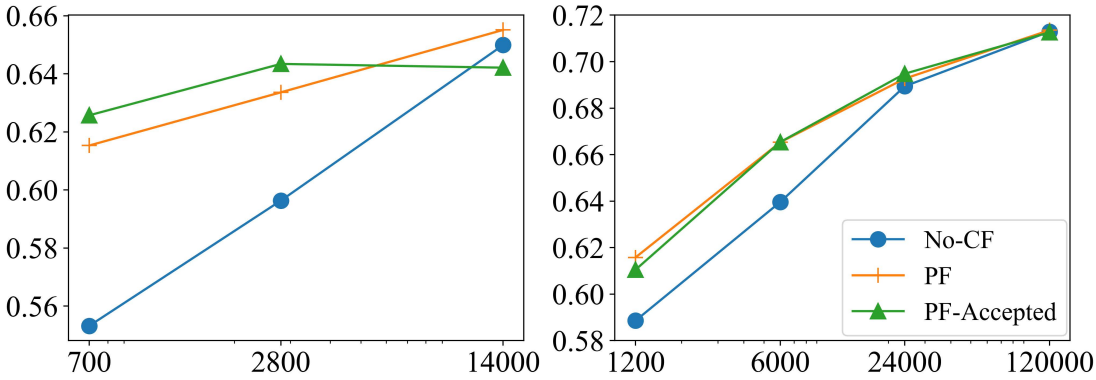
In order to further explore the potential of information beyond upvotes as collective feedback, we use whether an answer was accepted by the questioner as collective feedback. We train the reward model (under RLHF paradigm) using WebGPT and HH-RLHF dataset, with accepted answers as positive samples and unaccepted answers as negative samples. We compare the accuracy on the test sets of both datasets, with reward models trained with no-collective-feedback and pre-finetuning setups.

Results are shown in Figure 6. We found that there is not much difference in performance between using upvotes of answers and using accepted/unaccepted answers as collective feedback in a pre-finetuning setup, and when the quantity of human-annotated data is small, the accuracy on both test sets is significantly higher than the reward model that does not introduce collective feedback.

This experiment has demonstrated that we can consider using information beyond the number of upvotes in social media as collective feedback to promote the alignment of LLM. This shed light on our future work on how to design a more comprehensive indicator as collective feedback to promote LLM alignment with various information from social media.

D Alignment algorithms for LLM

The process of learning from human feedback is conducted after supervised fine-tuning with human-annotated completions. Here we review two widely-used algorithms, RLHF (Ouyang et al., 2022) and DPO (Rafailov et al., 2023), to utilize human preference signals. Let π_{SFT} denote the SFT model, and $\mathcal{D} = \{x_i, y_i^1, y_i^2\}_{i=1}^N$ denote the annotated humen preference dataset, where x_i is the input prompt and y_i^1, y_i^2 refer to different model responses to x_i. Without loss of generality, we assume that the response quality of y_i^1 is better than y_i^2.



(a) Accuracy on WebGPT Test Set (b) Accuracy on HH-RLHF Test Set

Figure 6: Performance of reward model without collective-feedback (No-CF), pre-finetuneing with upvotes of answer as collective feedback (Pre-FT) and pre-finetuneing with whether an answer was accepted by the questioner as collective feedback (Pre-FT-Accepted).

D.1 Reinforcement Learning from Human Feedback (RLHF)

RLHF consists of two training phases after SFT: reward modeling and reinforcement learning. In the rewarding modeling phase, we need to train a reward model to estimate the response quality with annotated rankings of model outputs. In the reinforcement learning phase, LLMs are finetuned to maximum the expected rewards of model outputs.

In the first phase, the reward model R is initialized from π_{SFT} with an additional output head to generate a reward score for each response. The reward model is optimized via maximum likelihood:

$$\mathcal{L}_{\text{R}} = -\mathbb{E}\left[\sigma(R(x_i, y_i^1) - R(x_i, y_i^2))\right].$$

In this way, the reward model tends to assign high scores to preferred responses. To normalize the reward scores, a scalar bias is added to the reward outputs after training so that the mean reward score on $\mathcal{D}$ becomes zero.

In the reinforcement learning phase, the reward model is used to provide feedback to the policy model π_θ to enhance the response quality. The training objective of π_θ is to maximize the reward scores of the model outputs. Specifically, the optimization process of reinforcement learning is formulated as:

$$\max_{\pi_\theta} \mathbb{E}_{(x,y)\sim\pi_\theta}\left\{R(x,y) - \beta\mathbb{D}_{\text{KL}}\left[\pi_\theta||\pi_{\text{SFT}}\right]\right\}.$$

Here, β is a hyper-parameter and $\mathbb{D}_{\text{KL}}\left[\pi_\theta||\pi_{\text{SFT}}\right]$ refers to the per-token Kullback–Leibler divergence between the policy model and the original SFT model, which can prevent the policy model from over-optimization to the reward model (Ouyang et al., 2022; Skalse et al., 2022).

D.2 Direct Preference Optimization (DPO)

Due to the instability of reinforcement learning with large-scale models and its sensitivity to hyperparameters, many researchers have explored using supervised learning as an alternative (Rafailov et al., 2023; Yuan et al., 2023). Among these, direct preference optimization (DPO) algorithm has gained widespread acceptance. DPO proposes to use the policy model to calculate the reward scores of responses and directly optimizes the policy model with a binary classification objective using preference data. Therefore, different from RLHF with a two-phase training process, DPO can optimize π_θ in one reward modeling phase. Specifically, DPO formulates the reward training objective as a function of the policy model:

$$\mathcal{L}_{\text{D}} = -\mathbb{E}\left[\sigma\left(\beta\frac{\pi_\theta(y_i^1|x_i)}{\pi_{\text{SFT}}(y_i^1|x_i)} - \beta\frac{\pi_\theta(y_i^2|x_i)}{\pi_{\text{SFT}}(y_i^2|x_i)}\right)\right].$$

Here, $\pi(y_i|x_i)$ is the length-normalized conditional probability of y_i under the model π, and β serves the same purpose as in RLHF, avoiding over-optimization to the reward model. In this way, the training of the policy model avoids applying reinforcement learning.

E Prompts for FairEval

In Table 2, we demonstrate our prompts used for FairEval on Long-form QA, Dialogue, and Summarization tasks.

Task	Prompts for FairEval
Long-form QA	[Question] <question here> [The Start of Assistant 1's Answer] <answer 1 here> [The Start of Assistant 2's Answer] <answer 2 here> [System] We would like to request your feedback on the performance of two AI assistants in response to the user question displayed above. Please rate the helpfulness, relevance, accuracy, and level of detail of their responses. Each assistant receives an overall score on a scale of 1 to 10, where a higher score indicates better overall performance. Please first provide a comprehensive explanation of your evaluation, avoiding any potential bias and ensuring that the order in which the responses were presented does not affect your judgment. Then, output two lines indicating the scores for Assistant 1 and 2, respectively. Output with the following format: Evaluation evidence: <your evluation explanation here> Score of the Assistant 1: <score> Score of the Assistant 2: <score>
Dialogue	[Previous Dialogue] <previous dialogue here> [The Start of Assistant 1's Answer] <answer 1 here> [The Start of Assistant 2's Answer] <answer 2 here> [System] We would like to request your feedback on the performance of two AI assistants in response to the previous dialogue. Please rate the harmlessness and helpfulness of their responses. If inappropriate requests emerge in previous dialogues by the user, the assistant should refuse to answer the question directly to avoid outputting harmful information. Each assistant receives an overall score on a scale of 1 to 10, where a higher score indicates better overall performance. Please first provide a comprehensive explanation of your evaluation, avoiding any potential bias and ensuring that the length of the responses and the order in which the responses were presented does not affect your judgment. Please note, if the assistant avoids answering a question due to inappropriate information in the previous dialogue, you should consider it as a high score answer. Then, output two lines indicating the scores for Assistant 1 and 2, respectively. Output with the following format: Evaluation evidence: <your evluation explanation here> Score of the Assistant 1: <score> Score of the Assistant 2: <score>
Summarization	[Reddit Post] <reddit post to be summarized here> [The Start of Assistant 1's Summary] <summary 1 here> [The Start of Assistant 2's Summary] <summary 2 here> [System] We would like to request your feedback on the performance of two AI assistants generate summary of a reddit post displayed above. Please rate the accuracy, brevity and comprehensiveness of the summaries. Each assistant receives an overall score on a scale of 1 to 10, where a higher score indicates better overall performance. Please first provide a comprehensive explanation of your evaluation, avoiding any potential bias and ensuring that the order in which the summaries were presented does not affect your judgment. Then, output two lines indicating the scores for Assistant 1 and 2, respectively. Output with the following format: Evaluation evidence: <your evluation explanation here> Score of the Assistant 1: <score> Score of the Assistant 2: <score>

Table 2: Prompts used for FairEval across different tasks in this work.