WildFeedback: Aligning LLMs With In-situ User Interactions And Feedback

Taiwei Shi[*†], **Zhuoer Wang**[*‡], **Longqi Yang**[*◇], **Ying-Chun Lin**[○], **Zexue He**[▽], **Mengting Wan**[◇], **Pei Zhou**[◇], **Sujay Jauhar**[◇], **Xiaofeng Xu**[◇], **Xia Song**[◇], **Jennifer Neville**[*◇]
[◇]Microsoft Corporation, [○]Purdue University, [‡]Texas A&M University,
[▽]University of California San Diego, [†]University of Southern California

Abstract

As large language models (LLMs) continue to advance, aligning these models with human preferences has emerged as a critical challenge. Traditional alignment methods, relying on human or LLM annotated datasets, are limited by their resource-intensive nature, inherent subjectivity, and the risk of feedback loops that amplify model biases. To address these issues, we propose WILDFEEDBACK, a framework that uses real-time user interactions to build preference datasets that better reflect genuine human preferences. The process involves feedback signal identification, preference data construction, and user-guided evaluation. Applied to a large set of user-LLM interactions, WILDFEEDBACK generates datasets that capture nuanced user preferences by analyzing feedback within natural conversations. Our experiments show that LLMs fine-tuned with WILDFEEDBACK align more closely with user needs, as demonstrated by both traditional benchmarks and our user-guided evaluation. By leveraging real-time feedback, WILDFEEDBACK overcomes the limitations of current alignment approaches, offering a scalable, robust solution for developing LLMs that better meet diverse user needs and preferences.

1 WildFeedback

Existing preference datasets often fail to align with real human preferences, as synthetic datasets like ULTRAFEEDBACK (Cui et al., 2024) rely solely on GPT-4, risking the reinforcement of model biases. Human-annotated datasets, while more accurate, are difficult to scale due to resource constraints and the subjectivity of annotators (Bai et al., 2022; Ouyang et al., 2022). To overcome these issues, we introduce WILDFEEDBACK, a novel framework that aligns LLMs with in-situ user interactions. Applied to WildChat (Zhao et al., 2024), this framework yielded a preference dataset of 20,281 samples. The pipeline is shown in Figure 1. WILDFEEDBACK operates through a three-step process: feedback signal identification, preference dataset construction, and user-guided evaluation.

Feedback Signal Identification. To build preference data from human-LLM interactions, we first identify conversations with feedback signals through user satisfaction estimation. Users may express satisfaction (e.g., "thank you") or dissatisfaction (e.g., "revise it") explicitly or implicitly during conversations. Lin et al. (2024b) introduced SPUR, a framework that automatically identifies SAT (satisfaction) and DSAT (dissatisfaction) patterns using rubrics generalized from conversations with annotated feedback by recursively prompting GPT-4. We adopt the SAT/DSAT rubrics from Lin et al. (2024b), using 9 SAT rubrics (e.g., gratitude, praise) and 9 DSAT rubrics (e.g., revision, factual error). These rubrics are fed into GPT-4 to classify utterances at the conversation level, detailed in

[*] Corresponding authors: taiweish@usc.edu, wang@tamu.edu, longqi.yang@microsoft.com, jenneville@microsoft.com. The work was done when Taiwei Shi, Zhuoer Wang, Ying-Chun Lin, and Zexue He were interns at Microsoft Corporation.

NeurIPS 2024 Workshop on Behavioral Machine Learning.

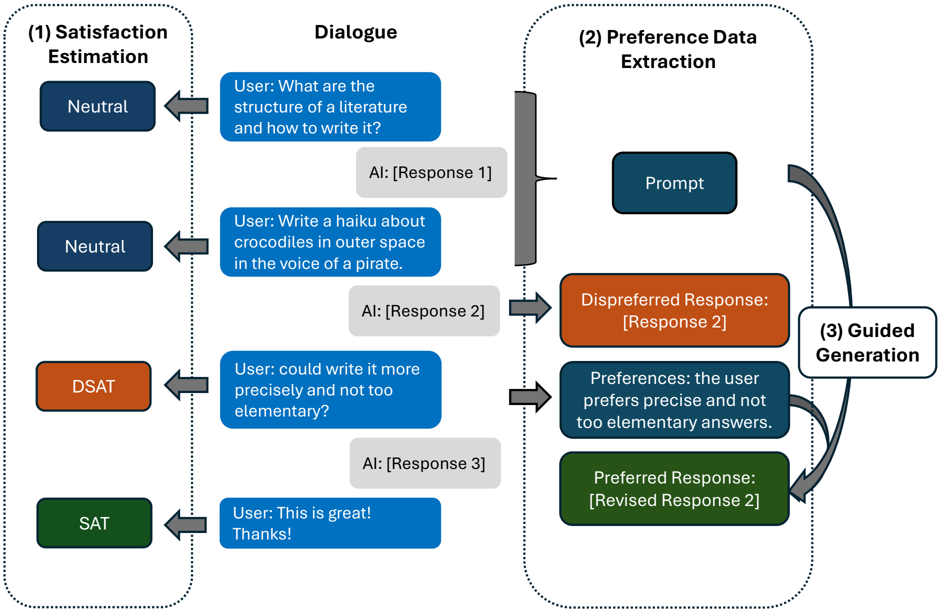

(1) Satisfaction Estimation
Dialogue
(2) Preference Data Extraction
Neutral
User: What are the structure of a literature and how to write it?
AI: [Response 1]
Prompt
Neutral
User: Write a haiku about crocodiles in outer space in the voice of a pirate.
AI: [Response 2]
Dispreferred Response: [Response 2]
(3) Guided Generation
DSAT
User: could write it more precisely and not too elementary?
Preferences: the user prefers precise and not too elementary answers.
AI: [Response 3]
Preferred Response: [Revised Response 2]
SAT
User: This is great! Thanks!

Figure 1: Overview of WILDFEEDBACK.

Appendix B.1. Among 148,715 WildChat multi-turn conversations, about 12.8% contain feedback signals (Table 1). To validate GPT-4's SAT/DSAT classifications, we conducted a human expert review with 50 conversations, yielding a Cohen's Kappa of $\kappa = 0.70$ for SAT and $\kappa = 0.54$ for DSAT. Comparatively, GPT-4 showed high agreement with human annotations, achieving $\kappa = 0.69$ for SAT and $\kappa = 0.50$ for DSAT. Detailed validation results are in Appendix B.2.

Table 1: Statistics of the SAT/DSAT conversations and utterances.

# Conv.	# Utt.	# SAT Conv.	# DSAT Conv.	# SAT Utt.	# DSAT Utt.
148,715	628,467	5,447	13,582	8,186	27,711

Preference Data Construction. After identifying feedback signals using SAT/DSAT rubrics, we construct a preference dataset comprising four components: the prompt, user preferences, the preferred response, and the dispreferred response. For conversations with SAT/DSAT signals, we extract the conversation up to the model response that triggered the feedback as the prompt. We then use GPT-4 to summarize user preferences from these signals (e.g., a preference for concise answers), allowing us to pinpoint which responses led to satisfaction or dissatisfaction. For generating preferred and dispreferred responses, we use two approaches: expert and on-policy. Expert responses are generated by GPT-4, while on-policy responses are generated by Phi 3 (Abdin et al., 2024), Mistral (Jiang et al., 2023), and LLaMA 3 (Dubey et al., 2024). In the expert approach, original responses that triggered DSAT signals are used as dispreferred, and GPT-4 generates preferred responses based on summarized preferences. For on-policy responses, the policy model generates both preferred and dispreferred responses, guided by user preferences for the preferred outputs. To ensure safety, especially when user preferences may be harmful (e.g., explicit content), we add an instruction: "the response should be safe" when generating preferred responses. Additionally, some conversations are filtered by the OpenAI moderation API. This method builds a robust dataset that helps models better align with user preferences while maintaining safety standards. Details on the prompt used for constructing preference data are in Appendix A.2.

User-guided Evaluation To better assess model alignment with user preferences, we implement user-guided evaluation alongside preference data construction. Existing benchmarks like AlpacaEval (Dubois et al., 2024) and MT-Bench (Zheng et al., 2023b) rely on LLMs as judges, which often leads to biased evaluations favoring longer responses or those generated by LLMs themselves (Liu et al., 2024b; Thakur et al., 2024). This can result in evaluations misaligned with actual user preferences. Similarly, evaluations by human annotators can be flawed due to subjective biases that may not reflect true user needs. To address these issues, we employ user-guided evaluation that focuses on real user preferences rather than the subjective ranking of responses. Annotators are guided to rank responses based on summarized preferences derived from user feedback rather than their own biases. For LLM evaluators, we provide an instance-level checklist based on these summarized user preferences to guide assessments. Our evaluation framework adapts from WILDBENCH (Lin et al., 2024a), which aligns well with human judgment in ranking model performance. We use a pairwise evaluation strategy where GPT-4 compares two responses using a preference-guided checklist to determine which performs better, providing clear win/lose rates for straightforward comparisons. The full evaluation prompt is detailed in Appendix A.3.

WILDFEEDBACK Data. To demonstrate that the generated preferred responses align with actual user preferences, we randomly selected 500 samples from the WILDFEEDBACK datasets and performed user-guided evaluation, comparing the preferred and dispreferred responses. As shown in Figure 4, we found that without providing the summarized user preferences as checklists, GPT-4 tends to prefer the dispreferred responses in our dataset, which are the model's zero-shot generations without guidance from summarized user preferences. However, after providing the preferences as checklists to guide the evaluation, GPT-4's selections more closely align with real users' preferences. Additionally, we observed that GPT-4 is significantly more steerable than smaller models: over 70% of its preferred responses align with in-situ user preferences, compared to only about 50% for smaller models. Consequently, for on-policy data, we additionally filter out any data that does not align with user preferences. A detailed analysis of the data can be found in Appendix C.

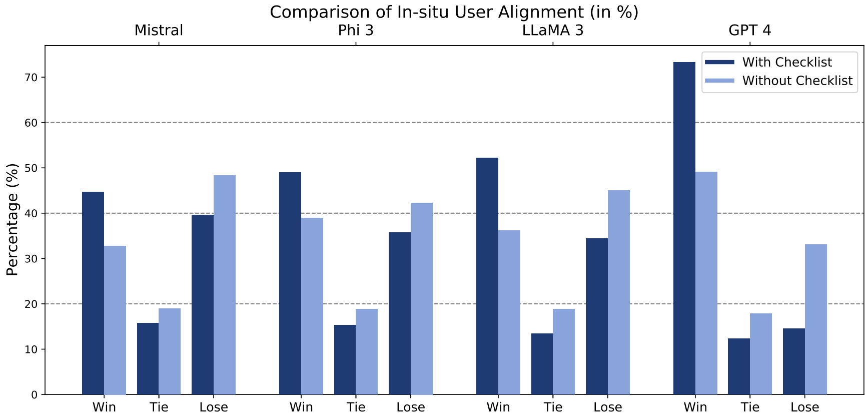


Figure 2: Comparison of in-situ user alignment across datasets generated by different models. "Win/Tie/Lose" represents the percentage of instances where the preferred responses win/tie/lose compared to the dispreferred responses in the WILDFEEDBACK dataset, prior to filtering.

2 Experiment

To validate the effectiveness of WILDFEEDBACK, We finetune Mistral, Phi 3, LLaMA 3 on both the GPT-4 and on-policy version of WILDFEEDBACK and compare their performances with the non-finetuned models. We first perform 1 epoch of supervised fine-tuning (SFT) on the preferred responses, followed by 1 epoch of direct preference optimization (DPO) (Rafailov et al., 2023) on the entire dataset. For more details, please refer to Appendix D.

We evaluate our models on MT-Bench (Zheng et al., 2023a), AlpacaEval 2 (Li et al., 2023), Arena-Hard (Li et al., 2024), and the held-out test set of WILDFEEDBACK. For WILDFEEDBACK evaluation, we report the win, tie, lose percentage against the off-the-shelf instruct models with GPT-4 as the judge. Results are shown in Table 2. Implementation details on the benchmarks can be found in Appendix E.

3 Result

In this section, we present the main results of our experiments, highlighting the effectiveness of WildFeedback on various benchmarks and ablation studies.

Training models on the GPT-4 version of WildFeedback can significantly and consistently boost model performance across all benchmarks. As shown in Table 2, models trained with the GPT-4 version of WildFeedback exhibit higher win rates across AlpacaEval 2, Arena-Hard, and MT-Bench, as well as improved performance in both settings of WildFeedback (with and without a checklist). For instance, Phi 3's win rate on AlpacaEval 2 increases from 17.39% to 36.6%, and its win rate on Arena-Hard improves from 15.4% to 32.4%. Additionally, Phi 3's performance on MT-Bench also sees an increase, with its score rising from 7.32 to 7.73. These consistent performance boosts across various benchmarks demonstrate that the GPT-4 version of WildFeedback is an effective tool for enhancing model performance and aligning it more closely with user preferences across diverse tasks.

WildFeedback significantly enhances model alignment with in-situ user feedback. As detailed in Section 1, WildFeedback has two versions, differing in whether the preferred responses are generated by GPT-4 or the policy models themselves. Compared to off-the-shelf instruction models, those trained on either version of WildFeedback demonstrate a stronger alignment with real user preferences. For example, LLaMA 3 trained on the on-policy version of WildFeedback wins against the off-the-shelf LLaMA 3 model 57.2% of the time, while only losing 28.3% of the time. Notably, even without user preferences provided as checklists during GPT-4 evaluation, the model still performs on par with or better than the off-the-shelf version, underscoring the robustness of this training approach.

WildFeedback does not compromise model performance on other benchmarks. Training on either version of WildFeedback not only aligns models more closely with user preferences but also does not compromise performance on other benchmarks; in most cases, it even leads to improvements. For instance, LLaMA 3 trained on the on-policy version of WildFeedback improves its length-controlled win rate (LC) on AlpacaEval 2 from 22.9% to 30.1% and its raw win rate (WR) from 22.6% to 29.6%. Similarly, Phi 3 shows an increase in its Arena-Hard win rate from 15.4% to 22.0% after training on the on-policy version. This indicates that the models are better tuned to real-world interactions without sacrificing their overall versatility or effectiveness across a range of tasks. These results demonstrate that WildFeedback provides a valuable framework for refining models to better meet user expectations while maintaining, and often enhancing, their general performance across various benchmarks.

Table 2: AlpacaEval 2, Arena-Hard, MT-Bench, and WildFeedback results under the four settings. LC and WR denote length-controlled and raw win rate, respectively. WF On-Policy or WF GPT-4 denotes the model trained on either the on-policy or GPT-4 version of WildFeedback.

Models	**AlpacaEval 2**		**Arena-Hard**	**MT-Bench**	**WildFeedback With Checklist**			**WildFeedback Without Checklist**		
	LC (%)	WR (%)	WR (%)	Score	Win (%)	Tie (%)	Lose (%)	Win (%)	Tie (%)	Lose (%)
Phi 3	24.3	17.39	15.4	7.32	–	–	–	–	–	–
↪ WF On-Policy	24.2	18.3	22.0	7.40	56.5	14.2	29.3	42.5	17.8	39.7
↪ WF GPT-4	34.9	36.6	32.4	7.73	66.6	9.90	23.5	54.2	14.0	31.8
LLaMA 3	22.9	22.6	20.6	7.10	–	–	–	–	–	–
↪ WF On-Policy	30.1	29.6	22.1	7.15	57.2	14.5	28.3	40.9	18.8	40.3
↪ WF GPT-4	34.2	42.8	32.9	7.57	61.8	11.7	26.4	48.1	17.2	34.8
Mistral	17.1	14.7	12.6	6.71	–	–	–	–	–	–
↪ WF On-Policy	12.9	12.3	10.3	6.42	52.1	11.2	36.7	37.4	16.1	46.5
↪ WF GPT-4	31.4	36.1	19.8	6.79	62.8	9.70	27.4	50.4	14.0	35.6

4 Conclusion

In this work, we propose a framework for constructing preference data and evaluating conversational AI models based on natural human-LLM interactions. By using SAT/DSAT rubrics to identify user satisfaction and dissatisfaction in conversations, we create a preference dataset that includes user prompts, preferences, and both preferred and dispreferred responses. This enables models to better align with user expectations. Additionally, we introduce a user-guided evaluation framework that addresses biases in existing benchmarks by using real user feedback to guide LLM evaluations, ensuring a more accurate reflection of user preferences. Our approach emphasizes the importance of aligning AI responses with diverse and inclusive human values, improving overall user satisfaction.

References

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

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 Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022. URL `https://arxiv.org/abs/2204.05862`.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. ULTRAFEEDBACK: Boosting language models with scaled AI feedback. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 9722–9744. PMLR, 21–27 Jul 2024. URL `https://proceedings.mlr.press/v235/cui24f.html`.

Sarkar Snigdha Sarathi Das, Chirag Shah, Mengting Wan, Jennifer Neville, Longqi Yang, Reid Andersen, Georg Buscher, and Tara Safavi. S3-dst: Structured open-domain dialogue segmentation and state tracking in the era of llms, 2023. URL `https://arxiv.org/abs/2309.08827`.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library, 2024.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David

Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov,

Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-controlled alpacaeval: A simple way to debias automatic evaluators, 2024. URL https://arxiv.org/abs/2404.04475.

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

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Minh Nguyen, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Alexandrovich Glushkov, Arnav Varma Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Julian Mattick. Openassistant conversations - democratizing large language model alignment. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=VSJotgbPHF.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline, 2024. URL https://arxiv.org/abs/2406.11939.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval, 5 2023.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. Wildbench: Benchmarking llms with challenging tasks from real users in the wild, 2024a. URL https://arxiv.org/abs/2406.04770.

Ying-Chun Lin, Jennifer Neville, Jack W. Stokes, Longqi Yang, Tara Safavi, Mengting Wan, Scott Counts, Siddharth Suri, Reid Andersen, Xiaofeng Xu, Deepak Gupta, Sujay Kumar Jauhar, Xia Song, Georg Buscher, Saurabh Tiwary, Brent Hecht, and Jaime Teevan. Interpretable user satisfaction estimation for conversational systems with large language models, 2024b. URL https://arxiv.org/abs/2403.12388.

Jie Liu, Zhanhui Zhou, Jiaheng Liu, Xingyuan Bu, Chao Yang, Han-Sen Zhong, and Wanli Ouyang. Iterative length-regularized direct preference optimization: A case study on improving 7b language models to gpt-4 level, 2024a. URL https://arxiv.org/abs/2406.11817.

Yiqi Liu, Nafise Sadat Moosavi, and Chenghua Lin. Llms as narcissistic evaluators: When ego inflates evaluation scores, 2024b. URL https://arxiv.org/abs/2311.09766.

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. Webgpt: Browser-assisted question-answering with human feedback, 2022. URL https://arxiv.org/abs/2112.09332.

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

Jupinder Parmar, Sanjev Satheesh, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. Reuse, don't retrain: A recipe for continued pretraining of language models, 2024. URL https://arxiv.org/abs/2407.07263.

Jiaxin Pei, Aparna Ananthasubramaniam, Xingyao Wang, Naitian Zhou, Apostolos Dedeloudis, Jackson Sargent, and David Jurgens. Potato: The portable text annotation tool. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HPuSIXJaa9.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun

Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Aman Singh Thakur, Kartik Choudhary, Venkat Srinik Ramayapally, Sankaran Vaidyanathan, and Dieuwke Hupkes. Judging the judges: Evaluating alignment and vulnerabilities in llms-as-judges, 2024. URL https://arxiv.org/abs/2406.12624.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. Zephyr: Direct distillation of lm alignment, 2023. URL https://arxiv.org/abs/2310.16944.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. Wildchat: 1m chatgpt interaction logs in the wild, 2024. URL https://arxiv.org/abs/2405.01470.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 46595–46623. Curran Associates, Inc., 2023a. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/91f18a1287b398d378ef22505bf41832-Paper-Datasets_and_Benchmarks.pdf.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023b. URL https://arxiv.org/abs/2306.05685.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL http://arxiv.org/abs/2403.13372.

A Prompts

A.1 Prompt for Feedback Signals Identification

The following is the full prompt we used for dialogue state tracking and SAT/DSAT classification. In addition, we also prompt GPT-4 to do domain and intent classification. The prompt is adapted from Das et al. (2023) and Lin et al. (2024b).

```
## LABEL DEFINITION ##
{
"valid_preceding_topical_relation_labels": [
{
"label": "YES",
"definition": "The current turn has **some or any** topical/subtopical
relation to the preceding conversation context."
},
{
"label": "NO",
"definition": "The current turn has **absolutely no** topical/subtopical
relation to the preceding conversation context OR is the first turn in the
conversation, marking the beginning of a new dialogue segment."
```

```
}
],
"valid_domain_labels": [
"AI MACHINE LEARNING AND DATA SCIENCE",
"ASTROLOGY",
"BIOLOGY AND LIFE SCIENCE",
"BUSINESS AND MARKETING",
"CAREER AND JOB APPLICATION",
"CLOTHING AND FASHION",
"COOKING FOOD AND DRINKS",
"CRAFTS",
"CULTURE AND HISTORY",
"CYBERSECURITY",
"DATING FRIENDSHIPS AND RELATIONSHIPS",
"DESIGN",
"EDUCATION",
"ENTERTAINMENT",
"ENVIRONMENT AGRICULTURE AND ENERGY",
"FAMILY PARENTING AND WEDDINGS",
"FINANCE AND ECONOMICS",
"GAMES",
"GEOGRAPHY AND GEOLOGY",
"HEALTH AND MEDICINE",
"HOUSING AND HOMES",
"HUMOR AND SARCASM",
"LANGUAGE",
"LAW AND POLITICS",
"LITERATURE AND POETRY",
"MANUFACTURING AND MATERIALS",
"MATH LOGIC AND STATISTICS",
"MUSIC AND AUDIO",
"NEWS",
"PETS AND ANIMALS",
"PHILOSOPHY",
"PHYSICS CHEMISTRY AND ASTRONOMY",
"PRODUCTIVITY",
"PSYCHOLOGY AND EMOTIONS",
"RELIGION AND MYTHOLOGY",
"SHIPPING AND DELIVERY",
"SHOPPING AND GIFTS",
"SMALL TALK",
"SOCIAL MEDIA",
"SOFTWARE AND WEB DEVELOPMENT",
"SPORTS AND FITNESS",
"TAXATION",
"TECHNOLOGY",
"TIME AND DATES",
"TRANSPORTATION AUTOMOTIVE AND AEROSPACE",
"TRAVEL",
"VISUAL ARTS AND PHOTOGRAPHY",
"WEATHER",
"WRITING JOURNALISM AND PUBLISHING",
"OTHER"
],
"valid_intent_labels": [
{
"label": "INTENT:1-INFORMATION_SEEKING",
"definition":      "The user wants to find factual information or answers to
specific questions."
```

```
},
{
"label": "INTENT:2-ANALYSIS",
"definition": "The user asks analytical or conceptual questions about
a complex topic or problem. The user’s questions require some degree
of reasoning, interpretation, argumentation, comparison, and/or data
processing."
},
{
"label": "INTENT:3-CREATION",
"definition": "The user asks the agent to either generate original content
or translate existing content into new content based on specified criteria
or constraints."
},
{
"label": "INTENT:4-OPEN-ENDED_DISCOVERY",
"definition": "The user wants to casually chat or play with the
agent out of curiosity, boredom, or humor, OR the user’s intent is so
unclear/underspecified that it’s impossible to categorize in any of the
other intent classes. The user mainly treats the agent as a conversation
or chitchat partner, and none of the other intent categories can be
assigned."
}
],
"valid_satisfaction_labels": [
{
"label": "Gratitude",
"definition": "The user thanks or compliments the AI agent for its
responses"
},
{
"label": "Learning",
"definition": "The user learns something new or useful by indicating
curiosity and satisfaction with the information provided"
},
{
"label": "Compliance",
"definition": "The user follows the AI agent’s suggestions or instructions
when applicable"
},
{
"label": "Praise",
"definition": "The user uses positive feedback words (e.g., excellent,
amazing) or emojis, indicating enthusiasm and enjoyment of the
conversation"
},
{
"label": "Personal_Details",
"definition": "The user shares more personal details or opinions with the AI
agent when satisfied with its responses"
},
{
"label": "Humor",
"definition": "The user jokes with or challenges the AI agent in a friendly
manner when suitable"
},
{
"label": "Acknowledgment",
"definition": "The user acknowledges or confirms that they understood or
```

```
agreed with the AI agent’s explanations when relevant"
},
{
"label": "Positive_Closure",
"definition": "The user ends the conversation on a positive note without
asking for more information or assistance"
},
{
"label": "Getting_There",
"definition": "The user acknowledges that the model’s response is
getting better or has merit but is not fully satisfied. Appropriate
dissatisfaction criteria may need to be checked as well when Getting_There
presents"
},
{
"label": "N/A",
"definition": "The user utterance of the turn does NOT match the definition
of any other valid satisfaction labels"
}
],
"valid_dissatisfaction_labels": [
{
"label": "Negative_Feedback",
"definition": "The user explicitly expresses dissatisfaction, frustration,
annoyance, or anger with the AI agent’s response or behavior"
},
{
"label": "Revision",
"definition": "The user explicitly asks the AI agent to revise its previous
response or repeatedly asks similar questions"
},
{
"label": "Factual_Error",
"definition": "The user points out the AI agent’s factual mistakes,
inaccuracies, or self-contradiction in its information or output"
},
{
"label": "Unrealistic_Expectation",
"definition": "The user has unrealistic expectations of what the AI agent
can do and does not accept its limitations or alternatives"
},
{
"label": "No_Engagement",
"definition": "The user does not respond to the AI agent’s questions,
suggestions, feedback requests, etc."
},
{
"label": "Ignored",
"definition": "The user implies that their query was ignored completely or
that the response did not address their intent/goal at all"
},
{
"label": "Lower_Quality",
"definition": "The user perceives a decline in quality of service compared
to previous experience with other agents/tools, etc."
},
{
"label": "Insufficient_Detail",
"definition": "The user wants more specific/useful information than what is
```

```
provided by the AI agent"
},
{
"label": "Style",
"definition": "The user feels that there is a mismatch between their
preferred style (e.g. bullet point vs paragraph, formal vs casual, short
vs long, etc.) and what is provided by the AI agent"
},
{
"label": "N/A",
"definition": "The user utterance of the turn does NOT match the definition
of any other valid dissatisfaction labels"
}
],
"valid_state_labels": [
{
"label": "FEEDBACK",
"definition": "The user utterance of the turn contains a comment or
evaluation or judgement of the previous turn's agent response"
},
{
"label": "REFINEMENT",
"definition": "The user utterance of the turn is a repetition or refinement
of unclear/underspecified instruction given in the previous turn's user
utterance"
},
{
"label": "NEWTOPIC",
"definition": "The user utterance of the turn is either the first turn
of the conversation or is not related in terms of topic or task to its
previous turn, introducing a new topic or task"
},
{
"label": "CONTINUATION",
"definition": "The user utterance of the turn is a topical or logical
continuation of the previous turn"
}
]
}
```

## TASK ##

You are given a dialogue between a user and an agent comprised of turns starting with T. For each turn, solely based on the turn's User utterance, you must carefully analyze the conversation and answer the following questions by replacing $instruction$ with correct answers in JSON format. - Summarize the user utterance in ≤ 3 sentences

- Analyze the user utterance's relation with the previous turn and output an appropriate label from the "valid_preceding_topical_relation_labels" list.
- Analyze the user utterance's domain and output an appropriate label from the "valid_domain_labels" list. If preceding_topical_relation is YES, the domain label must be consistent with the preceding turn's domain label.
- Analyze the user utterance's intent and output an appropriate label from the "valid_intent_labels" list.
- Analyze the user utterance's satisfaction with respect to the previous turn's AI response and output all applicable labels from the "valid_satisfaction_labels" list.
- Analyze the user utterance's dissatisfaction with respect to the previous turn's AI response and output all applicable labels from the "valid_dissatisfaction_labels" list.
- Analyze the user utterance's state and output an appropriate label from the "valid_state_labels" list.

## OUTPUT FORMAT ##
The length and turn order of the output list must match the length and turn order of the input list. The sample output format is given as follow: [{
"T-$turn number$": {
"summary": "$turn summary in ≤ 3 sentence$",
"preceding_topical_relation": "$an appropriate valid preceding topical relation label$",
"domain": "$an appropriate valid domain label$",
"intent": "INTENT:$an appropriate valid intent label$",
"satisfaction": [$a comma separated string list of applicable valid satisfaction label(s)$],
"dissatisfaction": [$a comma separated string list of applicable valid dissatisfaction label(s)$],
"state": "$an appropriate valid state label$"
}
}]

## INPUT ##
#D1#

## OUTPUT ##

A.2 Prompt for Preference Data Construction

The following is the prompt for constructing preference data.

# Conversation between User and AI
< |begin_of_history| >
history
< |end_of_history| >
# Instruction
What are the user's query and preferences? The query should be the user's first attempt before providing any feedbacks to the model. Only output the turn id. The preference should always be based on user's feedbacks and in complete sentences. Generate your answer in json format like

```
[ {
"query": turn id,
"preferences": [preference 1, preference 2, ...]
} ]
```

A.3 Prompt for User-guided Evaluation

The following is the prompt for user-guided evaluation. We borrow the WB-Reward prompt from WILDBENCH (Lin et al., 2024a).

# Instruction
You are an expert evaluator. Your task is to evaluate the quality of the responses generated by two AI models. We will provide you with the user query and a pair of AI-generated responses (Response A and B). You should first read the user query and the conversation history carefully for analyzing the task, and then evaluate the quality of the responses based on and rules provided below.
# Conversation between User and AI
## History
< |begin_of_history| >
{history}
< |end_of_history| >
## Current User Query
< |begin_of_query| >
{query}

```
< |end_of_query| >
## Response A
< |begin_of_response_A| >
{response_a}
< |end_of_response_A| >
## Response B
< |begin_of_response_B| >
{response_b}
< |end_of_response_B| >
# Evaluation
## Checklist
< |begin_of_checklist| >
{checklist}
< |end_of_checklist| >
Please use this checklist to guide your evaluation, but do not limit your assessment to the checklist.
## Rules
You should compare the above two responses based on your analysis of the user queries and the conversation history. You should first write down your analysis and the checklist that you used for the evaluation, and then provide your assessment according to the checklist. There are five choices to give your final assessment: [“A++”, “A+”, “A=B”, “B+”, “B++”], which correspond to the following meanings:
- ‘A++’: Response A is much better than Response B.
- ‘A+’: Response A is only slightly better than Response B.
- ‘A=B’: Response A and B are of the same quality. Please use this choice sparingly.
- ‘B+’: Response B is only slightly better than Response A.
- ‘B++’: Response B is much better than Response A.
## Output Format
First, please output your analysis for each model response, and then summarize your assessment to three aspects: “reason A=B”, “reason A > B”, and “reason B > A”, and finally make your choice for the final assessment. Please provide your evaluation results in the following json format by filling in the placeholders in []:
{
"analysis of A":  "[analysis of Response A]",
"analysis of B":  "[analysis of Response B]",
"reason of A=B":  "[where Response A and B perform equally well]",
"reason of A>B":  "[where Response A is better than Response B]",
"reason of B>A":  "[where Response B is better than Response A]",
"choice":  "[A++ or A+ or A=B or B+ or B++]"
}
```

B SAT and DSAT

B.1 Detailed SAT and DSAT Criteria

The detailed definitions of SAT and DSAT can be found in Table 3 and Table 4.

B.2 SAT and DSAT Annotation

GPT-4’s performances on SAT and DSAT classification can be found in table 5. GPT-4 demonstrates strong performance in classifying SAT (satisfaction) signals, with high accuracy at 91.7% and balanced precision and recall, both around 73%. The Cohen’s Kappa of 68.5% reflects substantial agreement with human annotators. For DSAT (dissatisfaction) signals, GPT-4 achieves a precision of 83.3%, with a recall of 48.4%, leading to an F1 score of 61.2% and a Cohen’s Kappa of 50.4%. These metrics indicate that GPT-4 is effective at recognizing both SAT and DSAT signals. For human annotation, we utilized a web-based annotation tool named Potato (Pei et al., 2022). The interface is shown in Figure 3.

Table 3: Detailed definitions of the SAT Rubrics.

Keyword	**Definition**
Gratitude	The user thanks or compliments the AI agent for its responses.
Learning	The user learns something new or useful by indicating curiosity and satisfaction with the information provided.
Compliance	The user follows the AI agent's suggestions or instructions when applicable.
Praise	The user uses positive feedback words (e.g., excellent, amazing) or emojis, indicating enthusiasm and enjoyment of the conversation.
Personal Details	The user shares more personal details or opinions with the AI agent when satisfied with its responses.
Humor	The user jokes with or challenges the AI agent in a friendly manner when suitable.
Acknowledgment	The user acknowledges or confirms that they understood or agreed with the AI agent's explanations when relevant.
Positive Closure	The user ends the conversation on a positive note without asking for more information or assistance.
Getting There	The user acknowledges that the model's response is getting better or has merit but is not fully satisfied.

C WildFeedback Data

To demonstrate that the generated preferred responses align with actual user preferences, we randomly selected 500 samples from the WildFeedback datasets and performed user-guided evaluation, comparing the preferred and dispreferred responses. As explained in Section §1, there are two versions of WildFeedback: the GPT-4 version and the on-policy version, which differ in whether the responses are generated by GPT-4 or the policy model. As shown in Figure 4, we found that without providing the summarized user preferences as checklists, GPT-4 tends to prefer the dispreferred responses in our dataset, which are the model's zero-shot generations without guidance from summarized user preferences. However, after providing the preferences as checklists to guide the evaluation, GPT-4's selections more closely align with real users' preferences. Additionally, we observed that GPT-4 is significantly more steerable than smaller models: over 70% of its preferred responses align with in-situ user preferences, compared to only about 50% for smaller models. Consequently, for on-policy data, we additionally filter out any data that does not align with user preferences.

We also compare WildFeedback with current open-source datasets in Table 6 [2]. To the best of our knowledge, WildFeedback is the first multi-turn pairwise preference dataset constructed from real human-LLM interactions. It is also the only dataset derived from in-situ user feedback, unlike existing preference datasets that are annotated by human annotators or LLMs, which often fail to fully capture real users' preferences. Additionally, although OpenAssistant Conversations (OASST1) (Köpf et al., 2023) also include multi-turn conversations, both its prompts and responses are entirely composed by human annotators, making it less reflective of the genuine dynamics of human-LLM interactions. Overall, WildFeedback outperforms existing datasets in accurately representing authentic human-LLM interactions, making it a more reliable resource for developing and evaluating preference-based models.

[2] For UltraFeedback, we refer to the pre-processed, binarized version that was used to train Zephyr (Tunstall et al., 2023).

Table 4: Detailed definitions of the DSAT Rubrics.

Keyword	Definition
Negative Feedback	The user explicitly expresses dissatisfaction, frustration, annoyance, or anger with the AI agent's response or behavior.
Revision	The user explicitly asks the AI agent to revise its previous response or repeatedly asks similar questions.
Factual Error	The user points out the AI agent's factual mistakes, inaccuracies, or self-contradiction in its information or output.
Unrealistic Expectation	The user has unrealistic expectations of what the AI agent can do and does not accept its limitations or alternatives.
No Engagement	The user does not respond to the AI agent's questions, suggestions, feedback requests, etc.
Ignored	The user implies that their query was ignored completely or that the response did not address their intent/goal at all.
Lower Quality	The user perceives a decline in quality of service compared to previous experience with other agents/tools, etc.
Insufficient Detail	The user wants more specific/useful information than what is provided by the AI agent.
Style	The user feels that there is a mismatch between their preferred style and what is provided by the AI agent.

Table 5: SAT and DSAT Classification Results. All numbers are in %.

	Accuracy	Precision	Recall	F1	GPT-Human κ	Human-Human κ
SAT	91.7	73.2	73.6	73.4	68.5	70.0
DSAT	81.8	83.3	48.4	61.2	50.4	54.1

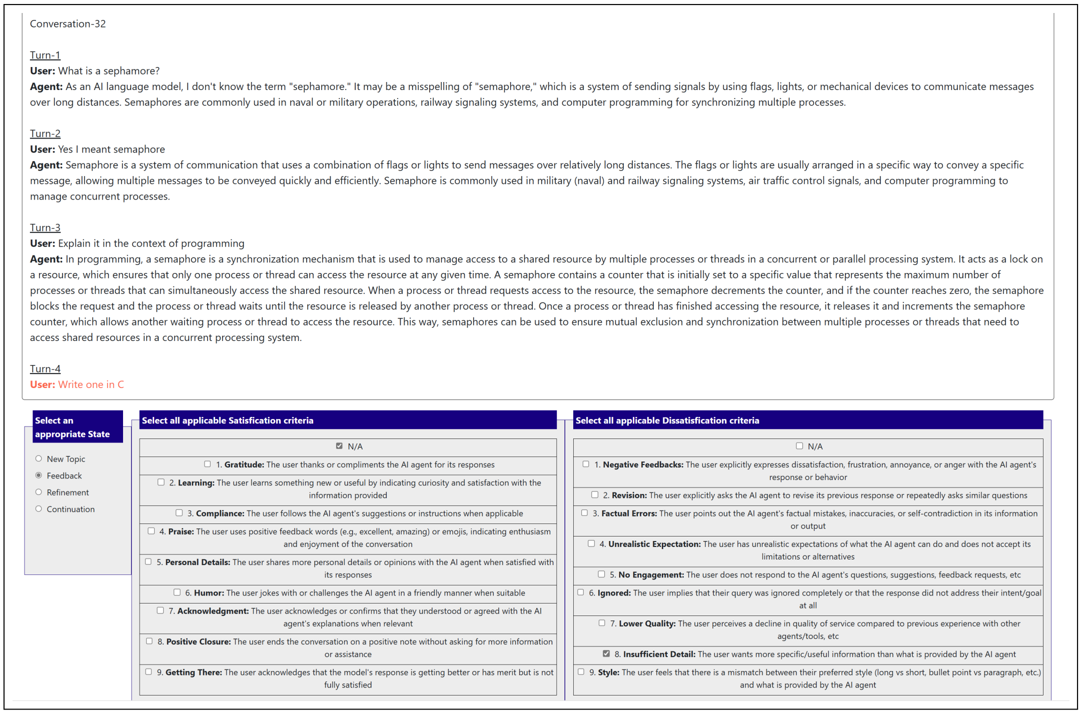
D Implementation Details

Unless otherwise specified, in all of our experiments, we use GPT-4o with the `gpt-4o-0513` engine. For open-weight models, we use `Phi-3-mini-4k-instruct, Mistral-7B-Instruct-v0.3, Meta-Llama-3-8B-Instruct`.

Additionally, we found that hyperparameter tuning is crucial for achieving optimal performance in preference optimization. Generally, on-policy data requires a lower learning rate than GPT-4o data,

Table 6: Statistics of existing preference datasets. The average length refers to the number of tokens. The responses of WILDFEEDBACK are either extracted from the original conversations or generated by GPT-4, Mistral, Phi 3, or LLaMA 3.

	# Conv.	Prompt Length	Response Length	Multi-Turn?	Feedback Type
WebGPT (Nakano et al., 2022)	38,925	51	188	✗	Human Annotators
Anthropic HH (Bai et al., 2022)	118,263	186	95	✗	Human Annotators
OASST1 (Köpf et al., 2023)	35,905	168	221	✓	Human Annotators
ULTRAFEEDBACK (Cui et al., 2024)	61,135	159	256	✗	GPT-4
WILDFEEDBACK (ours)				✓	In-situ Users
↪ GPT-4	20,281	929	440		
↪ Mistral	9,601	1,063	362		
↪ Phi 3	9,194	931	344		
↪ LLaMA 3	10,659	982	376		



Conversation-32

Turn-1
User: What is a sephamore?
Agent: As an AI language model, I don't know the term "sephamore." It may be a misspelling of "semaphore," which is a system of sending signals by using flags, lights, or mechanical devices to communicate messages over long distances. Semaphores are commonly used in naval or military operations, railway signaling systems, and computer programming for synchronizing multiple processes.

Turn-2
User: Yes I meant semaphore
Agent: Semaphore is a system of communication that uses a combination of flags or lights to send messages over relatively long distances. The flags or lights are usually arranged in a specific way to convey a specific message, allowing multiple messages to be conveyed quickly and efficiently. Semaphore is commonly used in military (naval) and railway signaling systems, air traffic control signals, and computer programming to manage concurrent processes.

Turn-3
User: Explain it in the context of programming
Agent: In programming, a semaphore is a synchronization mechanism that is used to manage access to a shared resource by multiple processes or threads in a concurrent or parallel processing system. It acts as a lock on a resource, which ensures that only one process or thread can access the resource at any given time. A semaphore contains a counter that is initially set to a specific value that represents the maximum number of processes or threads that can simultaneously access the shared resource. When a process or thread requests access to the resource, the semaphore decrements the counter, and if the counter reaches zero, the semaphore blocks the request and the process or thread waits until the resource is released by another process or thread. Once a process or thread has finished accessing the resource, it releases it and increments the semaphore counter, which allows another waiting process or thread to access the resource. This way, semaphores can be used to ensure mutual exclusion and synchronization between multiple processes or threads that need to access shared resources in a concurrent processing system.

Turn-4
User: Write one in C

Select an appropriate State

○ New Topic
● Feedback
○ Refinement
○ Continuation

Select all applicable Satisfication criteria

☑ N/A
☐ 1. **Gratitude:** The user thanks or compliments the AI agent for its responses
☐ 2. **Learning:** The user learns something new or useful by indicating curiosity and satisfaction with the information provided
☐ 3. **Compliance:** The user follows the AI agent's suggestions or instructions when applicable
☐ 4. **Praise:** The user uses positive feedback words (e.g., excellent, amazing) or emojis, indicating enthusiasm and enjoyment of the conversation
☐ 5. **Personal Details:** The user shares more personal details or opinions with the AI agent when satisfied with its responses
☐ 6. **Humor:** The user jokes with or challenges the AI agent in a friendly manner when suitable
☐ 7. **Acknowledgment:** The user acknowledges or confirms that they understood or agreed with the AI agent's explanations when relevant
☐ 8. **Positive Closure:** The user ends the conversation on a positive note without asking for more information or assistance
☐ 9. **Getting There:** The user acknowledges that the model's response is getting better or has merit but is not fully satisfied

Select all applicable Dissatisfication criteria

☐ N/A
☐ 1. **Negative Feedbacks:** The user explicitly expresses dissatisfaction, frustration, annoyance, or anger with the AI agent's response or behavior
☐ 2. **Revision:** The user explicitly asks the AI agent to revise its previous response or repeatedly asks similar questions
☐ 3. **Factual Errors:** The user points out the AI agent's factual mistakes, inaccuracies, or self-contradiction in its information or output
☐ 4. **Unrealistic Expectation:** The user has unrealistic expectations of what the AI agent can do and does not accept its limitations or alternatives
☐ 5. **No Engagement:** The user does not respond to the AI agent's questions, suggestions, feedback requests, etc
☐ 6. **Ignored:** The user implies that their query was ignored completely or that the response did not address their intent/goal at all
☐ 7. **Lower Quality:** The user perceives a decline in quality of service compared to previous experience with other agents/tools, etc
☑ 8. **Insufficient Detail:** The user wants more specific/useful information than what is provided by the AI agent
☐ 9. **Style:** The user feels that there is a mismatch between their preferred style (long vs short, bullet point vs paragraph, etc.) and what is provided by the AI agent

Figure 3: The interface used for human annotation.

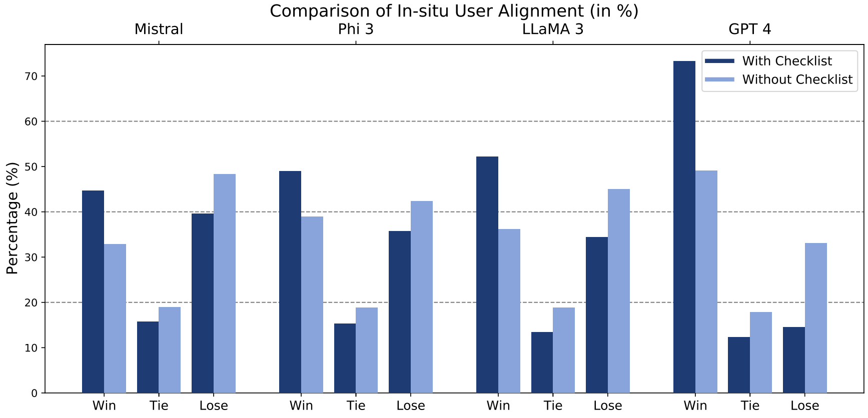


Figure 4: Comparison of in-situ user alignment across datasets generated by different models. "Win/Tie/Lose" represents the percentage of instances where the preferred responses win/tie/lose compared to the dispreferred responses in the WILDFEEDBACK dataset, prior to filtering. The comparison is made both with and without providing GPT-4 with summarized user preferences as checklists to guide its evaluation.

and instruct models need a lower learning rate than base models. Specifically, Mistral and Gemma (Team et al., 2024) require a lower learning rate than Phi 3 and LLaMA 3. Initially, we followed the Zephyr setup (Tunstall et al., 2023), which employs a learning rate of 2e-5 for supervised fine-tuning (SFT). However, we found that our models quickly collapsed, failing to generate sensible outputs after just a few dozen iterations. After conducting a grid search on the hyperparameters for both SFT and DPO training, we discovered that while it is acceptable to use a larger learning rate for training base models, a much smaller learning rate is required for instruct models, likely due to the various annealing techniques applied during the post-training process (Parmar et al., 2024). We also explored NLL regularization (Liu et al., 2024a) with a regularization strength of 0.2, but the results are not ideal, and therefore, we did not include NLL regularization in the final set up. We trained all the models using LLaMA Factory (Zheng et al., 2024), a unified efficient LLM finetuning framework. The following is the hyperparameters we used in our final experiment.

SFT Training. For SFT training, we trained all the models for 1 epoch with a batch size of 128, a learning rate of 5e-6, a linear warm-up ratio of 0.1, and a cosine learning rate scheduler. Better results may be achievable by decreasing the learning rate for Mistral. Additionally, it is recommended to use a higher learning rate (e.g., 2e-5) if you are fine-tuning from the base models.

DPO Training. For DPO training, we trained all the models for 1 epoch with a batch size of 32, a learning rate of 5e-7, and $\beta = 0.1$. All other hyperparameters remained the same as in the SFT training.

E Evaluation

Benchmarks Evaluation. We evaluate our models using three of the most popular open-ended instruction-following benchmarks: MT-Bench (Zheng et al., 2023a), AlpacaEval 2 (Li et al., 2023), and Arena-Hard (Li et al., 2024). AlpacaEval 2 consists of 805 questions from 5 datasets, and MT-Bench covers 8 categories with 80 questions. Arena-Hard is an enhanced version of MT-Bench, incorporating 500 well-defined technical problem-solving queries. We report scores following each benchmark's evaluation protocol. For AlpacaEval 2, we report both the raw win rate (WR) and the length-controlled win rate (Dubois et al., 2024). The LC metric is specifically designed to be robust against model verbosity. For Arena-Hard, we report the win rate (WR) against the baseline model. We use GPT-4-Turbo (`gpt-4-0125`) as the judge for both AlpacaEval 2 and Arena-Hard. For MT-Bench, we report the average MT-Bench score with GPT-4o (`gpt-4o-0513`) as the judge. We use the same, default decoding strategies specified by the evaluation benchmarks.

WILDFEEDBACK Evaluation. In addition to publicly available benchmarks, we also constructed our own evaluation benchmark from the held-out test set in WILDFEEDBACK and evaluated models using user-guided evaluation. We ensured that all samples in the test set were sourced from conversations and users that were never included in the training set. Constructing an evaluation dataset for user-guided evaluation is not a trivial task, as we can no longer randomly or stratifiedly select test samples from different domains. In user-guided evaluation, we always provide a user-inspired checklist for GPT-4 to guide its evaluation, making it more aligned with real users' preferences. However, individual user preferences can be highly subjective and specific. The goal of WILDFEEDBACK is not to align language models with the preferences of a specific individual but to learn the broader mode of all individuals' preferences. Therefore, we must ensure that the preferences reflected in the test samples represent the majority view. Additionally, since the user preferences we extracted are often particular to specific tasks, we also need to ensure that the tasks in the test set are at least somewhat similar to those in the training set.

To achieve this, we utilized FAISS (Douze et al., 2024) to cluster user prompts and their summarized preferences. We grouped all user prompts into 70 clusters. Within each cluster, we selected 10 samples where the preferences were most similar to the other preferences in the same group. We then applied similar data curation techniques as described in WILDBENCH (Lin et al., 2024a) to perform deduplication and remove nonsensical tasks, resulting in a final test set of 540 samples. This approach ensures that the evaluation set captures a representative range of user preferences, while also maintaining diversity within the clusters. By doing so, we aim to provide a more reliable and comprehensive evaluation that reflects the majority's preferences without overfitting to specific, idiosyncratic cases. This method allows us to test the model's ability to generalize across a broad spectrum of user needs, ultimately leading to a more robust and user-aligned language model.

For WILDFEEDBACK evaluation, we report the win, tie, lose percentage against the off-the-shelf instruct models with GPT-4 as the judge. We employ the WILDBENCH prompt Lin et al. (2024a) to perform the evaluation, which has been shown to correlate well with human judgement in ranking model performance. We report the results evaluated with or without the user preferences provided as a checklist to guide GPT-4o evaluation.