Aligning Large Language Models with Implicit Preferences from User-Generated Content

Anonymous ACL submission

Abstract

Learning from preference feedback is essential for aligning large language models (LLMs) with human values and improving the quality of generated responses. However, existing preference learning methods rely heavily on curated data from humans or advanced LLMs, which is costly and difficult to scale. In this work, we present PUGC, a novel framework that leverages implicit human **P**references in unlabeled **U**ser-**G**enerated **C**ontent (UGC) to generate preference data. Although UGC is not explicitly created to guide LLMs in generating human-preferred responses, it often reflects valuable insights and implicit preferences from its creators that has the potential to address readers' questions. PUGC transforms UGC into user queries and generates responses from the policy model. The UGC is then leveraged as a reference text for response scoring, aligning the model with these implicit preferences. This approach improves the quality of preference data while enabling scalable, domain-specific alignment. Experimental results on Alpaca Eval 2 show that models trained with DPO and PUGC achieve a 9.37% performance improvement over traditional methods, setting a 35.93% state-of-the-art length-controlled win rate using Mistral-7B-Instruct. Further studies highlight gains in reward quality, domain-specific alignment effectiveness, robustness against UGC quality, and theory of mind capabilities.

1 Introduction

Advancements in large language models (LLMs) have enhanced their ability to follow instructions and respond to user queries (Wei et al., 2021; Dubey et al., 2024; Chung et al., 2024), primarily through alignment with human values to ensure they are helpful, honest, and harmless (Askell et al., 2021). Reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022) achieves this alignment by using a reward model that captures human preferences to guide policy models (Schulman et al., 2017), but its multi-stage process is complex, resource-intensive, and prone to instability (Casper et al., 2023). Direct Preference Optimization (DPO) (Rafailov et al., 2024) simplifies alignment by directly training policy models from preference data without an explicit reward model, leading to widespread adoption due to its simplicity and stability (Meng et al., 2024). However, both RLHF and DPO rely on high-quality preference data, which is expensive and hard to scale as it typically requires human annotators (Köpf et al., 2024) or stronger models like GPT-4 (Cui et al., 2023), and inconsistent data quality further limits alignment performance (Zhou et al., 2024a; Wu et al., 2024; Liu et al., 2023b; Zheng et al., 2023b).

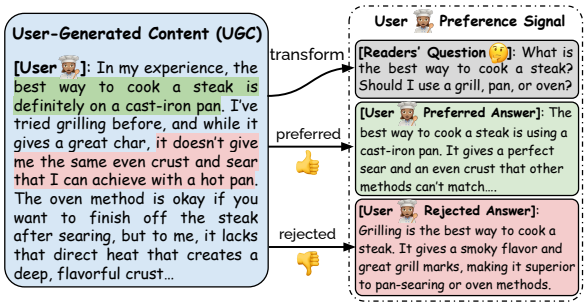

Figure 1: User-Generated Content (UGC) is created to share users' knowledge and insights with others, often addressing readers' questions. This creates the potential to transform UGC into reader queries and sharer-preferred answers, which can be utilized to align LLMs with user preferences.

Among the potential solutions to align LLMs with human preference without explicit human and powerful model annotation, leveraging large amounts of unlabeled data to align LLMs with human preference emerges as a promising approach. Existing methods focus on sourcing supervised fine-tuning (SFT) data from unlabeled text. HumpBack (Li et al., 2023a) translates web corpus data into instructions and fine-tunes models using high-quality, self-curated instructional data.

Back-and-forth translation (Nguyen et al., 2024) takes this further by applying additional filtering and rewriting steps to enhance instruction and response quality. However, research shows that SFT provides a biased estimate of human preferences by relying on predefined input-output pairs. In contrast, RLHF helps further align the model with human preferences by dynamically updating its responses based on feedback (Kirk et al., 2023; Dong et al., 2024). The research question arises: *Can we collect preference data for LLM response distantly from the massive unlabeled data?*

Among the vast amounts of textual data available, user-generated content (UGC)—including tweets, reviews, and blogs that contain user's insight and knowledge—serves as a valuable source for aligning LLMs with human preferences. UGC is abundant on platforms like Amazon reviews, Reddit, and StackExchange, making it a scalable resource. Although UGC is not explicitly created to annotate user preferences, it often contains the sharer's knowledge and insights, providing potential answers to readers' questions. These insights can be transformed into reader queries, capturing the sharer's preferences in their responses (Figure 1). By converting UGC into reader queries and sharer preferences, we can extract valuable supervision for human preference alignment, thereby constructing high-quality, large-scale preference data to improve LLMs.

Leveraging the implicit preferences in UGC, we propose PUGC, a novel framework to generate high-quality preference data from UGC for aligning LLMs. First, PUGC employs an LLM to extract potential reader queries from UGC. We then use the same LLM to verify that the UGC contains sufficient information relevant to the generated instructions, filtering out irrelevant content. Next, we feed these instructions into a policy model and sample multiple responses. PUGC evaluates these responses using a reward model, with the original UGC serving as a reference to uncover implicit user preferences. Thus, PUGC efficiently leverages unlabeled UGC for auxiliary supervision, enhancing reward quality compared to traditional preference data generation paradigms and enabling more flexible alignment across domains without collecting human instructions.

Our experiments show that using 60k UGC from Dolma dataset (Soldaini et al., 2024), PUGC can obtain a model that achieves the state-of-the-art length-controlled win rate of 35.93% based on Mistral-7B-Instruct-v0.2 on AlpacaEval 2.0 benchmark. PUGC supports different preference-tuning methods such as DPO (Rafailov et al., 2024) and SimPO (Meng et al., 2024), and showing consistent superiority compared to preference data derived from UltraFeedback (Cui et al., 2023) instructions. The performance advantage is consistent across various training setups and diverse instruction-following benchmarks, including AlpacaEval 2 (Li et al., 2023b) and MT-Bench (Zheng et al., 2023a). We also observe a significant performance gain in the theory of mind capabilities demonstrated by BigGen Bench (Kim et al., 2024b). Further studies also reveal PUGC's improved reward quality, robustness against UGC quality, comprehensive performance gain in fine-grained Alpaca Eval analysis, and its effectiveness and flexibility in domain-specific alignment.

In summary, PUGC is the first to harness implicit user preferences from UGC to cost-effectively construct high-quality preference data. This approach enables flexible, domain-specific alignment and offers significant advantages over traditional preference data generation methods. By introducing this novel pipeline, PUGC opens new avenues for utilizing unlabeled text in human preference alignment.

2 Preference Alignment using UGC (PUGC)

In this section, we first provide the background of current preference data generation pipeline in §2.1. We then introduce our proposed PUGC pipeline (§2.2), which leverages implicit user preference signals from UGC, highlighting how it differs from traditional preference data generation approach.

2.1 Traditional Preference Data Generation

In traditional preference data generation pipelines for alignment, a set of prompts $\mathcal{X}$ is collected, and multiple responses are sampled for each prompt $x \in \mathcal{X}$ from the policy model π_θ.

$$\{a_i\}_{i=1}^{K} \sim \pi_\theta(a \mid x),$$

where a denotes the generated response and K denotes the number of generated responses for each prompt. Once the responses are sampled, a reward model *RM* scores each response a_i for prompt x:

$$y_i = \mathit{RM}(a_i \mid x),$$

where y_i is the reward score corresponding to response a_i. After scoring all the responses $\{a_i\}_{i=1}^K$, the responses with the highest and lowest scores are selected as the preferred response a_w and rejected response a_l, respectively. This forms the preference data pair $(x, a_w > a_l)$ which is used for preference tuning.

2.2 Generating Preference Data from UGC

Motivated by the fact that UGC is often created to address potential questions from other users, PUGC begins by transforming UGC into a reader's question using an LLM (SFT version model by default). This transformation can be expressed as:

$$x = \textit{GenInst}((\phi_q(d)) \mid \pi_\theta),$$

where ϕ_q denotes the prompt construction function that converts the UGC d into a reader instruction x. To ensure the quality of the generated instructions, we filter out those that are irrelevant to the source UGC. Specifically, we utilize the same LLM for instruction generation π_θ to assess whether the UGC contains sufficient information to answer the generated instruction. This evaluation produces a filtering score $s \in \{\textit{True}, \textit{False}\}$ for each instruction-UGC pair.

$$s_x = \textit{Filter}(\phi_f(x, d) \mid \pi_\theta),$$

where ϕ_f is the filtering prompt template. We then only keep the instruction and UGC that corresponding s_x is *True* to ensure the instruction quality, the kept instructions are denoted as $x' \in \mathcal{X}'$. To leverage the sharers' insights and knowledge and implicit preference signals within the UGC, we then feed the UGC as a reference answer in the reward model for response scoring, mining the implicit preference signals in the UGC, where the reward score for the j-the response is

$$y_j = \textit{RM}(a_i \mid x', d).$$

Finally, using the instruction x', the sampled responses $\{a_j\}$, and the corresponding reward scores $\{y_j\}$, we can obtain the preference data pairs $\mathcal{P}$ for the entire instruction set $\mathcal{P} = \{(\overline{x}, y_w > y_l)\}, \overline{x} \in \mathcal{X}'$, where y^w is the chosen response, with $w = \arg\max_j(\{y_j\}_{j=1}^K)$, and y_l is the rejected response, with $l = \arg\min_j(\{y_j\}_{j=1}^K)$.

PUGC is a versatile preference data generation framework that leverages implicit user preference signals from unlabeled UGC, requiring only UGC instead of prompt instructions. Since domain-specific UGC is easier to collect compared to user instructions, PUGC offers a more flexible framework for domain-specific alignment. Moreover, PUGC is compatible with various preference tuning methods, such as DPO (Rafailov et al., 2024), SimPO (Meng et al., 2024), and others.

3 Experiment Settings

Datasets For the UGC data, we select the Dolma (Soldaini et al., 2024) pretraining corpus. Specifically, we sample data from *cc_en_head*, *megawika*, *stackexchange*, *tulu_flan*, *open_web_math*, and *starcoder* as the text corpus, and use Llama-3-70B-instruct (Dubey et al., 2024) via Bedrock API to generate a 5-point quality score for the unlabeled text. To balance data quantity and quality, we select UGC with scores greater than or equal to 4 and feed them into the UGC preference data generation pipeline.

Models and Training Settings We perform preference tuning with Mistral-7B (Jiang et al., 2023) model under the base and instruct setup following Meng et al. (2024). For the base setup, we use `alignment-handbook/zephyr-7b-sft-full` (Tunstall et al., 2023), trained on the UltraChat-200k dataset (Ding et al., 2023), which provides a high level of transparency. For the instruct setup, we use the off-the-shelf instruction-tuned model `mistralai/Mistral-7B-Instruct-v0.2`, which is more robust but less transparent as the RLHF process is not publicly disclosed. For response sampling from the policy model, we follow setting in SimPO and set the temperature to 0.8 and use nucleus sampling with p=0.95. The reward model used is `prometheus-eval/prometheus-7b-v2.0` (Kim et al., 2024c), chosen for its strong judgement correlation with GPT and human, and the ability to directly score responses using reference answers. We adopt self-consistency decoding with N=8 for reward score generation, as suggested by Kim et al. (2024b), and resolve ties by selecting the shortest high-scoring response as the preferred answer and the longest low-scoring response as the rejected answer to mitigate length bias.

Evaluation Benchmark and Evaluation Metrics We evaluate our method on two widely used open-ended generation quality benchmarks: MT-Bench (Zheng et al., 2023a) and AlpacaEval 2 (Li et al., 2023b). AlpacaEval 2 includes 805 questions

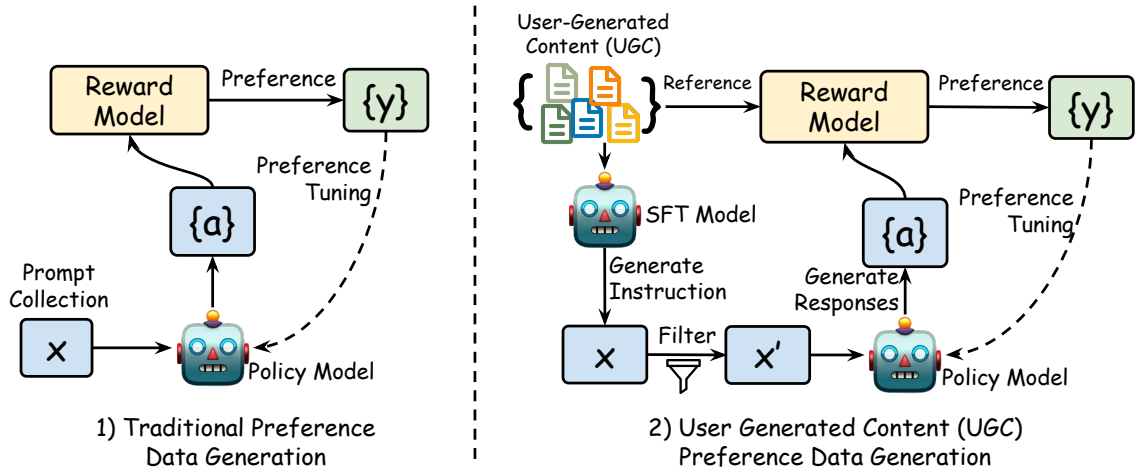

Figure 2: Overview of PUGC. Unlike traditional preference data generation methods, PUGC sources preference data from user-generated content (UGC) by transforming UGC into reader questions and using UGC as a reference for preferred answers. PUGC leverages implicit user preference signals to align LLMs.

from five datasets, using GPT-4 Turbo as a judge for pairwise comparisons between the evaluated model's generation and GPT-4 Turbo's output. We report both raw win rate and length-controlled win rate (Dubois et al., 2024) to mitigate verbosity bias. MT-Bench consists of 80 multi-turn questions across 8 categories, and we report the average MT score using GPT-4 as the judge.[1]

Baselines We compare PUGC against offline preference optimization methods using preference data generated via the traditional pipeline with 60k prompts from UltraFeedback (Cui et al., 2023), ensuring a fair comparison. Preference optimization methods include ORPO (Hong et al., 2024), KTO (Ethayarajh et al., 2024), DPO (Rafailov et al., 2024), R-DPO (Park et al., 2024), CPO (Xu et al., 2024), and SimPO (Meng et al., 2024). All baselines employ the same reward model, `prometheus-eval/prometheus-7b-v2.0`, with identical settings to those of our proposed PUGC, but without reference text for reward scoring to maintain fairness in the comparison.

4 Results

Table 1 shows the performance of PUGC and baselines on Alpaca Eval 2.0 and MT-Bench, we have the following observations.

PUGC significantly outperform baseline Models trained with PUGC outperform baseline methods that use UltraFeedback preference data. Notably, when using the DPO objective, PUGC achieves an absolute improvement in LC win rate of 2.0% and 9.4% on the Alpaca Eval 2.0 benchmark under the base and instruct settings, respectively. Similarly, models trained with SimPO show improvements of 2.1% and 3.0% in both settings. On MT-Bench, PUGC also demonstrates superiority over models trained with UltraFeedback data and the SFT baseline. The limited performance gains or occasional degradation could be attributed to the limited judging capability of the Prometheus-2-7B model in reasoning, coding, and math domains, and the limited UGC in related domains. Additionally, the small evaluation dataset and single-instance assessments may impact the quality of MT-Bench results.

PUGC shows moderate length exploitation While PUGC achieves significant improvements, moderate length exploitation is observed in both the base and instruct setups. Models trained with SimPO and DPO generate responses that are relatively shorter than those from the baselines, yet they still achieve higher win rates. This suggests that PUGC's preference data generation pipeline can better mitigate length bias.

PUGC+DPO generally shows the best performance Trained on UltraFeedback data, DPO and SimPO generally perform the best across both settings, which justifies the choice of applying DPO and SimPO to PUGC preference data. Although SimPO outperforms DPO in the Instruct setting, PUGC+DPO achieves a higher LC win rate compared to PUGC+SimPO in both settings.

[1]Decoding hyperparameters are provided in Appendix F for reproducibility.

Table 1: Alpaca Eval 2.0 and MT-bench results. LC and WR denotes length-controlled and raw win rate, respectively. We use the off-the-shelf model as the SFT model. We use GPT-4-Turbo as a judge for MT-Bench scoring. The best score is in **bold** and the second best is underlined.

Method	Mistral-Base (7B)				Mistral-Instruct (7B)			
	Alpaca Eval 2.0			MT-Bench	Alpaca Eval 2.0			MT-Bench
	LC (%)	WR (%)	Len.	GPT-4-Turbo	LC (%)	WR (%)	Len.	GPT-4-Turbo
SFT	7.32	4.41	928	4.68	17.11	14.72	1676	6.20
Preference Data Generated with ULTRAFEEDBACK								
ORPO	8.18	4.66	867	5.04	20.85	16.75	1574	6.18
KTO	7.44	6.31	1367	4.92	22.45	20.07	1741	6.04
DPO	15.60	10.23	1118	5.04	26.56	24.39	1750	6.04
R-DPO	15.23	8.72	922	**5.14**	27.46	23.75	1658	6.06
CPO	9.17	5.50	932	4.75	21.34	19.40	1802	6.22
SIMPO	14.84	8.59	733	4.71	29.16	32.92	2338	5.66
Preference Data Generated with PUGC *(Ours)*								
PUGC+DPO	**17.57**	**11.30**	1013	5.13	**35.93**	**33.05**	1705	**6.27**
PUGC+SIMPO	16.95	9.10	700	5.09	32.15	30.34	1844	6.13

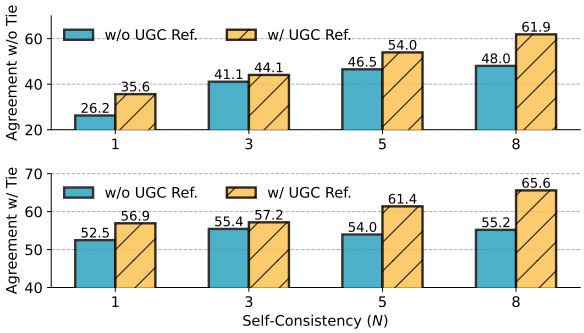

Figure 3: Reward judge agreement with GPT-4-Turbo on PUGC-sampled responses, under varying self-consistency settings. Reference UGC improves reward agreement, with higher self-consistency N leading to greater improvements.

This indicates that DPO, when paired with PUGC preference data, remains a highly effective strategy for aligning LLMs with human preferences.

The Instruct setting introduces significant performance gains Across both Alpaca Eval 2.0 and MT-Bench, we observe that Instruct setting consistently outperforms the Base setting. This improvement is likely due to the higher quality of SFT model used for instruction generation, as well as for preference tuning initialization and the generation of higher-quality preference by the Instruct setting models.

5 Analysis

UGC Reference Improves Reward Quality To validate the reward quality in PUGC generated data and the impact of UGC preference signals, we evaluate the judging accuracy on responses sampled from model-generated instructions and alter the number of N in self-consistency decoding. Specifically, we generate two responses using the policy model for each prompt, then generate preference labels with GPT-4-Turbo judge using the prompt in Alpaca Eval. To mitigate the position bias, we make two judgment with switched position for each response pair and retain the data where two judgements agree. This process yields a total of 202 pairs with preference judge labels. Subsequently, we use the reward model to generate reward score for those responses, under both with reference UGC and without UGC reference settings. We calculate two metrics: agreement (counting ties as 0.5) and agreement without ties (ignore ties). As is shown in Figure 3, judging responses with UGC as reference yields average 8.45% and 6% absolute improvement under with tie and without settings. Moreover, by increasing the number of sampling N in self-consistency decoding, the reward quality improves, while higher N can lead to greater improvements. These results demonstrate the effectiveness of the introducing reference UGC in harnessing the implicit preference signals, bringing its evaluations substantially closer to those of more sophisticated language models like GPT-4-Turbo.

UGC Quality *v.s.* Quantity To assess the impact of UGC quality and quantity in aligning LLMs with human preferences, we compare models trained on augmented data with varying levels of quality

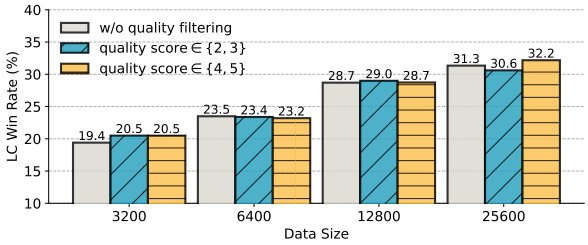

Figure 4: Alapaca Eval 2.0 performance of models trained with varying quality and quantity of preference data generated by PUGC. Increasing data quantity significantly improves performance, while PUGC remains robust to variations in UGC quality.

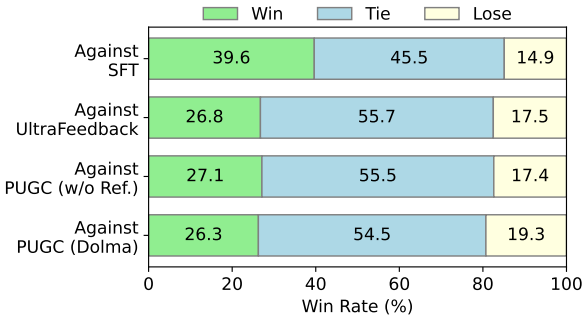

Figure 5: Domain-specific alignment performance of head-to-head win rate of PUGC trained with Goodreads book review against diverse baselines judged by GPT-4-Turbo. PUGC (Goodreads) has a 7% higher win rate against the strongest baseline.

and quantity. Specifically, we evaluate models trained on UGC without quality filtering, with quality scores in the range of $2, 3$, and with scores in the range of $4, 5$, across different data sizes. As shown in Figure 4, model performance improves significantly as data size increases, while UGC quality has a relatively minor effect on alignment performance. It demonstrates the robustness of PUGC to UGC quality variations, with larger data quantities yielding substantial performance gains.

Domain-Specific UGC Alignment As PUGC adapts to different domains using domain-specific UGC, we evaluate its effectiveness with user book review data from Goodreads[2] (Wan et al., 2019). For evaluation, we use the MT-Bench pairwise prompt with GPT-4-Turbo to compare PUGC (Goodreads) against the SFT baseline, the model trained on UltraFeedback preference data, and PUGC (general UGC from Dolma), and model trained on Goodreads review but without reference UGC in reward scoring. Evaluation prompts are generated by Claude-3-sonnet via Bedrock API based on book reviews to avoid distribution overlap with the SFT model. As shown in Figure 5, PUGC (Goodreads) consistently outperforms all four baselines, with a 7% higher win rate over PUGC (Dolma). These results demonstrate that PUGC is capable of generating high-quality, domain-specific preference data for more effective and flexible alignment.

Fine-Grained Alpaca Eval LC Win-Rate Analysis We analyze the Alpaca Eval test set to identify the task types contributing to the observed improvements by categorizing the set based on instruction complexity, task type, and expected response length. This grouping is achieved using prompts from Claude-3-Sonnet, with details and clustering statistics provided in Appendix J. The results in Figure 6 reveal the following insights: *i)* PUGC significantly improves performance on more complex instructions (complexity ≥ 4), with greater complexity leading to larger gains, likely due to the limited reward quality of the Prometheus model on simpler instructions. *ii)* PUGC substantially boosts LC win rates across most task categories, except for the math problems, programming, and summarization tasks, likely due to the lack of UGC in related domains and the distribution shift of instructions sourced from UGC. Compared to models trained with UltraFeedback data, PUGC shows notable improvements in categories like general knowledge, historical topics, review/letter writing, critique, and hypothetical scenarios, which are common in user-generated content. Moreover, the Prometheus-2 model was not trained on reasoning-intensive tasks, such as coding and math, which also limited the alignment performance. We leave developing a more domain comprehensive reward model that can accept the reference text as future work. *iii)* PUGC also shows significant LC win rate increases on tasks expecting outputs longer than one sentence, possibly due to length bias in UGC, as longer content tends to provide more insightful preference signals.

PUGC Significantly Improves Theory-of-Mind Capability As defined in BigGen Bench (Kim et al., 2024b), the theory of mind objective evaluates whether a language model can understand another individual's beliefs, intentions, and emotions through discourse, narratives, or dialogue. User-generated content (UGC) often reflects these aspects, providing rich theory of mind knowledge

[2] https://www.goodreads.com/

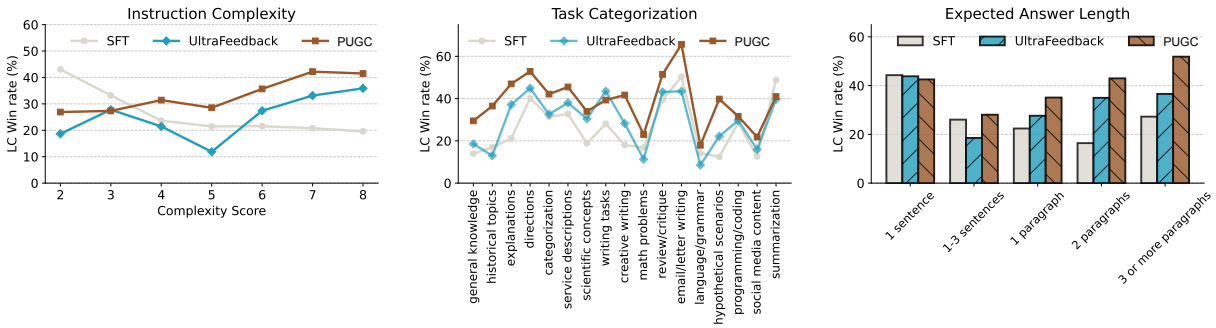

Figure 6: Fine-grained Alpaca Eval LC win rate by categorizing test instructions based on instruction complexity, task categorization, and expected answer length, showing particularly strong gains in handling more complex instructions and those requiring longer responses. We use DPO objective for preference tuning.

Table 2: Theory of Mind capability evaluated on BigGen Bench (Kim et al., 2024b), with scoring performed by GPT-4-Turbo, GPT-4, and Prometheus-bgb-8x7b-v2.0 as judge models.

Model	BigGen Bench (ToM)			
	GPT-4-Turbo	GPT-4	Prometheus2	Avg.
gpt-4-0125-preview	4.26	4.21	4.15	4.21
PUGC + DPO	4.22	4.10	4.17	4.16
gpt-4-1106-preview	4.24	4.07	4.09	4.13
gpt-4o-2024-05-13	4.08	4.04	4.06	4.07
gpt-4-turbo-2024-04-09	4.12	4.03	4.06	4.07
Claude-3-Opus	4.09	4.08	4.05	4.07
LLama-3-70B-Instruct	4.04	3.92	3.97	3.98
PUGC + SimPO	4.03	3.89	3.84	3.92
Mixtral-8x7B-Instruct-v0.1	3.81	3.80	3.78	3.80
Meta-Llama-3-8B-Instruct	3.83	3.65	3.71	3.73
Mistral-7B-Instruct-v0.2	3.77	3.68	3.49	3.65
Nous-Hermes-Mistral-7B-DPO	3.69	3.51	3.46	3.55

Table 3: Performance of PUGC across different ablated versions trained with the DPO objective. *Claude-3-Sonnet Gen. Inst.* refers to using Claude-3-Sonnet to generate and filter reader instructions for improved quality. These results validate the effectiveness of our design choices, highlighting PUGC's robustness to instruction quality and the crucial role of reward model selection.

Ablation Settings	Alpaca Eval 2.0		
	LC Win Rate	Win Rate	Avg. Len
full PUGC	35.93	33.05	1705
w/o reference UGC	31.12	30.72	2072
w/o instruction filtering	28.92	28.68	2254
w/ random reference UGC	16.74	16.33	2093
Claude-3-sonnet Gen. Inst.	35.71	34.00	1845
Skywork-Llama-8B as RM	16.12	17.08	2396
ArmoRM-Llama3-8B as RM	5.86	6.59	4402

that can enhance the preference data generated by PUGC. To validate this hypothesis, we evaluated models trained with PUGC's preference data against state-of-the-art LLMs on BigGen Bench, using GPT-4-Turbo, GPT-4, and the Prometheus-eval/prometheus-bgb-8x7b-v2.0 model as judge models, with the results shown in Table 2.

Our findings reveal that DPO trained with PUGC's preference data outperforms all open-source models and performs close to the GPT-4 series in terms of theory of mind capability. Additionally, when trained using the SimPO objective, PUGC achieves a 7.4% relative performance gain in this area. Compared to other models based on Mistral-7B, PUGC also demonstrates strong improvements in theory of mind capability. Overall, by leveraging preference data derived from UGC, PUGC effectively captures implicit user intentions, beliefs, and emotions, significantly enhancing the model's ability to understand these aspects and improving its theory of mind performance.

Ablation Study Given PUGC's strong performance in alignment benchmarks, we conduct an ablation study to assess the impact of key design choices. We specifically analyze the inclusion of reference UGC in response scoring and the filtering mechanism for generated instructions. As shown in Table 3, removing instruction filtering results in a larger performance drop, highlighting the importance of aligning instructions with relevant UGC. To investigate instruction quality, we utilize a strong proprietary model, Claude-3-Sonnet, for generating and filtering instructions based on UGC. Training with randomly selected UGC also results in a notable performance drop, highlighting the role of implicit user preferences. The model trained with Claude-3-Sonnet-generated instructions performs similarly to the default setup using SFT baseline-generated instructions, indicating that instruction quality has a limited impact on alignment performance, consistent with Ivison et al. (2024). For the reward model ablation,

we replace prometheus-7b-v2.0 with Skywork-Reward-Llama-3.1-8B and ArmoRM-Llama3-8B-v0.1, which shows the leading performance on RewardBench (Lambert et al., 2024). Despite strong results on RewardBench, Skywork and ArmoRM underperform within PUGC, where preference data generated with both models produce excessively long responses and low win rates. This highlights the need for a stronger reward model capable of utilizing reference text input, which is critical for enhancing the performance of PUGC.

6 Related Work

Reinforcement Learning from Human Feedback RLHF is a technique that aligns LLMs with human values (Christiano et al., 2017; Ziegler et al., 2019; Bai et al., 2022). The classical RLHF pipeline typically comprises three stages: supervised fine-tuning (Taori et al., 2023; Köpf et al., 2024; Wang et al., 2023a), reward model training (Gao et al., 2023; Lightman et al., 2023; Lambert et al., 2024), and policy model optimization (Schulman et al., 2017; Anthony et al., 2017). Proximal Policy Optimization (PPO) (Schulman et al., 2017) is a widely used algorithm for policy model optimization and is applied to mitigate toxicity (Korbak et al., 2023), ensure safety (Dai et al., 2023), enhance factuality (Tian et al., 2023) and reasoning capabilities (Havrilla et al., 2024). Given that online preference optimization algorithms are complex and difficult to optimize (Zheng et al., 2023b; Santacroce et al., 2023). Researchers proposed Direct Preference Optimization (DPO) (Rafailov et al., 2024) to avoid training the reward model, and instead directly train the LLM using human preferences. A variety of preference optimization objectives have been proposed besides DPO, including RRHF (Yuan et al., 2024a), SLiC-HF (Zhao et al., 2023), IPO (Azar et al., 2024), CPO (Xu et al., 2024), KTO (Ethayarajh et al., 2024), ORPO (Hong et al., 2024), R-DPO (Park et al., 2024), SimPO (Meng et al., 2024), and WPO (Zhou et al., 2024b).

Alignment Data Generation and Curation The alignment data can be categorized into supervised finetuning (SFT) data for instruction following and preference data for human value alignment. Alignment data can be broadly classified into two categories: supervised fine-tuning (SFT) data for instruction following and preference data for aligning models with human values. SFT data typically comes from human-annotated sources like Open Assistant (Köpf et al., 2024), Super-NI (Wang et al., 2022), FLAN (Longpre et al., 2023), and Natural Instructions (Mishra et al., 2021), though large-scale manual annotation is costly. To address this, researchers generate synthetic instructions (Wang et al., 2023b; Xu et al., 2023; Mitra et al., 2023), while BackHump (Li et al., 2023a) creates SFT data by back-translating unlabeled web corpora. Nguyen et al. (2024) improves response diversity and quality with rewriting techniques. For preference data, researchers have experimented with augmenting datasets using trained SFT policies (Zhao et al., 2023) or refined SFT policies coupled with rejection sampling (Liu et al., 2023a), allowing the policy to learn from data generated by optimal policies. Some studies reveal that offline DPO and PPO methods cannot strategically explore the environment (Xiong et al., 2023), emphasizing the value of suboptimal, on-policy data (Tajwar et al., 2024; Yu et al., 2024). As a result, preference optimization has been extended into iterative online settings, continuously updating reference models with the latest policy models (Kim et al., 2024a; Dong et al., 2024; Xiong et al., 2024; Yuan et al., 2024b). Sourced from user data, WildFeedBack (Shi et al., 2024) generates preference data from real-time, in-situ user interaction.

Building on insights from UGC that reflect human values and preferences (Tan and Jiang, 2023; Tan et al., 2024), we are the first to construct preference data from unlabeled UGC, offering a cost-effective method to generate high-quality data using implicit user preferences.

7 Conclusion

We introduced PUGC, a framework that leverages implicit user preferences from UGC to generate high-quality preference data for aligning LLMs with human values. By transforming UGC into reader queries and sharer-preferred responses, PUGC enhances preference alignment. It achieved a state-of-the-art 35.93% length-controlled win rate on Alpaca Eval 2.0 based on Mistral-7B-Instruct, and consistently outperformed models trained with traditional preference data. Additional studies demonstrated improvements in reward quality, UGC robustness, domain-specific flexibility, and theory of mind capabilities. Our proposed framework opens new opportunities for leveraging implicit human preferences from unlabeled text to align LLMs more effectively with human values.

Limitations

We identify two key limitations in PUGC. First, while it is designed to optimize LLM generation quality and align outputs with human preferences using UGC, UGC may contain toxic, harmful, or unsafe content. PUGC does not explicitly address safety or honesty, which are critical in real-world applications. Future work should focus on integrating safety and honesty considerations into the alignment process when leveraging UGC for preference learning. Second, we observe a performance drop in math, coding, and reasoning tasks. As shown in Tables 9 and 10, PUGC underperforms compared to the SFT model on GSM8k and Coding&Debugging tasks from WildBench (Lin et al., 2024). We hypothesize two possible reasons for this: *i)* the scarcity of high-quality UGC that provides strong preference signals in reasoning-intensive domains like coding and math, and *ii)* the limitations of the reward model, Prometheus-7b-v2.0, which was trained primarily to evaluate chat and generation quality, rather than reasoning accuracy (Kim et al., 2023). Developing a more robust reward model with better generalization across domains, particularly one that can handle reasoning tasks while utilizing reference text, remains an area for future exploration.

Ethical Considerations

Data Quality and Bias Leveraging UGC to align LLMs with human preferences introduces the risk of amplifying existing biases present in the data. UGC, often sourced from diverse and unmoderated platforms, may contain biased, harmful, or misleading content. If these biases are not identified and mitigated, they could be reflected in the model's outputs, potentially leading to unfair or prejudiced responses. Ensuring that the UGC used for preference alignment is representative, diverse, and free of harmful biases is crucial to prevent perpetuating negative effects. Future work should focus on bias detection and mitigation strategies when using UGC for training.

Safety and Misinformation UGC can sometimes include harmful, toxic, or incorrect information, which poses significant challenges when aligning LLMs with such content. Although PUGC enhances alignment by extracting implicit preferences from UGC, it does not explicitly account for the safety, accuracy, or truthfulness of the information being processed. This could inadvertently lead to the generation of unsafe or misleading responses. Future improvements should incorporate mechanisms to ensure that aligned models not only reflect human preferences but also prioritize safety, truthfulness, and the exclusion of harmful content.

Privacy and Consent UGC often originates from individuals who may not be aware that their content is being used to train AI models. This raises important privacy and consent issues, especially when personal or sensitive information is involved. Even when anonymized, user-generated content may carry implicit private information that could be exposed through model outputs. Ensuring that UGC is collected and used in a way that respects privacy and obtains appropriate consent from content creators is essential. Strong privacy safeguards should be implemented to protect users' data throughout the alignment process.

References

Thomas Anthony, Zheng Tian, and David Barber. 2017. Thinking fast and slow with deep learning and tree search. *Advances in neural information processing systems*, 30.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, et al. 2021. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. 2024. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep

reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

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

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

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. 2023. Safe rlhf: Safe reinforcement learning from human feedback. *arXiv preprint arXiv:2310.12773*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*.

Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. 2024. Rlhf workflow: From reward modeling to online rlhf. *arXiv preprint arXiv:2405.07863*.

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

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

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

Leo Gao, John Schulman, and Jacob Hilton. 2023. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pages 10835–10866. PMLR.

Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. 2020. Array programming with numpy. *Nature*, 585(7825):357–362.

Alex Havrilla, Yuqing Du, Sharath Chandra Raparthy, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, Sainbayar Sukhbaatar, and Roberta Raileanu. 2024. Teaching large language models to reason with reinforcement learning. *arXiv preprint arXiv:2403.04642*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Jiwoo Hong, Noah Lee, and James Thorne. 2024. Reference-free monolithic preference optimization with odds ratio. *arXiv preprint arXiv:2403.07691*.

Hamish Ivison, Yizhong Wang, Jiacheng Liu, Zeqiu Wu, Valentina Pyatkin, Nathan Lambert, Noah A Smith, Yejin Choi, and Hannaneh Hajishirzi. 2024. Unpacking dpo and ppo: Disentangling best practices for learning from preference feedback. *arXiv preprint arXiv:2406.09279*.

AQ Jiang, A Sablayrolles, A Mensch, C Bamford, DS Chaplot, D de las Casas, F Bressand, G Lengyel, G Lample, L Saulnier, et al. 2023. Mistral 7b (2023). *arXiv preprint arXiv:2310.06825*.

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

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

Seungone Kim, Juyoung Suk, Ji Yong Cho, Shayne Longpre, Chaeeun Kim, Dongkeun Yoon, Guijin Son, Yejin Cho, Sheikh Shafayat, Jinheon Baek, et al. 2024b. The biggen bench: A principled benchmark for fine-grained evaluation of language models with language models. *arXiv preprint arXiv:2406.05761*.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024c. Prometheus 2: An open source language model specialized in evaluating other language models. *arXiv preprint arXiv:2405.01535*.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. 2023. Understanding the effects of rlhf on llm generalisation and diversity. *arXiv preprint arXiv:2310.06452*.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, et al. 2024. Openassistant conversations-democratizing large language model alignment. *Advances in Neural Information Processing Systems*, 36.

Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Vinayak Bhalerao, Christopher Buckley, Jason Phang, Samuel R Bowman, and Ethan Perez. 2023. Pretraining language models with human preferences. In *International Conference on Machine Learning*, pages 17506–17533. PMLR.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. 2024. Rewardbench: Evaluating reward models for language modeling. *arXiv preprint arXiv:2403.13787*.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. 2024. From live data to high-quality benchmarks: The arena-hard pipeline.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Omer Levy, Luke Zettlemoyer, Jason Weston, and Mike Lewis. 2023a. Self-alignment with instruction backtranslation. *arXiv preprint arXiv:2308.06259*.

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

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. 2024. Wildbench: Benchmarking llms with challenging tasks from real users in the wild. *arXiv preprint arXiv:2406.04770*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Yong Lin, Lu Tan, Hangyu Lin, Zeming Zheng, Renjie Pi, Jipeng Zhang, Shizhe Diao, Haoxiang Wang, Han Zhao, Yuan Yao, et al. 2023. Speciality vs generality: An empirical study on catastrophic forgetting in fine-tuning foundation models. *arXiv preprint arXiv:2309.06256*.

Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. 2023a. Statistical rejection sampling improves preference optimization. *arXiv preprint arXiv:2309.06657*.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. 2023b. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. *arXiv preprint arXiv:2312.15685*.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*, pages 22631–22648. PMLR.

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

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2021. Cross-task generalization via natural language crowdsourcing instructions. *arXiv preprint arXiv:2104.08773*.

Arindam Mitra, Luciano Del Corro, Shweti Mahajan, Andres Codas, Clarisse Simoes, Sahaj Agarwal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, et al. 2023. Orca 2: Teaching small language models how to reason. *arXiv preprint arXiv:2311.11045*.

Thao Nguyen, Jeffrey Li, Sewoong Oh, Ludwig Schmidt, Jason Weston, Luke Zettlemoyer, and Xian Li. 2024. Better alignment with instruction back-and-forth translation. *arXiv preprint arXiv:2408.04614*.

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

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

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Michael Santacroce, Yadong Lu, Han Yu, Yuanzhi Li, and Yelong Shen. 2023. Efficient rlhf: Reducing the memory usage of ppo. *arXiv preprint arXiv:2309.00754*.

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

Taiwei Shi, Zhuoer Wang, Longqi Yang, Ying-Chun Lin, Zexue He, Mengting Wan, Pei Zhou, Sujay Jauhar, Xiaofeng Xu, Xia Song, et al. 2024. Wildfeedback: Aligning llms with in-situ user interactions and feedback. *arXiv preprint arXiv:2408.15549*.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Evan Walsh, Luke Zettlemoyer, Noah Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. 2024. Dolma: an open corpus of three trillion tokens for language model pretraining research. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15725–15788, Bangkok, Thailand. Association for Computational Linguistics.

Fahim Tajwar, Anikait Singh, Archit Sharma, Rafael Rafailov, Jeff Schneider, Tengyang Xie, Stefano Ermon, Chelsea Finn, and Aviral Kumar. 2024. Preference fine-tuning of llms should leverage suboptimal, on-policy data. *arXiv preprint arXiv:2404.14367*.

Zhaoxuan Tan and Meng Jiang. 2023. User modeling in the era of large language models: Current research and future directions. *arXiv preprint arXiv:2312.11518*.

Zhaoxuan Tan, Qingkai Zeng, Yijun Tian, Zheyuan Liu, Bing Yin, and Meng Jiang. 2024. Democratizing large language models via personalized parameter-efficient fine-tuning. *arXiv preprint arXiv:2402.04401*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Katherine Tian, Eric Mitchell, Huaxiu Yao, Christopher D Manning, and Chelsea Finn. 2023. Fine-tuning language models for factuality. *arXiv preprint arXiv:2311.08401*.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Shengyi Huang, Kashif Rasul, Alvaro Bartolome, Alexander M. Rush, and Thomas Wolf. The Alignment Handbook.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Mengting Wan, Rishabh Misra, Ndapa Nakashole, and Julian McAuley. 2019. Fine-grained spoiler detection from large-scale review corpora. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2605–2610, Florence, Italy. Association for Computational Linguistics.

Guan Wang, Sijie Cheng, Xianyuan Zhan, Xiangang Li, Sen Song, and Yang Liu. 2023a. Openchat: Advancing open-source language models with mixed-quality data. *arXiv preprint arXiv:2309.11235*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023b. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. *arXiv preprint arXiv:2204.07705*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason Weston, and Sainbayar Sukhbaatar. 2024. Meta-rewarding language models: Self-improving alignment with llm-as-a-meta-judge. *arXiv preprint arXiv:2407.19594*.

Wei Xiong, Hanze Dong, Chenlu Ye, Ziqi Wang, Han Zhong, Heng Ji, Nan Jiang, and Tong Zhang. 2024. Iterative preference learning from human feedback: Bridging theory and practice for rlhf under kl-constraint. In *Forty-first International Conference on Machine Learning*.

Wei Xiong, Hanze Dong, Chenlu Ye, Han Zhong, Nan Jiang, and Tong Zhang. 2023. Gibbs sampling from human feedback: A provable kl-constrained framework for rlhf. *arXiv preprint arXiv:2312.11456*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. Contrastive preference optimization: Pushing the boundaries of llm performance in machine translation. *arXiv preprint arXiv:2401.08417*.

Xiao Yu, Qingyang Wu, Yu Li, and Zhou Yu. 2024. Lions: An empirically optimized approach to align language models. *arXiv preprint arXiv:2407.06542*.

Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2024a. Rrhf: Rank responses to align language models with human feedback. *Advances in Neural Information Processing Systems*, 36.

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

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. 2023. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*.

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. 2023a. Judging llm-as-a-judge with mt-bench and chatbot arena. *Preprint*, arXiv:2306.05685.

Rui Zheng, Shihan Dou, Songyang Gao, Yuan Hua, Wei Shen, Binghai Wang, Yan Liu, Senjie Jin, Qin Liu, Yuhao Zhou, et al. 2023b. Secrets of rlhf in large language models part i: Ppo. *arXiv preprint arXiv:2307.04964*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2024a. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36.

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

Wenxuan Zhou, Ravi Agrawal, Shujian Zhang, Sathish Reddy Indurthi, Sanqiang Zhao, Kaiqiang Song, Silei Xu, and Chenguang Zhu. 2024b. Wpo: Enhancing rlhf with weighted preference optimization. *arXiv preprint arXiv:2406.11827*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

Table 4: PUGC's performance on Alpaca Eval 2.0 benchmark under online iterative training setting. We find the online iterative setting can further improve the model performance armed with PUGC.

Ablation Settings	Alpaca Eval 2.0		
	LC Win Rate	Win Rate	Avg. Len
SFT	17.11	14.72	1676
Offline (1 Iteration)	35.93	33.05	1705
PUGC+Iter1	30.85	28.62	1821
PUGC+Iter2	34.07	35.48	2200
PUGC+Iter3	37.51	42.31	2587

A Online Iterative Training can Further Improve the Performance

While our main experiments focus on the offline setting, where all UGC is used for a single iteration update, online iterative RLHF has demonstrated stronger performance compared to the offline approach (Dong et al., 2024; Wu et al., 2024; Yuan et al., 2024b). In this section, we present the results of the online iterative training process to evaluate the effectiveness of our proposed PUGC in this setting. Specifically, we followed the instruction generation and filtering process from the main experiment, using the same 60k instructions and corresponding UGC. The data was randomly split into three subsets, each used for one iteration of training. After each iteration, we sampled responses from the updated model for reward scoring. As shown in Table 4, each iteration yielded steady improvements in LC win rates, with gains of 13.74%, 3.22%, and 3.44% for each respective iteration. Compared to the 35.93% LC win rate in the offline setting, PUGC achieved a 37.51% win rate by the third iteration. Additionally, we observed a significant increase in generation length in the online iterative setting, indicating the need for a better length control method.

B Reward Score Distribution

To further evaluate the reward quality, we visualize the reward distribution for both cases, with and without UGC as reference text, using prompts generated from UGC, as shown in Figure 7. We observe that the reward distribution without UGC reference is sharper, while the distribution with UGC reference is smoother. This suggests that incorporating implicit user preference signals from UGC facilitates more fine-grained reward signals.

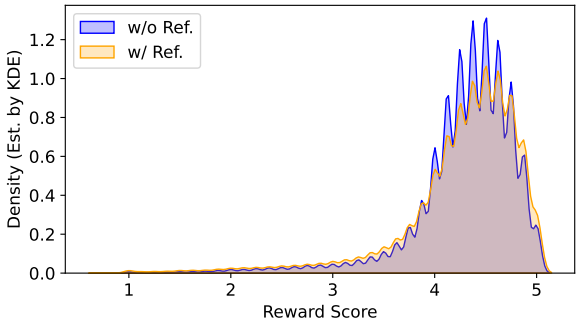

Figure 7: We compare the reward score distribution with and without UGC as a reference. We find that using UGC as a reference results in a smoother reward score distribution, providing a more fine-grained reward signal.

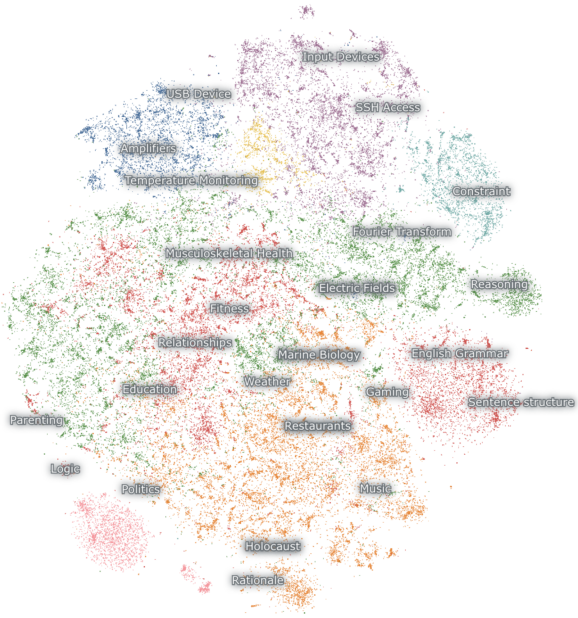

Figure 8: Visualization of our collected UGC data via Nomic Atlas. The figure is colored by topics, which are auto-generated by the Nomic Atlas custom topic model.

C Selected UGC Distribution

We visualize the distribution of the collected UGC using Nomic Atlas[3] and present the results in Figure 8. The UGC demonstrates a broad distribution across various domains, with the majority of data concentrated in areas such as technology and devices, health and medical topics, education and language, social and political issues, entertainment and leisure, as well as legal and historical subjects. This distribution highlights the model's significant performance improvements in theory of mind capabilities, enabling a deeper understanding of users' intentions and beliefs. However, the

[3] https://atlas.nomic.ai/

Table 5: Breakdown of Alpaca Eval test set instructions by instruction category.

Category	Number	Percentage
general knowledge	55	6.8%
historical/cultural topics	37	4.6%
explanations/reasoning	130	16.1%
directions/navigation	24	3.0%
categorization/classification	59	7.3%
product/service descriptions	68	8.4%
scientific concepts	63	7.8%
writing tasks	19	2.4%
creative writing	87	10.8%
math problems	35	4.3%
review/critique	17	2.1%
email/letter writing	18	2.2%
language/grammar	33	4.1%
hypothetical scenarios	34	4.2%
programming/coding	76	9.4$
social media content	14	1.7%
summarization	10	1.2%
game/puzzle solutions	6	7.4%
script/dialogue writing	9	1.1%
data analysis/interpretation	9	1.1%

Table 6: Breakdown of Alpaca Eval test set instructions by instruction complexity.

Category	Number	Percentage
1	2	0.2%
2	17	2.1%
3	72	8.9%
4	95	11.8%
5	59	7.3%
6	214	26.6%
7	251	31.2%
8	92	11.4%
9	1	0.1%

scarcity of reasoning-related data limits the model's effectiveness in reasoning-intensive tasks, such as mathematics and coding.

D Generated Query Distribution Analysis

To better understand the distribution of generated query from UGC, and its relation with prompts from UltraFeedback and Alpaca Eval test set, we visualize the prompt distribution in Figure 9. Specifically, we use DeBERTa-v3-Large (He et al., 2021) for prompt encoding and apply t-SNE to visualize the prompt embeddings. Our results show that the prompts generated by PUGC are more closely aligned with those from the Alpaca Eval test set, while the UltraFeedback prompts exhibit greater diversity. This observation aligns with the findings in Nguyen et al. (2024).

Table 7: Breakdown of Alpaca Eval test set instructions by expected response length.

Category	Number	Percentage
1 sentence	23	2.9%
1-3 sentences	177	22.0%
1 paragraph	322	40.0%
2 paragraphs	252	31.3%
3 or more paragraphs	29	3.6%

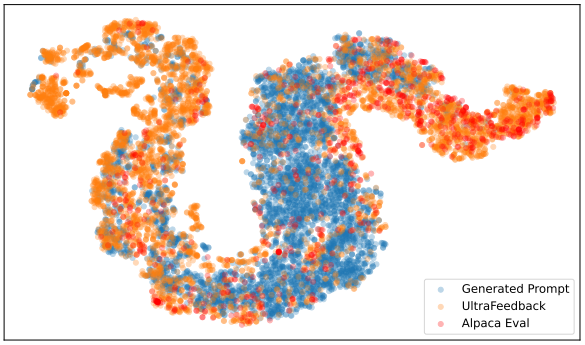

Figure 9: T-SNE visualization of generated prompt from PUGC, UltraFeedback, and Alpaca Eval. The embedding of the prompts are calculated by DeBERTa-v3-Large (He et al., 2021). Our generated prompt are closer to the Alpaca Eval test set prompt, while UltraFeedback prompts distribute more diverse.

E Fine-grained Alpaca Eval Data Statistics

We use Claude-3-sonnet to categorize prompt in Alpaca Eval test set based on task category, instruction complexity, and expected response length. The category statistics can be found in Table 5, 6, and 7. We discard categories that contain less than 10 samples to ensure the evaluation quality.

F Decoding Hyperparameters

In preference data generation, we set temperature to 0.8 with top_p to 0.95 under the base setting and temperature to 0.7 and top_p to 0.9 for readers' instruction generation based on UGC. For instruction filter, we restrict the model's output token to {True, False} and do greedy decoding with max_token=1. For reward score generation, we follow the default hyperparameters in Prometheus (Kim et al., 2024c) and set temperature to 1.0 with top_p=0.9.

In response generation for evaluation, we follow the setting in SimPO (Meng et al., 2024). Specifically, for Alpaca Eval 2.0, we use a sampling decoding strategy to generate responses, with a temperature of 0.7 for the Mistral-Base setting

Table 8: Performance of PUGC on WildBench (Lin et al., 2024) and Arena-Hard (Li et al., 2024).

Model	**WildBench**						**Arena-Hard**
	WB score	Creative Tasks	Planning&Reasoning	Math&Data Analysis	Information/Advice Seeking	Coding&Debugging	
Mistral-7B-instruct	25.63	42.07	30.06	10.08	40.10	18.4	12.6
PUGC+DPO	26.95	46.56	33.36	11.43	40.20	17.16	13.7

Table 9: Performance of PUGC on BigGen Bench (Kim et al., 2024b) evaluate by prometheus-bgb-8x7b-v2.0.

Model	Ground.	Inst. Follow.	Plan.	Reason.	Refine.	Safe.	ToM	Tool.	Multi.	Overall
Mistral-7B-instruct	3.73	3.84	3.63	3.11	3.26	3.75	3.66	3.21	2.33	3.40
UltraFeedback+DPO	3.64	3.68	3.80	3.02	3.29	3.22	3.79	2.49	2.49	3.27
PUGC + DPO	3.71	3.90	3.87	3.09	3.29	3.20	4.30	3.09	2.39	3.43

Table 10: Performance of PUGC on GSM8k (Cobbe et al., 2021), MMLU-Pro (Wang et al., 2024), IFEval (Zhou et al., 2023), and TruthfulQA (Lin et al., 2021).

Model	**GSM8k**	**MMLU-Pro**	**IFEval**	**TruthfulQA**
Mistral-7B-instruct	43.06	30.42	38.83	52.26
PUGC+DPO	41.17	28.37	40.11	42.77

following zephyr-7b-beta,13 a temperature of 0.5 for the Mistral-Instruct setting following Snorkel-Mistral-PairRM-DPO. For MT-Bench, we follow the official decoding configuration which defines different sampling temperatures for different categories.

G Computation Resources Details

All the training experiments in this paper were conducted on a single ndoe with 8 × NVIDIA A100-SXM4-40GB GPUs.

H Scientific Artifacts

PUGC is built with the help of many existing scientific artifacts, including PyTorch (Paszke et al., 2019), Numpy (Harris et al., 2020), huggingface, and transformers (Wolf et al., 2020), alignment-handbook (Tunstall et al.), and RLHFlow (Dong et al., 2024). We will make the PUGC implementation publicly available to facilitate further research.

I Performance on Additional Benchmarks

We also evaluate PUGC on additional benchmarks, including WildBench (Lin et al., 2024), Arena-Hard (Li et al., 2024), BigGen Bench (Kim et al., 2024b), GSM8k (Cobbe et al., 2021), MMLU-Pro (Wang et al., 2024), IFEval (Zhou et al., 2023), and TruthfulQA (Lin et al., 2021). The results are presented in Tables 8, 9, and 10.

In WildBench, PUGC demonstrates significant improvements in creative tasks and reasoning & planning, while showing comparable performance in math & data analysis and information/advice seeking. We hypothesize that PUGC focuses on leveraging implicit user preferences from UGC for alignment, rather than enhancing coding and data analysis capabilities. A similar trend is observed in the Arena-Hard benchmark, which primarily consists of math and coding instructions.

The BigGen Bench results in Table 9 indicate stable performance in grounding, instruction following, planning, reasoning, refinement, and multilingual tasks. However, we observe a notable gain in theory of mind capabilities, supporting the hypothesis that UGC provides rich information about implicit human beliefs, values, and preferences. On the other hand, we observe performance drops in safety and tool use, which are not the focus of this work, and leave improvements in these areas for future research.

For GSM8k, MMLU-Pro, IFEval, and TruthfulQA, shown in Table 10, we note a performance drop in GSM8k, MMLU-Pro, and TruthfulQA, potentially due to the alignment tax (Lin et al., 2023; Ouyang et al., 2022). However, moderate improvement is observed in IFEval, indicating stronger instruction-following capabilities in PUGC.

J Case Study

We present examples of preference data generated from UGC sourced from Dolma in Tables 11, 12, and 13, as well as Pref data derived from Goodreads book reviews in Tables 14, 15, and 16.

In Dolma's preference data, the chosen response highlights implicit human preferences in UGC by focusing on trends like digital subscriptions, social media, and video content, which reflect user

demands for convenience, personalization, and engagement. The emphasis on AI, machine learning, and voice assistants indicates a growing preference for automation, while the resurgence of local news shows a desire for community-focused reporting. These insights capture the shift toward diverse, adaptive, and immersive content experiences.

In Goodreads' preference data, the response mirrors insightful human preferences by emphasizing detailed evaluations of novels, such as language richness, character complexity, and thematic depth. It reflects readers' appreciation for vivid imagery and emotional engagement, while also considering accessibility and discussion potential, aligning with UGC's focus on immersive, thought-provoking content.

Prompt for UGC Quality Scoring

Below is a text document. Evaluate whether the document is a good source for identifying potential user queries and crafting corresponding responses to demonstrate how an AI Assistant should handle user instructions. Please assign a score using the following 5-point scale:

1: The document is incomplete, vague, off-topic, or does not address what the user might want to know. It lacks any originality or valuable insights, making it difficult to extract relevant queries or responses. For example, the content might be missing key information, be poorly organized, or contain irrelevant or promotional text.

2: The document has some potential but does not provide all the necessary information to extract relevant queries or responses. It might offer a high-level overview or general ideas but lacks depth or specificity.

3: The document is helpful and covers basic topics but lacks depth or insightful information. It addresses common user concerns but does so in a generic or superficial manner. While it is self-contained and complete, it may not offer substantial content for crafting nuanced AI Assistant responses.

4: The document is a strong source for extracting potential user queries and crafting responses. It provides a clear, comprehensive, and well-organized presentation of information. While the content is informative and useful, there may still be minor areas for improvement, such as clarity, conciseness or focus.

5: The document is a perfect source for extracting potential user queries and responses. It is high-quality, demonstrating expert knowledge and offering clear, logical, and engaging insights. The document is infused with creative and original ideas that challenge conventional thinking, making it a perfect model for how an AI Assistant should respond to user instructions.

Please first provide a brief reasoning you used to derive the rating score, and then write "Score: " on the last line.

<document start>
{User-Generated Content}
<document end>

Prompt for Reader Question Generation

###Task Instruction
Act as a human and generate a question or instruction based on the provided context. Here is an instruction or query example, you can follow its pattern.

<example start> {Instruction Example} <example end>

<context start> {User-Generated Content} <context end>

Ensure that the provided context contains sufficient information and details to support responses to the question or instruction. Your question or instruction should be complete, independent, and self-contained. Give specific and detailed information in the question or instruction. Only generate one question or instruction and nothing else. The question or instruction cannot explicitly refer to the context.

GPT Judge Prompt Template

<|im_start|>system
You are a highly efficient assistant, who evaluates and selects the best large language model (LLMs) based on the quality of their responses to a given instruction. This process will be used to create a leaderboard reflecting the most accurate and human-preferred answers.
<|im_end|>
<|im_start|>user
I require a leaderboard for various large language models. I'll provide you with prompts given to these models and their corresponding outputs. Your task is to assess these responses, and select the model that produces the best output from a human perspective.

Instruction

{
"instruction": ""{instruction}"",
}

Model Outputs

Here are the unordered outputs from the models. Each output is associated with a specific model, identified by a unique model identifier.

{
{
"model_identifier": "m",
"output": ""{Output_1}""
},
{
"model_identifier": "M",
"output": ""{Output_2}""
}
}

Task

Evaluate the models based on the quality and relevance of their outputs, and select the model that generated the best output. Answer by providing the model identifier of the best model. We will use your output as the name of the best model, so make sure your output only contains one of the following model identifiers and nothing else (no quotes, no spaces, no new lines, ...): m or M.

Best Model Identifier
<|im_end|>

Prompt for Question Filtering

###Context
{User-Generated Content}

###Question
{Generated Instruction}

###Task Instruction
Determine if the context contains accurate, comprehensive, profound, creative, and insightful information relevant to the question. Respond with only True or False without any additional characters or formatting.

Reward Scoring Prompt Template

###Task Description:
An instruction (might include an Input inside it), a response to evaluate, a reference document that contains relevant information, and a score rubric representing a evaluation criteria are given.
1. Write a detailed feedback that assess the quality of the response strictly based on the given score rubric, not evaluating in general.
2. After writing a feedback, write a score that is an integer between 1 and 5. You should refer to the score rubric.
3. The output format should look as follows: "(write a feedback for criteria) [RESULT] (an integer number between 1 and 5)"
4. Please do not generate any other opening, closing, and explanations.

###The instruction to evaluate:
`{Instruction}`

###Response to evaluate:
`{Response}`

###Reference Document:
`{Reference_UGC}`

###Score Rubrics:
`{Rubric}`

###Feedback:

Score Rubric

`criteria`:"Does the response meet the criteria of quality, considering factors such as helpfulness, relevance, accuracy, depth, creativity, and level of detail?"

`score1_description`: "The response is incomplete, vague, off-topic, or not aligned with the user's request. It may contain missing content, start lists incorrectly, or repeat the user's question. The response might reflect another person's perspective, include irrelevant information such as promotional text, or resemble a forum post rather than a direct answer."

`score2_description`: "The response addresses most of the user's requests but does not directly answer the question. It may provide a high-level methodology instead of the exact solution, leaving the user without the specific guidance they were seeking."

`score3_description`: "The response is useful but appears to be written by someone other than an AI assistant. It meets the basic requirements of the user's question but is written from another person's perspective, resembling content from a blog post, web page, or search results rather than an AI-generated answer."

`score4_description`: "The response is crafted from the perspective of an AI assistant, with a clear focus on addressing the user's instruction. It provides a complete, clear, and comprehensive answer without missing or irrelevant information. While well-organized and helpful, there may be minor areas for improvement in clarity, conciseness, or focus."

`score5_description`: "The response is an exemplary AI-generated answer, perfectly aligned with the user's instruction. It is focused, helpful, and free of irrelevant content, delivering high-quality insights and expert knowledge. The response is engaging, logical, insightful, and easy to follow, reflecting the highest standard of quality for an AI assistant."

Rubric Template

[`{criteria}`]
Score 1: `{score1_description}`
Score 2: `{score2_description}`
Score 3: `{score3_description}`
Score 4: `{score4_description}`
Score 5: `{score5_description}`

Prompt used to obtain instruction categories in the AlpacaEval test set.

`<LIST ALL ALPACAEVAL INSTRUCTIONS>`
Given the above list of possible instructions, define a maximum of 20 categories that would cover the types of instructions, for example recipes, reasoning tasks, general knowledge etc. Try to cover as many of the instructions as possible with the maximum 20 categories, while keeping the categories high-level, simple and easy to understand.

Prompt for categorizing instructions.

Instruction: `{Instruction}`

Given the above, categorize it into one of the following 20 categories:

`<LIST ALL CATEGORIES>`

Secondly, score the instruction in terms of complexity: how complex you think it is to answer from 1-10 (where 10 is a complex question whereby first reasoning or breaking down the question into multiple subquestions for example might help improve the answer).

Thirdly, indicate how long you think the response to the instruction should be, either (a) 1 sentence, (b) 1-3 sentences, (c) 1 paragraph, (d) 2 paragraphs, or (e) 3 or more paragraphs.

Provide your final response in the following format:
Category: <one of the 20 categories>
Complexity: <score out of 10>
Length: <length category>. Do not provide the actual response.

Prompt for Domain-Specific Evaluation

[System]
Please act as an impartial judge and evaluate the quality of the responses provided by two AI assistants to the user question displayed below. You should choose the assistant that follows the user's instructions and answers the user's question better. Your evaluation should consider factors such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of their responses. Begin your evaluation by comparing the two responses and provide a short explanation. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Do not favor certain names of the assistants. Be as objective as possible. After providing your explanation, output your final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]" if assistant B is better, and "[[C]]" for a tie.

[User Question]
`{Question}`

[The Start of Assistant A's Answer]
`{Answer_a}`
[The End of Assistant A's Answer]

[The Start of Assistant B's Answer]
`{Answer_b}`
[The End of Assistant B's Answer]

Table 11: Preference data example from Dolma UGC.

User-Generated Content	We all wish we had the power to predict what the world will look like in five, 10 or 50 years. As a sector currently undergoing turbulent change, the media industry is frantically shaking a cloud-filled crystal ball up and down in the hopes of figuring out what the future holds. The afternoon-long workshop provided us with a glimpse of how the IFTF goes about the forecasting process. It turns out that future-gazing is not as mystical as one might think. The goal of the workshop – to explore the future of news and journalism – was largely focused on brainstorming. Together with staff and affiliate researchers from the IFTF, we took it in turns to share "signals" for the future of the media. A signal is an existing example that indicates the shape of the future. e.g. a prototype service, a lab demo or a current news item. The task wasn't simply to share ideas about the latest media-centric iPhone app or crowd-sourced content strategy. Marina and her team also charged us with inferring what an existing signal says about a future trend for journalism. Over the course of about 90 minutes, we came up with a voluminous slew of signals and trends, filling the long white-board at the front of IFTF's workspace with dozens of colorful sticky notes. Marina and her colleague Mike Liebhold (a distinguished fellow at the Institute) then tried to organize the ideas into clusters representing various trend areas. The breadth of ideas we came up with suggests nothing if not an excitingly multi-dimensional future for our industry – a future that is already in embryonic stage today. For example, the existence of norran.se, a Swedish website which enables readers to participate in the editorial decision-making process, and the "sousveillance" movement, point to deeper public involvement in news-gathering and disseminating in the years ahead; the demise of the personal computer and rise of the "deckchair-friendly" tablet interface suggest a bigger market for the consumption of longer-form journalism; meanwhile, the increasing prevalence of non-native-language news sources in various parts of the world illustrates a trend towards the trans-border news interests of a switched-on society that is deeply curious about how people in other cultures feel and think. A robust future? From capitalizing on new interfaces and platforms for the dissemination of news to creating innovative forms of algorithmic journalism that tailor information to suit a user's location, interests and even mood, the media industry has plenty of room for a robust future. Of course, while there is no shortage of interesting ways in which our sector can grow in the years ahead, one thing remains cloudy in that crystal ball: how anyone is going to make any money. Not even the augurs at IFTF have a handle on that.
Generated Instruction	What trends in the media industry, indicated by specific signals, suggest new ways for news-gathering, dissemination, and consumption, and how might these trends impact the revenue model for media companies?
Chosen Response	1. Digital Subscriptions: The trend of consumers paying for digital content is on the rise. According to a report by PwC, global digital subscription revenue in the media industry is expected to reach $52 billion by 2024. This shift towards digital subscriptions is a significant change from the traditional revenue model of advertising-driven revenue. 2. Social Media: Social media platforms have become an integral part of news-gathering, dissemination, and consumption. According to a study by Pew Research Center, 62% of US adults get news on social media. This trend has significant implications for media companies as they need to adapt to this new distribution channel and find ways to monetize it. 3. Video Content: Video content is becoming increasingly important in the media industry. According to a report by Cisco, video will account for 82% of all internet traffic by 2022. Media companies are investing heavily in video production and distribution to meet the growing demand for video content. 4. Podcasts: Podcasts are gaining popularity as a new medium for news and entertainment. According to a report by Edison Research, 51% of Americans have listened to a podcast, up from 40% in 2018. Media companies are exploring new revenue streams through sponsorships and subscriptions for podcasts. (continued)

Table 12: Preference data example from Dolma UGC (continued).

Chosen Response (continued)	(continued) 5. Artificial Intelligence and Machine Learning: AI and ML are being used to personalize news content for individual consumers. This trend is expected to continue as media companies look for ways to differentiate themselves in a crowded market. However, the use of AI and ML also raises ethical concerns around privacy and data security. 6. Voice Assistants: Voice assistants such as Amazon Alexa and Google Assistant are becoming increasingly popular for news consumption. According to a report by eMarketer, 24.5 million US adults will use a voice assistant for news in 2022. Media companies are exploring new ways to deliver news through voice assistants and monetize this new distribution channel. 7. Local News: Local news is experiencing a resurgence as consumers look for more personalized and community-focused content. According to a report by Pew Research Center, 43% of US adults get local news daily. Media companies are investing in local news reporting and delivery to meet this demand. Overall, these trends suggest a shift towards personalized and subscription-based news consumption, with a focus on digital and video content. Media companies need to adapt to these trends by investing in new technologies, delivery channels, and business models to remain competitive. However, these trends also come with challenges around privacy, data security, and ethical considerations that need to be addressed.
Rejected Response	1. Digital Subscriptions: The shift towards digital consumption of news and the success of subscription models, as seen in the case of The New York Times, The Washington Post, and The Guardian, suggest a new revenue model for media companies. These subscriptions provide a steady stream of revenue, unlike traditional advertising models that rely on high traffic and impressions. 2. Social Media as News Sources: The increasing use of social media as a news source indicates a trend towards decentralized news-gathering and dissemination. This trend challenges traditional media companies' monopoly on news production and distribution. It also creates new opportunities for revenue generation through targeted advertising and partnerships with social media platforms. 3. Podcasting: The rise of podcasting as a popular form of news consumption indicates a trend towards audio content. This trend provides an opportunity for media companies to explore new revenue streams through podcast advertising and sponsorships. 4. Artificial Intelligence and Automated Journalism: The use of artificial intelligence (AI) and automated journalism indicates a trend towards data-driven journalism and content generation. This trend can lead to cost savings for media companies and increased efficiency, but it also raises ethical concerns regarding accuracy and fact-checking. 5. Video Content: The trend towards video content, as seen in the success of YouTube and streaming services, suggests a shift towards visual storytelling and entertainment. This trend creates new opportunities for media companies to generate revenue through advertising and subscription models. However, it also requires significant investment in production and distribution infrastructure. 6. Personalization: The trend towards personalized content and recommendations, as seen in the success of Netflix and Spotify, suggests a future where media consumption is tailored to individual preferences. This trend provides opportunities for media companies to generate revenue through targeted advertising and subscription models based on user data. 7. Localized News: The trend towards localized news and hyperlocal journalism indicates a shift towards community-focused reporting and away from national and international news. This trend creates new opportunities for revenue generation through targeted advertising and subscription models based on local demographics and interests. (continued)

Table 13: Preference data example from Dolma UGC (continued).

Rejected Response (continued)	(continued) 8. Programmatic Advertising: The trend towards programmatic advertising, which uses algorithms to target specific audiences, suggests a future where advertising is more efficient and effective. This trend provides opportunities for media companies to generate revenue through targeted advertising based on user data. 9. Virtual and Augmented Reality: The trend towards virtual and augmented reality content suggests a future where news and entertainment are more immersive and interactive. This trend creates new opportunities for revenue generation through advertising and subscription models based on user engagement and experience. 10. Blockchain Technology: The trend towards blockchain technology in the media industry suggests a future where content is verified, secure, and transparent. This trend creates new opportunities for revenue generation through subscription models based on trust and security. It also provides opportunities for new business models based on microtransactions and tokenization.

Table 14: Preference data example from Goodreads UGC.

User-Generated Content	SECOND READING: Sept 2, 2014 I've just re-read THE GOLDFINCH by Donna Tartt because I chose this book for my book club to discuss and I wanted to refresh my memory on its details. The first time I read it (see my original review below), I loved it for many reasons. While most of them still hold true for me, I have to say I didn't find it worthy of the 2014 Pulitzer Prize for Fiction which it won since my previous review. Donna Tartt is a storyteller extraordinaire. I love the depth of her details and the reaches of her imagination. I am surprised however by her abundant usage of adverbs and cliches. This alone should have knocked it out of contention for the Pulitzer, in my opinion. Beyond that, while I grew extremely tired of Theo and Boris's use of drugs during my first reading, I accepted it because I was so into the story. The second time through (plus hearing how tired everyone else grew of it too) made me realize that it was truly way over done. I would recommend this book only to people who love an epic novel. I would also knock this down to 4.5 Stars if Goodreads would let me. FIRST READING: Jan 31, 2014 You know how sometimes a person, or a thought, or a book hits your life at just the right moment, and you learn something from him/her/it? And you somehow know that if it had happened at any other time in your life, it might never have touched you in the same way? Well, THE GOLDFINCH hit me at just the right moment. I loved this book. THE GOLDFINCH is 771 pages. Is that too long? Probably. Yet I can't think of a single scene I would remove. Even during the passages where I wondered, What's the point of this? Where is it going? Yeah, Donna Tartt always answered those questions in due time. Where to begin? The prose, the plot, the characters, the theme? All wonderful and unique. Tartt's description of each character is so vivid, so fantastically specific that I could picture each one in turn. What a lesson in flawed characters! Even the ones we're "supposed" to like, we don't always. And isn't that the way it is in real life? Don't you even dislike your best friend every now and then? Get a little mad at your spouse on occasion? Exactly. THE GOLDFINCH is the story of Theo Decker, a thirteen-year-old boy who finds himself in possession of a masterwork of art from the Metropolitan Museum after a terrorist bombing that killed his mother. The way Tartt leads him from that shell-shocked boy to a damaged man of twenty-eight feels so realistic, so deep and probable that if someone told me this was based on a true story, I would probably believe it. Not that I have much in common with Theo or his childhood friend, Boris, or his mentor, Dobie, or any of the other characters (at least not the big stuff), but I loved them all because of how this story ended. The ending was perfect. Plausible, yet not sweetly Hollywood-slick. Though, the book is so descriptive, it could easily be turned into a movie. I would love to discuss the theme of this book and the character development (Theo is the perfect example of a reluctant hero), but I don't want to give too much away. If you're debating on this book due to the length, let me say, Yes, it's long and winding, but the journey is fabulous and the ending will feel like a reward. I have a few knocks against the book, but they're not important enough to alter my rating. I will be choosing this book for my book club because I am eager to discuss so many elements. Also, if anyone is curious, not only did I read the book, I listened to it on audible.com. The narrator, David Pittu, does an excellent job with a multitude of characters and foreign accents. The audio version added so much to my reading experience. I highly recommend it. Five drunk, gritty, gorgeous Stars (Read the book; you'll understand.)
Generated Instruction	How would you evaluate the use of descriptive language and character development in a lengthy novel, and what elements would you consider before recommending it to a book club?

Table 15: Preference data example from Goodreads UGC (continued).

Chosen Response	Evaluating the use of descriptive language and character development in a lengthy novel involves a close reading and analysis of the text. Here are some elements to consider: 1. Descriptive Language: - Richness and Vividness: Does the author use vivid and rich language to paint a picture of the setting, characters, and actions? Is the language engaging and evocative, making the reader feel immersed in the story? - Consistency: Is the descriptive language consistent throughout the novel? Does it help to establish a tone and mood that is maintained throughout the story? - Sensory Details: Does the author use sensory details effectively to help the reader experience the story through the five senses? Are there compelling descriptions of sights, sounds, smells, tastes, and textures? - Imagery: Does the author use powerful and memorable imagery to help the reader visualize the story? Are there metaphors, similes, and other figurative language that add depth and meaning to the narrative? 2. Character Development: - Complexity: Are the characters multi-dimensional and complex, with well-defined motivations, strengths, weaknesses, and backstories? Do they grow and change throughout the novel, or do they remain static? - Consistency: Is the character development consistent with the narrative and the characters' actions and dialogue? Do the characters behave in ways that are true to their nature and the story? - Relatability: Are the characters relatable and sympathetic, or are they distant and unengaging? Do readers care about what happens to them and why? - Interactions: How do the characters interact with each other? Are there compelling relationships and conflicts between them? Before recommending a lengthy novel to a book club, here are some additional elements to consider: - Themes: What are the major themes of the novel, and how do they resonate with readers? Are they timely and relevant, and do they spark meaningful discussion? - Writing Style: Is the writing style accessible and engaging, or is it dense and challenging? Will all members of the book club be able to engage with the text? - Diversity: Does the novel offer diverse perspectives and representations of people and cultures? Does it challenge readers to broaden their understanding of the world? - Accessibility: Is the novel accessible to all members of the book club, or does it require a significant investment of time and resources (e.g., specialized knowledge, research, or translation)? - Discussion Questions: Are there thought-provoking discussion questions that will help the book club explore the novel's themes and characters in depth? Will these questions encourage lively and engaging conversation?
Rejected Response	Evaluating the use of descriptive language and character development in a lengthy novel involves a close reading and analysis of the text. Here are some key elements to consider: 1. Descriptive Language: - Vivid Imagery: Does the author use descriptive language to paint a clear and vivid picture of the settings, characters, and emotions? Is the imagery evocative and transporting, allowing readers to immerse themselves in the story? - Sensory Details: Does the author use all five senses to describe the world of the novel? Are there rich and specific details about sights, sounds, smells, tastes, and textures that bring the story to life? - Figurative Language: Does the author use figurative language effectively to add depth and meaning to the descriptions? Are there metaphors, similes, or other figurative devices that enhance the reader's understanding of the story? (continued)

Table 16: Preference data example from Goodreads UGC (continued).

Rejected (continued)	**Response**	(continued) 2. Character Development: - Complexity: Are the characters well-developed and complex? Do they have distinct personalities, motivations, and backstories that make them feel real and relatable? - Consistency: Are the characters consistent in their actions, thoughts, and dialogue throughout the novel? Do they grow and change in believable ways, or do they remain stagnant? - Interactions: How do the characters interact with each other? Are there compelling and dynamic relationships that drive the plot forward and deepen the readers' understanding of the characters? Before recommending a lengthy novel to a book club, here are some additional elements to consider: - Thematic Depth: Does the novel explore complex themes that would lead to rich and engaging discussions? Are there thought-provoking ideas that would resonate with the members of the book club? - Accessibility: Is the novel accessible to a wide range of readers? Is the language complex or dense, making it challenging for some members to engage with? - Diversity: Does the novel offer diverse perspectives and representations? Are there characters from different cultural backgrounds, or does the novel explore issues of race, gender, sexuality, or disability in meaningful ways? - Engaging Plot: Does the novel have a compelling and engaging plot that would keep the members interested and invested in the story? Overall, a lengthy novel that uses rich descriptive language and develops complex and compelling characters, while exploring complex themes and offering diverse perspectives, would make for an excellent book club selection.