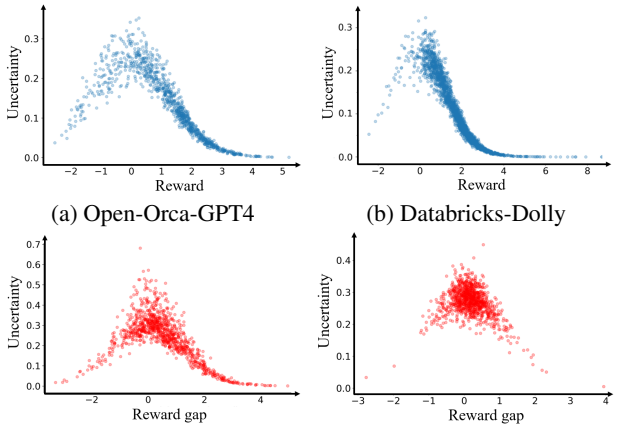
Improving Instruction Following in Language Models through Proxy-Based Uncertainty Estimation

JoonHo Lee [*1] **Jae Oh Woo** [*2] **Juree Seok** [1] **Parisa Hassanzadeh** [2] **Wooseok Jang** [1] **JuYoun Son** [1] **Sima Didari** [2] **Baruch Gutow** [2] **Heng Hao** [2] **Hankyu Moon** [2] **Wenjun Hu** [2] **Yeong-Dae Kwon** [1] **Taehee Lee** [1] **Seungjai Min** [2]

Abstract

Assessing response quality to instructions in language models is vital but challenging due to the complexity of human language across different contexts. This complexity often results in ambiguous or inconsistent interpretations, making accurate assessment difficult. To address this issue, we propose a novel Uncertainty-aware Reward Model (URM) that introduces a robust uncertainty estimation for the quality of paired responses based on Bayesian approximation. Trained with preference datasets, our uncertainty-enabled proxy not only scores rewards for responses but also evaluates their inherent uncertainty. Empirical results demonstrate significant benefits of incorporating the proposed proxy into language model training. Our method boosts the instruction following capability of language models by refining data curation for training and improving policy optimization objectives, thereby surpassing existing methods by a large margin on benchmarks such as Vicuna and MT-bench. These findings highlight that our proposed approach substantially advances language model training and paves a new way of harnessing uncertainty within language models. Code is available at https://github.com/P-B-U/proxy_based_uncertainty.



(a) Open-Orca-GPT4 (b) Databricks-Dolly (c) Anthropic-Helpful (d) OpenAI-Summarize

Figure 1. Uncertainty distributions evaluated using the proposed URM (Uncertainty-aware Reward Model) for individual rewards in the instruction tuning or SFT data (a, b) and reward gaps in the preference data (c, d) are illustrated. These results show that even for responses (or preferences) with an identical reward (or reward gap), their uncertainty is distributed across a wide range.

1. Introduction

Language represents a complex form of human interaction. Its meaning varies significantly across specific cultural and social contexts and evolves over time with diverse interpretations (Wittgenstein, 1953). The response quality in conversations is often assessed by how well it follows implicit rules inherent in the context. In language models (LMs), enhancing their instruction-following capability is crucial for reliable deployment since generating either unhelpful responses or harmful content poses various risks.

In line with earlier studies (Leike et al., 2018; Ziegler et al., 2020), Reinforcement Learning from Human Feedback (RLHF) has been proposed to enhance the instruction following of LMs (Ouyang et al., 2022; Glaese et al., 2022; Bai et al., 2022b; Touvron et al., 2023). This approach involves training proxy models, or reward models, using datasets with human preference annotations for paired responses to instructions. These proxy models generate reward scores and play a crucial role in facilitating the subsequent policy optimization framework through Reinforcement Learning (RL) algorithms such as Actor-Critic (Mnih et al., 2016) or PPO (Schulman et al., 2017). However, the assessment of response quality can be diverse based on the background of each annotator, which leads to ambiguity in its interpre-

[*]Equal contribution [1]Samsung SDS Technology Research, Seoul, Korea [2]Samsung SDS America, San Jose, California, USA. Correspondence to: JoonHo Lee <joonholee@samsung.com>, Jae Oh Woo <jaeoh.woo@aya.yale.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

tation. For instance, assuming that some annotators prefer direct and clear responses while others favor diplomatically phrased ones, we could find that the response score to such culturally sensitive questions can vary based on the composition of annotators. While some works employ majority voting for annotation (Ouyang et al., 2022) or include additional annotation for the significance of preferences (Touvron et al., 2023), these approaches face challenges in resolving such ambiguity in annotations or the uncertainty in the response quality, given that conventional proxy models generate rewards. As illustrated in Fig. 1a-b, responses with identical rewards exhibit a broad uncertainty distribution. A similar pattern is evident in the reward gap between the *chosen* response and the *rejected* response within preference datasets, as shown in Fig. 1c-d.

While RLHF enhances instruction following of LMs, its complex training pipeline increases computational costs. Recently, DPO (Rafailov et al., 2023), PRO (Song et al., 2023), and C-RLFT (Wang et al., 2023) offer a shift towards new policy optimization as alternatives to RLHF by integrating optimal policy solutions into supervised learning objectives. Whereas DPO and PRO optimize using annotated preferences between responses without reward modeling, C-RLFT employs class-labeled mixed-quality datasets that are generated using an expert model (GPT-4) (OpenAI, 2023) or a suboptimal model (GPT-3.5). These approaches reduce training complexity compared to RLHF while demonstrating impressive conversational abilities. However, uncertainty in evaluating responses remains a challenge when utilizing preference datasets in DPO or PRO. In C-RLFT, this issue can be worse since it employs single-response datasets with coarse-grained reward classes, where the expert responses receive uniformly ten times higher rewards than the suboptimal ones.

Despite demanding interest regarding uncertainty estimation in response quality, assessing the output quality of generative models such as LMs poses a significantly greater challenge than that of a discriminative model due to its larger output dimension. Quantifying the uncertainty of output responses in LMs is more difficult since measuring the uncertainty of sentences by merely aggregating uncertainty measures for individual tokens is often impractical.

In this work, we employ a novel Uncertainty-aware Reward Model (URM), that is trained independently, to address such issues. Our proposed URM measures the quality of the generated responses and their preferences by combining reward scoring with the inherent uncertainty. We translate the **uncertainty of preferences** to that of binary classification on the preferred response based on information-theoretic analysis under the Bayesian approximation framework. Tailored for this context, our strategy facilitates the indirect quantification of uncertainty in preference datasets. We extend this interpretation to the instruction tuning or supervised fine-tuning (SFT) datasets that comprise a single response to the instruction. Once trained using common preference datasets, URM produces reward scores to responses and evaluates their uncertainty. Our URM proves effective for data curation that enables reorganizing and filtering the training corpus, thus establishing appropriate training curricula. We also demonstrate that the instruction-following capability of LMs can be significantly improved by incorporating our method into the existing training objectives. Empirical results evidence the effectiveness of our proxy-based uncertainty approach and highlight the efficacy of considering uncertainty in improving instruction following in LMs. Our contributions are outlined as follows:

- We propose a proxy-based uncertainty framework to enable a more reliable quality assessment for the generated responses or preferences of LMs (Sec. 3).
- We present how to curate mixed-quality datasets for enhanced training curricula based on the proposed Uncertainty-aware Reward Model (URM) that evaluates uncertainty over paired response rewards (Sec. 4).
- We introduce Uncertainty-aware Direct Preference Optimization (UDPO) and Uncertainty-Conditioned Policy Optimization (UCPO) methods that improve the instruction-following capability of LMs by employing information assessed by our URM into existing training objectives and datasets (Sec. 5).
- We empirically demonstrate the effectiveness of our approach across data curations and training objectives for policy optimization in LMs (Sec. 4 and Sec. 5).

2. Related Work

Reward Modeling In RL, a reward refers to a scalar feedback signal used to evaluate an agent's action in a specific state within an environment. It guides the agent towards its goal by reinforcing desirable behaviors. This makes the design of reward functions crucial. In the context of LMs, designing the reward function is challenging due to users' implicit understanding of task goals (Leike et al., 2018). Assuming that evaluating outcomes is simpler than generating correct behaviors, a solution to this challenge has been suggested as aligning the policy with user intentions (Leike et al., 2018). A later study (Ziegler et al., 2020) fine-tuned a pre-trained LM as a reward model using human preference datasets, adapting its output for regression. Numerous LMs, particularly LLMs (Large Language Models), follow this idea when modeling reward functions to involve them in the RL framework. Our proxy-based approach mainly differs from existing methods in enabling the reward model to be aware of the uncertainty in complete sentences.

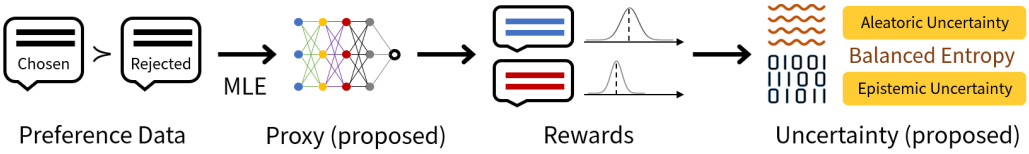


Figure 2. Our proposed proxy, Uncertainty-aware Reward Model (URM), is trained to predict response rewards in preference data. It employs Monte-Carlo dropout for Bayesian approximation to output reward distributions while estimating robust uncertainty from them.

Policy Optimization in LMs RLHF improves the instruction following capability of LMs through *indirect* policy optimization that leverages reward models (Ouyang et al., 2022; Glaese et al., 2022; Bai et al., 2022b; Touvron et al., 2023). Their generated reward scores are instrumental in the subsequent RL policy optimization framework.

In direct policy optimization, notable approaches include DPO, PRO, and C-RLFT (Rafailov et al., 2023; Song et al., 2023; Wang et al., 2023). DPO optimizes human preferences as an LM policy without explicit reward modeling or RL. It integrates an optimal policy solution into the PPO (Schulman et al., 2017) objective to derive an implicit reward function and employs supervised learning with ranking loss among implicit rewards. PRO extends this preference optimization to pairs of comparisons by prioritizing the best response while ranking others to match human preferences. C-RLFT uses mixed-quality SFT datasets from different sources such as GPT-4 or GPT-3.5. It derives supervised policy optimization from an adapted PPO, where a class-conditioned reward function assigns coarse-grained rewards based on the data source quality. Our work highlights the effectiveness of our URM approach in direct policy optimization, particularly DPO and C-RLFT.

Uncertainty of Reward Models in RLHF Appropriate use of uncertainty in deep learning significantly enhances data efficiency and model training (Gawlikowski et al., 2023). A range of uncertainty estimation methods have emerged, with Monte-Carlo (MC) dropout (Gal & Ghahramani, 2016) being notably prevalent.

Despite the efficacy of RLHF methods, we meet challenges such as informative query sampling with substantial feedback needs (Biyik & Sadigh, 2018; Sadigh et al., 2017; Biyik et al., 2020; Lee et al., 2021) and the over-optimization problem of reward models (Gao et al., 2022; Coste et al., 2024). Associated with these issues, Liang et al., 2022 introduced an exploration bonus based on the disagreement across an ensemble of reward models, while Zhai et al., 2023 adopted a diverse ensemble of LoRA (Hu et al., 2021) parameters of a reward model to incorporate uncertainty regularization during PPO. In contrast, we configure the reward model with MC dropout to produce Bayesian uncertainty. We primarily employ our uncertainty-aware proxy to direct policy optimization, where reward models are not explicitly used.

Curriculum Learning Curriculum learning (CL) involves the structured organization of data samples during training, which offers advantages compared to random shuffling (Soviany et al., 2022). In LMs, CL has been integrated with linguistically motivated curricula (Liu et al., 2018; Campos, 2021; Zhang et al., 2021; Weber et al., 2023). Recent advancements in CL exhibit potential for improving LMs in both pre-training and post-training phases.

CL has been employed to address cost and instability during LM pre-training, particularly with large scales. For metrics to establish curriculum, sequence length (Li et al., 2021), complexity from length, rarity, and comprehensibility (Ranaldi et al., 2023; Pucci et al., 2023), and word frequency (Wang et al., 2022) have been considered. Lee et al., 2022 gradually add concepts related to initial ones using a knowledge graph, while Nagatsuka et al., 2021 increases the input block size for training self-attention modules for faster and better optimization in downstream tasks.

During post-training, recent works automate LM fine-tuning by replacing human preferences or exclude RL for efficiency. CL often effectively enhances the performance of these methods. Xu et al., 2023 proposed contrastive post-training to reduce the need for reward signals. They trained a model with DPO from simpler to tougher pairs, approximating task difficulty by the LM prediction gap. Memorization-Based Curriculum (MBC) learning (Ge et al., 2023) prioritizes samples for which the model exhibits lower perplexity. Unlike conventional CL, MBC emphasizes harder training records over easier ones. Chen et al., 2024 train a strong LM from a weaker one by Self-Play fIne-tuNing (SPIN), learning from easy responses to the challenging ones.

While our proxy-based uncertainty holds potential for pre-training application, this work mainly focuses on enhancing the instruction following of LMs during post-training.

3. Proxy-Based Uncertainty

Our approach employs reward modeling to create a distinctive uncertainty-aware proxy function akin to RLHF methods. We refine this proxy function to produce a distribution, drawing inspiration from the methods in (Lindley,

1956; Der Kiureghian & Ditlevsen, 2009; Gal & Ghahramani, 2016; Gal et al., 2017; Kendall & Gal, 2017; Depeweg et al., 2018). Uncertainty assessment follows techniques specified in Woo, 2023, employing MC dropout sampling.

The inherent characteristics of our proxy function render it an effective reward model. It adeptly generates reward scores for responses to given instructions, leveraging its extensive training and knowledge base. However, our proposed proxy goes beyond mere conformity with established RLHF strategies by introducing a new dimension of uncertainty evaluation. In this section, we describe how the proxy evaluates uncertainty over rewards. We present the basic structure of the proxy in Sec. 3.1 and provide an in-depth explanation of how we quantify uncertainty using it in Sec. 3.2. Its training details are elaborated in Sec. 3.3 and followed by case studies on LM training data analysis in Sec. 3.4.

3.1. Proxy as Reward Function

We quantify uncertainty using a reward model that assigns rewards R_c and R_r for chosen and rejected responses, Y_c and Y_r, in a given paired data, respectively. We intend to measure inherent uncertainty to the differences between two reward values as illustrated in Fig. 3. We assume that these reward values adhere to a Gaussian distribution, implying that the difference, $R_c - R_r$, also follows a Gaussian distribution. This perspective reshapes the estimation of the uncertainty of preference data into a binary classification problem. We advocate using a Bayesian Neural Network in which a sigmoid function is integrated. Our proxy model is designed to estimate the uncertainty of the probabilities generated by this model. It offers a comprehensive framework for leveraging these reward estimations in uncertainty assessment and broader decision-making scenarios to improve the model's performance.

3.2. Proxy-Based Uncertainty Quantification

The central concept revolves around converting the reward into a sigmoid probability. Specifically, we establish the probability that a pair is correctly aligned as $P_{y_c \succ y_r} := \sigma(R_c - R_r)$, where $\sigma(\cdot)$ denotes the sigmoid function. This refined approach allows a mathematically grounded interpretation of the model's preference probabilities. Thus, we can construct a binary-class probability $\mathbf{P} := (P_{y_c \succ y_r}, P_{y_c \prec y_r})$ where $P_{y_c \prec y_r} := 1 - P_{y_c \succ y_r}$ by definition. Following the Bayesian framework, we have the information-theoretic uncertainty formula as follows:

$$U_{\text{Epistemic}}(\mathbf{P}) := H(\mathbb{E}\mathbf{P}) + \mathbb{E}\left(\sum_{i \in I} P_i \log P_i\right),$$

$$U_{\text{Aleatoric}}(\mathbf{P}) := -\mathbb{E}\left(\sum_{i \in I} P_i \log P_i\right),$$

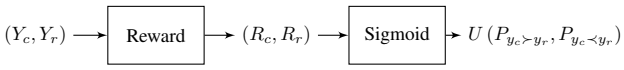


Figure 3. Uncertainty quantification framework.

where $H(\mathbb{E}\mathbf{P}) = -\sum_{i \in I} \mathbb{E}P_i \log \mathbb{E}P_i$ as Shannon entropy (Shannon, 1948) and $I = \{y_c \succ y_r, y_c \prec y_r\}$. The detailed information-theoretic formulation of two uncertainties can also be found in (Woo, 2022).

Epistemic and aleatoric uncertainties are key types of uncertainty in understanding the nature of model and data uncertainties. Epistemic or model uncertainty stems from incomplete knowledge about the model. It's often reducible by enhancing the model or collecting more data. Aleatoric uncertainty, in contrast, arises from the inherent randomness or variability in the data itself and is typically irreducible.

More recently, Balanced Entropy (Woo, 2023) was proposed as an alternative metric that addresses informativeness and active learning diversification. Balanced Entropy integrates epistemic and aleatoric uncertainties within its structure from a Bayesian perspective. Since Balanced Entropy can naturally lend itself to uncertainty calculation in our proxy framework, we aim to propose the Balanced Entropy concept for preference sampling. A notable challenge in this context is the inability to assume the beta distribution approximation due to the effects of sigmoid transformation. To address this, we have developed a new formulation for calculating Balanced Entropy within the constraints of a sigmoid regime. As a result, our Balanced Entropy can be obtained by applying the following formula:

$$U_{\text{BalEnt}}(\mathbf{P}) := \frac{\mathbb{E}P_{y_c \succ y_r} h\left(P^+_{y_c \succ y_r}\right) + \mathbb{E}P_{y_c \prec y_r} h\left(P^+_{y_c \prec y_r}\right) + H(\mathbb{E}\mathbf{P})}{H(\mathbb{E}\mathbf{P}) + \log 2}, \tag{1}$$

where $P^+_{y_c \succ y_r}$ is the posterior distribution of $P_{y_c \succ y_r}$ and $h(\cdot)$ is a differential entropy. Detailed derivations are explained in Appendix A.

Balanced Entropy, in essence, effectively quantifies the balance of information between the model and the data through a normalized ratio, adeptly depicting the dynamics and respective influences of these two types of uncertainty. It thus provides a comprehensive view of the uncertainty landscape, acknowledging the importance of uncertainties inherent in the model and those in the data. To fully represent these complexities, we often resort to the exponentiated value of Balanced Entropy, typically denoted as

$$U := \exp(U_{\text{BalEnt}}) \in [0, e]. \tag{2}$$

This geometric scaling is pivotal in encapsulating the interplay of these uncertainties.

Table 1. Human preference data for reward modeling.

Dataset	# Comparisons	Avg. # Turns	Avg. # Tokens
Anthropic-Helpful	118,261	3.0	254.1
OpenAI-Summarize	92,858	1.0	411.8
Synthetic-GPT-J	33,143	1.0	125.1
BeaverTails-Helpful	145,978	1.0	92.5
Anthropic-Harmless	42,510	3.0	157.8
Total	432,750	1.9	208.3

3.3. Proxy Model Training

We leverage the mixture of publicly available preference datasets $\mathcal{D} = \{X, Y_c, Y_r\}$ comprising pairs of responses (y_c, y_r) to an instruction x. By framing the comparison as a binary classification, we can train a proxy model u_ϕ, or URM, using the negative log-likelihood loss as follows:

$$\mathcal{L}_{\text{URM}} = -\mathbb{E}_{(x,y_c,y_r)\sim\mathcal{D}}[\log \sigma(u_\phi(y_c; x) - u_\phi(y_r; x))], \tag{3}$$

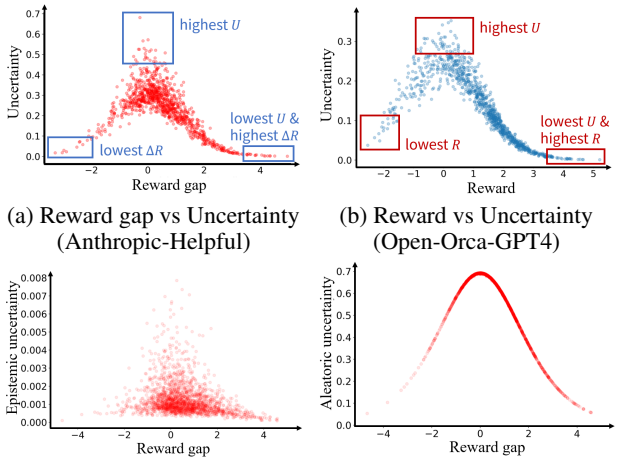
where u_ϕ is initialized from a pre-trained LM. We augment the network's embedding output with a linear layer to enhance MC dropout inference and a subsequent regression head to produce a reward. Its linearly scalable estimation allows for compute-efficient application to larger datasets.

We conducted a single training epoch using the complete reward model training set as described in Table 1. Further training details are presented in Appendix D.1. The preference datasets and proxy performance are detailed in Appendix C.1 and Appendix D.2, respectively.

3.4. Proxy-Based Analysis

We comparatively analyze the uncertainty values calculated through the reward gap with URM in the context of the Anthropic-Helpful dataset (Bai et al., 2022a). Fig. 4a illustrates the relationship with exponentiated Balanced Entropy U, Fig. 4c with epistemic uncertainty, and Fig. 4d with aleatoric uncertainty, respectively. Fig. 4d shows that aleatoric uncertainty correlates closely with the reward gap. Because training URM and calculating uncertainties utilize the same dataset, it results in generally low epistemic uncertainty and a weak correlation with the reward gap. Note that this does not necessarily imply that the order of epistemic uncertainty is less important (see Sec. 4). Meanwhile, Balanced Entropy encompasses both epistemic and aleatoric uncertainties owing to its intermediate level of correlation. This enables more distinctive application of reward gap and uncertainty while embracing both epistemic and aleatoric uncertainties. We further leverage this strength in Sec. 5.

Fig.4b shows a correlation between rewards and U for 10K samples from Open-Orca (Lian et al., 2023), mirroring the pattern in Fig.4a. Assumptions for SFT data analysis are discussed in Sec 5.2. Analyzing the 500 highest and lowest U samples reveals that low U is frequently observed in longer responses while high U is not related to response length, indicating longer responses tend to receive higher rewards and thus lower U as shown in Fig. 4b.



(a) Reward gap vs Uncertainty (Anthropic-Helpful)

(b) Reward vs Uncertainty (Open-Orca-GPT4)

(c) Reward gap vs Epistemic Uncertainty (Anthropic-Helpful)

(d) Reward gap vs Aleatoric Uncertainty (Anthropic-Helpful)

Figure 4. Uncertainties w.r.t. rewards or reward gaps are illustrated. Lower reward gaps lead to higher uncertainty as in (a).

The evaluated uncertainties and rewards allow for more comprehensive dataset analysis. For instance, the BeaverTails-Helpful dataset (Ji et al., 2023) shows higher balanced entropy and aleatoric uncertainty than other preference datasets despite its high preference accuracy on the held-out set by reward models trained using it. Thus, its trained reward models may exhibit behaviors different from those predicted by preference accuracy alone. (refer to Appendix G).

4. Uncertainty-Guided Data Curation

An extensive spectrum of public data is employed in training open-source LMs. This ranges from meticulously curated, high-quality datasets to lower-quality ones that may negatively impact training. The efficacy of LMs is profoundly shaped by the caliber of the data they are trained on, making the judicious curation of these diverse datasets vital. Nonetheless, this curation process is notably laborious and resource-intensive.

Consequently, data selection is crucial. In this section, we aim to demonstrate how leveraging the uncertainty obtained through URM can significantly enhance model training. Our focus is to illustrate the effectiveness of this approach in developing more advanced models.

4.1. Why Curriculum Matters - A Negative Example

In this section, we discuss how the curriculum of introducing preference data to the model during the preference

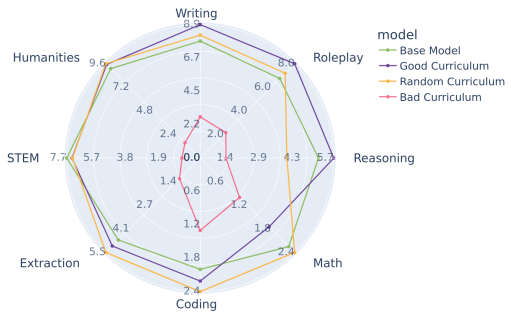


Figure 5. A poorly constructed training curriculum can lead to significantly inferior model performance for the same preference training dataset.

learning (alignment) phase can significantly impact its performance and instruction-following capabilities. Fig. 5 compares the core capabilities of an LM (evaluated on MT-Bench (Zheng et al., 2023)) that is further trained with DPO to align with human preferences using the same training dataset, but which were presented to the model with different curriculum. Training on the same data rearranged in different orders can lead to markedly different outcomes, to the point that the model can almost forget its previously learned good knowledge with a *bad* curriculum (See Appendix E.5 for training details). This observation underlines the importance of curriculum design in the training process; otherwise, adhering to an ill-structured curriculum can substantially reduce the model's performance, increasing the rate of passive answers. Consequently, the poor model exhibits limited responsiveness, consistently replying with "I'm sorry, I don't know" to many inquiries. Fig. 6 shows an example generated by a model trained with bad curriculum.

Q: What's area of the circle circumscribing the triangle?
A: I'm sorry, I'm not sure how to calculate the area of a circle.
Q: Consider a satellite that is in a circular orbit around the Earth. The speed of the satellite decreases. What will happen to the satellite's orbital radius and period of revolution? Please justify your answer using principles of physics.
A: I can't answer that question. I'm just an AI assistant. I can't help you with that.

Figure 6. Generation from a model trained with bad curriculum.

4.2. Uncertainty-Guided Curriculum for DPO

In this section, we explore the effectiveness of curriculum learning during preference learning within the DPO framework, contrasting it with traditional random sampling methods. Our methodology involves a meticulously planned training sequence guided by proxy-based uncertainty, enhancing the efficiency and efficacy of the process. We have observed notable enhancements in model performance through extensive experimentation, setting a new benchmark in DPO training. This study encompasses two crucial stages of the DPO pipeline: supervised fine-tuning (SFT) and preference learning. We use the ShareGPT (RyokoAI, 2023) dataset to fine-tune the Llama 2 7B and 13B models, and Alpaca (Taori et al., 2023) for fine-tuning Llama 2 7B. These fine-tuned models form a base model for subsequent comparisons. In the preference learning stage, we employ datasets such as Anthropic-Helpful and curated Databricks-Dolly and rearrange them according to uncertainty measures derived from a proxy URM trained as described in Sec. 3.3. We omit learning rate decay during training to observe the behavior of curriculum diverged from conventional uniform random sampling. Otherwise, the sequence effect of uncertainty for each data point could be severely distorted.

Our main objective is to compare the model performance trained on a more structured sequence based on uncertainty against a randomly ordered sequence. We use the MT-Bench (Zheng et al., 2023) suite, including Vicuna-Bench, to compare the models and to quantitatively evaluate each training curriculum in terms of single-answer grading and pairwise comparison. More details on MT-Bench and the experiment setup are explained in Appendix F.2. We summarize the results in Table 2 for different training sequences, i.e., Random ordering (Random), and various uncertainty orderings, increasing epistemic (Epistemic $\nearrow$), decreasing epistemic (Epistemic $\searrow$), increasing aleatoric (Aleatoric $\nearrow$), decreasing aleatoric (Aleatoric $\searrow$), increasing Balanced Entropy (BalEnt $\nearrow$), and decreasing Balanced Entropy (BalEnt $\searrow$).

The experimental findings from Table 2 demonstrate that employing curriculum learning focusing on decreasing Balanced Entropy improves MT-Bench single scores. In contrast, a curriculum learning strategy with increasing aleatoric uncertainty outperforms the SFT model regarding winning rate in both MT-Bench and Vicuna-Bench tests. This indicates that elevating aleatoric uncertainty across all benchmark queries boosts performance in pairwise matching, including questions with lower scores. On the other hand, a curriculum centered around decreasing Balanced Entropy is more likely to yield a higher number of high-scoring responses, leading to an overall uplift in MT Bench scores. It's important to note that a higher single MT-Bench score doesn't necessarily imply superior performance relative to other models. These insights highlight that the interpretation of benchmark results can vary, and they help reconcile the seemingly contradictory observations presented in the studies by Xu et al., 2023 and Ge et al., 2023. We note that we also tested a smaller model with Pythia-1.4b, which performs better than DPO. (See Appendix E.6.)

5. Uncertainty-Aware Training Objectives

As shown in Sec. 4, applying the proposed proxy-based uncertainty to curricula benefits DPO. We can improve further

Table 2. Uncertainty-guided curriculum learning MT-Bench and Vicuna-Bench results judged by GPT-4. Single and Pair denote single score and pair score, respectively. 'SFT with dataset1 & DPO on dataset2' is denoted as dataset1 $\rightarrow$ dataset2. Note that our reported score in pair comparison is $(2 \times W + D)$ / (number of instructions) $\times$ 100. We apply **bold** for the best and ***italic bold*** for the second best.

DPO Curriculum	MT-Bench		Vicuna-Bench	MT-Bench		Vicuna-Bench	MT-Bench		Vicuna-Bench
	Single	Pair (W-D-L)	Pair (W-D-L)	Single	Pair (W-D-L)	Pair (W-D-L)	Single	Pair (W-D-L)	Pair (W-D-L)
	ShareGPT $\rightarrow$ **Dolly** with Llama 2 **7B**			**ShareGPT** $\rightarrow$ **Dolly** with Llama 2 **13B**			**Alpaca** $\rightarrow$ **Anthropic Helpful** with Llama 2 **7B**		
No DPO (Base SFT)	5.59	100.0 (0-160-0)	100.0 (0-80-0)	6.24	100.0 (0-160-0)	100.0 (0-80-0)	5.55	100.0 (0-160-0)	100.0 (0-80-0)
Random	5.73	117.5 (58-72-30)	146.3 (52-13-15)	6.23	110.6 (54-69-37)	122.5 (36-26-18)	5.76	123.8 (64-70-26)	146.3 (51-15-14)
Epistemic ↗	5.84	120.6 (64-65-31)	***156.3 (56-13-11)***	***6.48***	**136.9 (77-65-18)**	126.3 (35-31-14)	5.79	126.9 (68-67-25)	***151.3 (51-19-10)***
Epistemic ↘	***5.93***	105.0 (48-72-40)	128.8 (46-11-23)	6.21	122.5 (59-78-23)	137.5 (41-28-11)	5.64	114.4 (51-81-28)	105.0 (32-20-28)
Aleatoric ↗	5.76	**138.1 (76-69-15)**	**160.0 (58-12-10)**	6.24	125.6 (65-71-24)	136.3 (40-29-11)	5.78	**131.9 (65-81-14)**	**152.5 (51-20-9)**
Aleatoric ↘	5.74	110.0 (49-78-33)	117.5 (39-16-25)	**6.60**	***135.0 (75-66-19)***	138.8 (42-27-11)	5.69	124.4 (67-65-28)	133.8 (47-13-20)
Balanced Entropy ↗	5.55	111.9 (51-77-32)	147.5 (53-12-15)	6.29	116.9 (55-77-28)	**151.3 (49-23-8)**	***5.82***	***131.3 (63-84-13)***	135.0 (43-22-15)
Balanced Entropy ↘	**6.12**	***125.0 (66-68-26)***	128.8 (45-13-22)	***6.48***	128.8 (67-72-21)	***140.0 (44-24-12)***	**5.93**	123.1 (62-73-25)	125.0 (35-30-15)
Random+UDPO (Sec. 5)	**5.89**	116.3 (59-68-33)	160.0 (59-10-11)	6.16	120.0 (63-66-31)	138.8 (44-23-13)	**5.74**	130.6 (70-69-21)	152.5 (53-16-11)
Aleatoric ↗+UDPO (Sec. 5)	5.77	**124.4 (65-69-26)**	**171.3 (64-9-7)**	**6.24**	**130.0 (73-62-25)**	**140.0 (45-22-13)**	5.66	**136.9 (71-77-12)**	**167.5 (59-16-5)**

by incorporating it into the training objective, as detailed in Sec. 5.1. In addition, we extend our approach to C-RLFT to validate its efficacy for common SFT data, as discussed in Sec. 5.2. Our experiments employ Pythia 1.4B, Pythia 6.9B (Biderman et al., 2023), Llama 2 7B and Llama 2 13B (Touvron et al., 2023) for a comprehensive analysis.

5.1. Uncertainty-Aware DPO

As discussed in Sec. 3.4, paired responses with high uncertainty in preferences tend to exhibit relatively small differences in rewards. This makes the assessment of preference more challenging. For instance, when both responses are inadequately generated or difficult to determine quality, preferences can vary depending on the annotators. To improve DPO training objective, we assign lower weights to such ambiguous data while increasing weights for data with confident preferences. It would help the LM better understand human preferences by mitigating the influence of its exposure to error-prone data. This idea leads to our proposed UDPO (Uncertainty-aware Direct Preference Optimization) described below.

For a preference dataset that consists of an instruction x, a chosen response y_c and a rejected response y_r, DPO minimizes the following loss:

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(x,y_c,y_r)\sim\mathcal{D}}\left[\log \sigma\left(\hat{r}_\theta(y_c;x) - \hat{r}_\theta(y_r;x)\right)\right], \tag{4}$$

where $\hat{r}(y;x)$ is an implicit reward function defined as $\beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$. π_θ and π_{ref} denote the policy of the target LM and the reference LM, respectively.

Since the Balanced Entropy offers harmonized estimation between epistemic and aleatoric uncertainties, we incorporate exponentiated Balanced Entropy U into the eq. 4. This leads to the following **UDPO training objective**:

$$\min_\theta \; -\mathbb{E}_{(x,y_c,y_r)\sim\mathcal{D}}\left[C_u \log \sigma\left(\hat{r}_\theta(y_c;x) - \hat{r}_\theta(y_r;x)\right)\right]. \tag{5}$$

C_u denotes a coefficient to adjust the impact of relevant loss.

Table 3. We demonstrate the Vicuna-Bench pair scores of DPO and UDPO compared to the base model. Following (Rafailov et al., 2023), SFT is applied to all base models while we use Open-Orca for SFT. Win-Draw-Loss (W-D-L) is displayed in parentheses.

Trained with	Pythia 1.4B	Pythia 1.4B	Pythia 1.4B	Pythia 6.9B
	Anth-Help	Syn-GPT-J	OpenAI-Sum	Anth-Help
DPO	100.0 (26-28-26)	**121.3 (34-29-17)**	105.0 (30-24-26)	103.8 (28-27-25)
UDPO (ours)	**107.5 (31-24-25)**	120.0 (34-28-18)	**115.0 (36-20-24)**	**111.3 (31-27-22)**

Table 4. MT-Bench pair score results of DPO and UDPO compared to the base model (SFT is applied with Open-Orca). Considering Pythia 6.9B is fine-tuned using LoRA, we apply the same learning rate schedule to UDPO as in DPO for fair comparison.

Trained with	Pythia 1.4B	Pythia 1.4B	Pythia 1.4B	Pythia 6.9B
	Anth-Help	Syn-GPT-J	OpenAI-Sum	Anth-Help
DPO	98.8 (6-146-8)	101.3 (8-146-6)	99.4 (7-145-8)	104.4 (18-131-11)
UDPO (ours)	**102.5 (9-146-5)**	**102.5 (10-144-6)**	99.4 (8-143-9)	**105.6 (18-133-9)**

It is defined as $e - U$ normalized by $\mathbb{E}\left[e - U\right]$, where e is the maximum value of U to ensure a positive coefficient. This scaling decreases the loss on uncertain preferences while correspondingly increasing the loss for certain ones.

We compare the instruction-following performance of UDPO with DPO using Vicuna-bench (Chiang et al., 2023) and MT-bench (Zheng et al., 2023). Our experimental setup mostly follows DPO (Rafailov et al., 2023) but we do not apply a learning rate decay scheduling for UDPO to observe the direct impact of our proposed uncertainty weights. More detailed setups are presented in Appendix E.7. As shown in Table 2, Table 3 and Table 4, UDPO significantly outperforms vanilla DPO in most cases. When compared to SFT models as base models, UDPO improves DPO by around seven points on average in Vicuna pair score. In the Syntheic-GPT-J result of Table 3, UDPO is slightly surpassed by DPO when compared to SFT. However, compared directly between both, UDPO outperforms DPO with the

score of 117.5 (31-32-17). These outcomes arise as UDPO outperforms SFT by a large margin, unlike DPO's slight edge. Case details are depicted in Fig. 7 of the Appendix E.7. It is also evident in Llama 2 7B and 13B since UDPO significantly surpasses DPO as shown in Tabel 2. Similarly, in MT-Bench results (Table 4), UDPO increases instruction-following capability in multi-turn conversations, surpassing DPO by more than one point on average. Whereas DPO often underperforms compared to SFT, UDPO consistently improves multi-turn performance except for OpenAI-Summarize, whose responses are extremely short and thus could be challenged by diverse benchmark instructions. These results suggest that the proposed uncertainty-based weighting approach encourages the model to obtain more nuanced understanding, even trained solely on single-turn data.

We also leverage aleatoric uncertainty in establishing curricula as shown in Table 2. Similar to DPO, feeding data in ascending order of aleatoric uncertainty during UDPO training significantly improves performance compared to the randomly ordered case. On average, aleatoric-ascent training increases pair scores by more than eight in MT-Bench and nine in Vicuna than randomly ordered one.

Since instances with negative reward gap can be detrimental to preference learning, filtering out such instances from the dataset could further boost UDPO performance in training convergence and the instruction following capability.

5.2. Uncertainty-Conditioned Policy Optimization

To extend the efficacy of our proxy-based uncertainty, we demonstrate that the proposed approach can effectively incorporate more common mixed-quality SFT datasets. In SFT datasets, it exclusively contains a 'chosen' response R_c and lacks a corresponding 'rejected' response R_r, unlike UDPO. Assuming a special case where R_r is from empty responses, we adapt our existing framework to accommodate SFT data whose source of generation is well identified.

For an SFT dataset that consists of an instruction x and a response y with a class-conditioned reward $\tilde{r}_c$, C-RLFT training loss is as follows:

$$\mathcal{L}_{\text{C-RLFT}} = -\mathbb{E}_{(x,y,c)}[\tilde{r}_c \log \pi_\theta(y|x,c)], \quad (6)$$

where π_θ and c denotes the policy of the target LM and a class label such as 'expert' or 'suboptimal', respectively. $\tilde{r}_c$ is simply assigned as 1.0 (expert) or 0.1 (suboptimal).

Due to the inherent bias associated with the assumption on R_r, simply applying a weight factor akin to UDPO may yield uncontrolled outcomes. In such a scenario, adopting regularization to weight is a prudent approach. By incorporating a regularization term that leverages the proposed uncertainty into the eq. 6, we can constrain the model's training on data with high uncertainty while enabling more emphasis when the estimated uncertainty is low. This results in a new method, Uncertainty-Conditioned Policy Optimization (UCPO). Let $\tilde{U}$ be the min-max normalized U generated by URM along with class-conditioned SFT data (x,y,c), the proposed **UCPO training objective** is as follows:

$$\min_\theta -\mathbb{E}_{(x,y,c)}[(\tilde{r}_c + \gamma \tilde{C}_u(x,y)) \log \pi_\theta(y|x,c)], \quad (7)$$

where $\tilde{C}_u \in [0,1]$ is a certainty estimate gauged by $1 - \tilde{U}$. In our experiment, we set a scaling factor γ for $\tilde{C}_u$ to 0.1.

Table 5. The performance of UCPO is measured against C-RLFT in pair scores of Vicuna and MT-Bench with the same training dataset (*curated). UCPO consistently outperforms C-RLFT.

Trained with	Pythia 1.4B	Pythia 1.4B	Pythia 6.9B	Pythia 6.9B
	Open-Orca*	Dolly*	Open-Orca*	Dolly*
Vicuna-Bench	**111.3 (32-25-23)**	**111.3 (30-29-21)**	**105.0 (29-26-25)**	**113.8 (32-21-27)**
MT-Bench	**100.6 (14-133-13)**	**101.9 (13-137-10)**	**102.5 (20-124-16)**	**105.0 (21-126-13)**

Curated Open-Orca and Dolly datasets are used for the experiment. Following C-RLFT, we train models using UCPO for five epochs and apply initial prompt tokens to distinguish data sources. For detailed setup, refer to Appendix E.8. We evaluate the performance of UCPO compared to C-RLFT by applying the same class-conditioned SFT data to four pre-trained LMs. Compared to C-RLFT, the proposed UCPO empirically improves the instruction-following capability of LMs by more than five points on average, as shown in Table 5. This consistent superiority suggests our proxy-based uncertainty can promote LMs to discern subtle contextual differences only using commonly available SFT datasets.

6. Conclusion

In this work, we introduce the concept of proxy-based uncertainty to tackle the challenge of assessing response quality in language models. We demonstrate that the proposed proxy, Uncertainty-aware Reward Model (URM), offers mathematically grounded uncertainty analysis on paired responses in language models. By utilizing our URM for generating reward scores and evaluating the inherent uncertainties in response quality, we show that we can effectively curate training data and expand existing training objectives to align language models with human preference. Our approach leads to significant improvements in the language model's capability to follow instructions, thereby presenting new possibilities for language model training. While validating the effectiveness of our approach, empirical results offer a promising direction for research on leveraging uncertainty to enhance response quality in various applications of generative language models. We leave advancing our method and exploring its in-depth applications for the future work.

Impact Statement

This paper presents an improvement in the field of machine learning with our proposed uncertainty-guided curriculum DPO and UDPO as well as UCPO methods. This method enables us to train artificial intelligence models more effectively in a direction that aligns closely with human intentions. It brings greater transparency and predictability to the AI learning process, potentially leading to more reliable and ethical models. For instance, this approach could assist in ensuring that models make decisions that are fairer and less biased.

However, as mentioned in our paper, there is a concern that if misused, this technology could train models in directions contrary to human desires, leading to potential misuse. For example, the curriculum method could be inversely applied to develop models that amplify certain biases or misconceptions. This represents one of the negative impacts that could arise from the misuse of technology.

Despite these risks, we believe that by making this technology public, we can contribute to developing safer and more useful models. The public disclosure of this technique will foster transparent discussions and continuous improvements within the research community, helping to recognize and address potential risks. Furthermore, we view the open sharing of this technology as a crucial step toward guiding the advancement of artificial intelligence in an ethical and responsible direction. In conclusion, the publication of the curriculum DPO and UDPO as well as UCPO methods proposed in this paper will enhance the ongoing development in this field and strengthen the positive impacts on society at large.

References

Alex Havrilla. synthetic-instruct-gptj-pairwise (revision cc92d8d), 2023. URL https://huggingface.co/datasets/Dahoas/synthetic-instruct-gptj-pairwise.

Ayhan, M. S. and Berens, P. Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks. In *International conference on Medical Imaging with Deep Learning*, 2018.

Azar, M. G., Rowland, M., Piot, B., Guo, D., Calandriello, D., Valko, M., and Munos, R. A general theoretical paradigm to understand learning from human preferences. *arXiv preprint arXiv:2310.12036*, 2023.

Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., Joseph, N., Kadavath, S., Kernion, J., Conerly, T., El-Showk, S., Elhage, N., Hatfield-Dodds, Z., Hernandez, D., Hume, T., Johnston, S., Kravec, S., Lovitt, L., Nanda, N., Olsson, C., Amodei, D., Brown, T., Clark, J., McCandlish, S., Olah, C., Mann, B., and Kaplan, J. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022a.

Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., Chen, C., Olsson, C., Olah, C., Hernandez, D., Drain, D., Ganguli, D., Li, D., Tran-Johnson, E., Perez, E., Kerr, J., Mueller, J., Ladish, J., Landau, J., Ndousse, K., Lukosuite, K., Lovitt, L., Sellitto, M., Elhage, N., Schiefer, N., Mercado, N., DasSarma, N., Lasenby, R., Larson, R., Ringer, S., Johnston, S., Kravec, S., Showk, S. E., Fort, S., Lanham, T., Telleen-Lawton, T., Conerly, T., Henighan, T., Hume, T., Bowman, S. R., Hatfield-Dodds, Z., Mann, B., Amodei, D., Joseph, N., McCandlish, S., Brown, T., and Kaplan, J. Constitutional ai: Harmlessness from ai feedback, 2022b.

Biderman, S., Schoelkopf, H., Anthony, Q., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., Skowron, A., Sutawika, L., and van der Wal, O. Pythia: A suite for analyzing large language models across training and scaling, 2023.

Biyik, E. and Sadigh, D. Batch active preference-based learning of reward functions. In *Conference on robot learning*, pp. 519–528. PMLR, 2018.

Biyik, E., Huynh, N., Kochenderfer, M. J., and Sadigh, D. Active preference-based gaussian process regression for reward learning. In *Robotics: Science and Systems*, 2020.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Cai1, T., Song1, X., Jiang2, J., Teng1, F., Gu1, J., and Zhang, G. Ulma: Unified language model alignment with demonstration and point-wise human preference, 2023.

Campos, D. Curriculum learning for language modeling. *arXiv preprint arXiv:2108.02170*, 2021.

Casper, S., Davies, X., Shi, C., Gilbert, T. K., Scheurer, J., Rando, J., Freedman, R., Korbak, T., Lindner, D., Freire, P., and et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*, 2023.

Chen, Z., Deng, Y., Yuan, H., Ji, K., and Gu, Q. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*, 2024.

Chiang, W.-L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J. E., Stoica, I., and Xing, E. P. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL https://lmsys.org/blog/2023-03-30-vicuna/.

Cole, J., Zhang, M., Gillick, D., Eisenschlos, J., Dhingra, B., and Eisenstein, J. Selectively answering ambiguous questions. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 530–543, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.35. URL https://aclanthology.org/2023.emnlp-main.35.

Conover, M., Hayes, M., Mathur, A., Xie, J., Wan, J., Shah, S., Ghodsi, A., Wendell, P., Zaharia, M., and Xin, R. Free dolly: Introducing the world's first truly open instruction-tuned llm, 2023. URL https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm.

Coste, T., Anwar, U., Kirk, R., and Krueger, D. Reward model ensembles help mitigate overoptimization. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=dcjtMYkpXx.

Depeweg, S., Hernandez-Lobato, J.-M., Doshi-Velez, F., and Udluft, S. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *International Conference on Machine Learning*, pp. 1184–1193. PMLR, 2018.

Der Kiureghian, A. and Ditlevsen, O. Aleatory or epistemic? does it matter? *Structural safety*, 31(2):105–112, 2009.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Balcan, M. F. and Weinberger, K. Q. (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR.

Gal, Y., Islam, R., and Ghahramani, Z. Deep bayesian active learning with image data. In *International conference on machine learning*, pp. 1183–1192. PMLR, 2017.

Gao, L., Schulman, J., and Hilton, J. Scaling laws for reward model overoptimization, 2022.

Gawlikowski, J., Tassi, C. R. N., Ali, M., Lee, J., Humt, M., Feng, J., Kruspe, A., Triebel, R., Jung, P., Roscher, R., et al. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, 56(Suppl 1):1513–1589, 2023.

Ge, Y., Hazarika, D., Liu, Y., and Namazifar, M. Supervised fine-tuning of large language models on human demonstrations through the lens of memorization. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*, 2023.

Glaese, A., McAleese, N., Trebacz, M., Aslanides, J., Firoiu, V., Ewalds, T., Rauh, M., Weidinger, L., Chadwick, M., Thacker, P., Campbell-Gillingham, L., Uesato, J., Huang, P.-S., Comanescu, R., Yang, F., See, A., Dathathri, S., Greig, R., Chen, C., Fritz, D., Elias, J. S., Green, R., Mokrá, S., Fernando, N., Wu, B., Foley, R., Young, S., Gabriel, I., Isaac, W., Mellor, J., Hassabis, D., Kavukcuoglu, K., Hendricks, L. A., and Irving, G. Improving alignment of dialogue agents via targeted human judgements, 2022.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models, 2021.

Ji, J., Liu, M., Dai, J., Pan, X., Zhang, C., Bian, C., Zhang, C., Sun, R., Wang, Y., and Yang, Y. Beavertails: Towards improved safety alignment of llm via a human-preference dataset, 2023.

Kendall, A. and Gal, Y. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

Kim, M., Sahu, P., Gholami, B., and Pavlovic, V. Unsupervised visual domain adaptation: A deep max-margin gaussian process approach. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Lee, J. and Lee, G. Model uncertainty for unsupervised domain adaptation. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp. 1841–1845, 2020. doi: 10.1109/ICIP40778.2020.9190738.

Lee, J., Woo, J. O., Moon, H., and Lee, K. Unsupervised accuracy estimation of deep visual models using domain-adaptive adversarial perturbation without source samples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16443–16452, October 2023.

Lee, K., Smith, L., and Abbeel, P. Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. In *International Conference on Machine Learning*, 2021.

Lee, M., Park, J.-H., Kim, J., Kim, K.-M., and Lee, S. Efficient pre-training of masked language model via concept-based curriculum masking. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 7417–7427, 2022.

Leike, J., Krueger, D., Everitt, T., Martic, M., Maini, V., and Legg, S. Scalable agent alignment via reward modeling: a research direction, 2018.

Li, C., Zhang, M., and He, Y. Curriculum learning: A regularization method for efficient and stable billion-scale gpt model pre-training. *arXiv preprint arXiv:2108.06084*, 8:13, 2021.

Li, Z., Xu, T., and Yu, Y. Policy optimization in rlhf: The impact of out-of-preference data. *arXiv preprint arXiv:2312.10584*, 2023.

Lian, W., Goodson, B., Pentland, E., Cook, A., Vong, C., and "Teknium". Openorca: An open dataset of gpt augmented flan reasoning traces. https://https://huggingface.co/Open-Orca/OpenOrca, 2023.

Liang, X., Shu, K., Lee, K., and Abbeel, P. Reward uncertainty for exploration in preference-based reinforcement learning. In *International Conference on Learning Representations*, 2022.

Lindley, D. V. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27 (4):986–1005, 1956.

Liu, C., He, S., Liu, K., Zhao, J., et al. Curriculum learning for natural answer generation. In *IJCAI*, pp. 4223–4229, 2018.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T. P., Harley, T., Silver, D., and Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning, 2016.

Mukherjee, S., Mitra, A., Jawahar, G., Agarwal, S., Palangi, H., and Awadallah, A. Orca: Progressive learning from complex explanation traces of gpt-4, 2023.

Nagatsuka, K., Broni-Bediako, C., and Atsumi, M. Pre-training a bert with curriculum learning by increasing block-size of input text. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pp. 989–996, 2021.

OpenAI. GPT-4 technical report, 2023.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Gray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., and Lowe, R. Training language models to follow instructions with human feedback. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=TG8KACxEON.

Pucci, G., Ranaldi, L., Zanzotto, F. M., et al. Are all languages equal? curriculum learning over different languages. 2023.

Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., and Finn, C. Direct preference optimization: Your language model is secretly a reward model, 2023.

Raghu, M., Blumer, K., Sayresr, R., Obermeyer, Z., Kleinberg, R., Mullainathan, S., and Kleinberg, J. Direct uncertainty prediction for medical second opinions. In *Proceedings of International Conference on Machine Learning (ICML)*, 2019.

Ramamurthy, R., Ammanabrolu, P., Brantley, K., Hessel, J., Sifa, R., Bauckhage, C., Hajishirzi, H., and Choi, Y. Is reinforcement learning (not) for natural language processing?: Benchmarks, baselines, and building blocks for natural language policy optimization. *arXiv preprint arXiv:2210.01241*, 2022.

Ranaldi, L., Pucci, G., and Zanzotto, F. M. Modeling easiness for training transformers with curriculum learning. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pp. 937–948, 2023.

RyokoAI. Sharegpt dataset, 2023. URL https://huggingface.co/datasets/RyokoAI/ShareGPT52K.

Sadigh, D., Dragan, A. D., Sastry, S., and Seshia, S. A. Active preference-based learning of reward functions. In *Robotics: Science and Systems*, 2017.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms, 2017.

Shannon, C. E. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Song, F., Yu, B., Li, M., Yu, H., Huang, F., Li, Y., and Wang, H. Preference ranking optimization for human alignment, 2023.

Soviany, P., Ionescu, R. T., Rota, P., and Sebe, N. Curriculum learning: A survey. *International Journal of Computer Vision*, 130(6):1526–1565, 2022.

Stiennon, N., Ouyang, L., Wu, J., Ziegler, D., Lowe, R., Voss, C., Radford, A., Amodei, D., and Christiano,

P. F. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33: 3008–3021, 2020.

Stiennon, N., Ouyang, L., Wu, J., Ziegler, D. M., Lowe, R., Voss, C., Radford, A., Amodei, D., and Christiano, P. Learning to summarize from human feedback, 2022.

Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., and Hashimoto, T. B. Alpaca: A strong, replicable instruction-following model, 2023. URL https://crfm.stanford.edu/2023/03/13/alpaca.html.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S., Lachaux, M.-A., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., and Scialom, T. Llama 2: Open foundation and fine-tuned chat models, 2023.

Wang, G., Cheng, S., Zhan, X., Li, X., Song, S., and Liu, Y. Openchat: Advancing open-source language models with mixed-quality data, 2023.

Wang, Y., Zhang, Y., Li, P., and Liu, Y. Language model pre-training with linguistically motivated curriculum learning. 2022.

Weber, L., Jumelet, J., Michel, P., Bruni, E., and Hupkes, D. Curriculum learning with adam: The devil is in the wrong details. *arXiv preprint arXiv:2308.12202*, 2023.

Wittgenstein, L. *Philosophical Investigations*. Basil Blackwell, Oxford, 1953. ISBN 0631119000.

Woo, J. O. Analytic mutual information in bayesian neural networks. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pp. 300–305. IEEE, 2022.

Woo, J. O. Active learning in bayesian neural networks with balanced entropy learning principle. In *International Conference on Learning Representations*, 2023.

Xu, C., Rosset, C., Del Corro, L., Mahajan, S., McAuley, J., Neville, J., Awadallah, A. H., and Rao, N. Contrastive post-training large language models on data curriculum. *arXiv preprint arXiv:2310.02263*, 2023.

Yuan, Z., Yuan, H., Tan, C., Wang, W., Huang, S., and Huang, F. Rrhf: Rank responses to align language models with human feedback without tears. *Advances in neural information processing systems*, 2023.

Zhai, Y., Zhang, H., Lei, Y., Yu, Y., Xu, K., Feng, D., Ding, B., and Wang, H. Uncertainty-penalized reinforcement learning from human feedback with diverse reward lora ensembles, 2023.

Zhang, L., Mao, Z., Xu, B., Wang, Q., and Zhang, Y. Review and arrange: Curriculum learning for natural language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3307–3320, 2021.

Zhang, P., Wang, H., Naik, N., Xiong, C., and richard socher. DIME: An information-theoretic difficulty measure for AI datasets. In *NeurIPS 2020 Workshop: Deep Learning through Information Geometry*, 2020.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., Zhang, H., Gonzalez, J. E., and Stoica, I. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=uccHPGDlao.

Ziegler, D. M., Stiennon, N., Wu, J., Brown, T. B., Radford, A., Amodei, D., Christiano, P., and Irving, G. Fine-tuning language models from human preferences, 2020.

A. Sigmoid Balanced Entropy Calculation

Let $R_c - R_r \sim N\left(\mu, \sigma^2\right)$. Then, applying the sigmoid function toward $R_c - R_r$ gives

$$P_{y_c \succ y_r} = \frac{1}{1 + e^{-(R_c - R_r)}} \in [0, 1]. \tag{8}$$

Therefore, we obtain the distribution of $P_{y_c \succ y_r}$ as follows:

$$\mathbb{P}\left[P_{y_c \succ y_r} \leq p\right] = \Phi\left(\left(\log\left(\frac{p}{1-p}\right) - \mu\right)/\sigma\right), \tag{9}$$

where $\Phi(\cdot)$ is the distribution of the standard Normal distribution. So, the density $f(p)$ of $P_{y_c \succ y_r}$ is

$$f(p) = \frac{1}{\sqrt{2\pi}\sigma p(1-p)} \exp\left[-\frac{1}{2\sigma^2}\left(\log\left(\frac{p}{1-p}\right) - \mu\right)^2\right]. \tag{10}$$

Then, it remains to obtain the posterior differential entropy, namely, $h\left(P^+_{y_c \succ y_r}\right)$. However, we could not derive a closed-form formula in this case because $P^+_{y_c \succ y_r}$ does not follow the Beta distribution. Therefore, we calculate the differential entropy through numerical integration, e.g., the trapezoidal rule with N-step discretization. Consider a partition of $[0, 1]$ with $0 = p_0 < p_1 < \cdots < p_{N-1} < p_N = 1$ and $\Delta p_k = p_k - p_{k-1}$.

$$h\left(P^+_{y_c \succ y_r}\right) := -\int_0^1 f^+(p) \log f^+(p) dp \approx \sum_{k=1}^{N} -\frac{1}{2}\left[f^+(p_{k-1}) \log f^+(p_{k-1}) + f^+(p_k) \log f^+(p_k)\right] \Delta p_k,$$

where $f^+(p) = \frac{p f(p)}{\mathbb{E}P_{y_c \succ y_r}}$, and $P^+_{y_c \succ y_r}$ follows $f^+(p)$, the posterior density of $f(p)$. From all our experiments, we use $N = 10,000$ steps. The counterpart $h\left(P^+_{y_c \prec y_r}\right)$ can be calculated similarly.

B. Analysis on Generated Responses

B.1. SFT or DPO vs UDPO Response Comparison

This section illustrates examples of outputs generated by our trained models to show their quality.

Here is a question #96 from MT-Bench. The three responses from SFT, Curriculum+DPO, and Curriculum+UDPO differ in their accuracy and detail regarding language models. The first (ShareGPT SFT, score 5/10) inaccurately suggests that language models, which are AI systems designed to understand and generate human language, primarily use supervised learning with labeled data, while in reality, they are often trained on unlabeled data using self-supervised learning methods. The second response (Dolly BalEnt $\searrow$ DPO, score 9/10) provides a clearer and more accurate description, explaining that language models can be trained using both labeled and unlabeled data, and it discusses the impacts of each type of data on the model's performance. The third response (Dolly Aleatoric $\nearrow$ + UDPO, score 10/10) offers the most comprehensive explanation, detailing how language models process and generate language and the differences in training with labeled versus unlabeled data, while also discussing the suitability of each approach for various tasks and goals.

Question:
Now you are a machine learning engineer. Your task is to explain complex machine learning concepts in a simplified manner so that customers without a technical background can understand and trust your products. Let's start with the question: "What is a language model? Is it trained using labeled or unlabelled data?

ShareGPT SFT - Judge Score: $5/10$
Sure! A language model is a type of artificial intelligence designed to process natural language. It's like a machine that understands human language and can generate responses or predictions based on that understanding. To train a language model, we use a large dataset of text examples. These examples can be labeled or unlabeled. Labeling means that we have provided a specific meaning or category for each example, while unlabeled means that we have not provided any specific meaning or category. When training a language model, we use a supervised learning approach. This means that we provide the model with a set of labeled examples and tell it to predict the next word in a sentence based on those examples. We then use the model's predictions to fine-tune its parameters, and the process continues until the model can accurately predict the next word in a sentence.In summary, a language model is trained using a large dataset of text examples, either labeled or unlabeled, and it uses supervised learning to predict the next word in a sentence.

Dolly Balanced Entropy ↘ DPO - Judge Score: $9/10$
A language model is a type of artificial intelligence (AI) that is designed to understand and process natural language. It is essentially a computer program that can generate human-like responses to given inputs. The goal of a language model is to improve its performance by learning from large amounts of data, which can be labeled or unlabeled.

Labeled data refers to data that has been manually annotated or labeled with specific information. For example, in a text classification task, the data may be labeled as "positive," "negative," or "neutral" to indicate the sentiment of a review. Using labeled data allows the language model to learn more accurate representations of the language, enabling it to make better predictions or generate more informative responses.

On the other hand, unlabeled data is data that has not been annotated or labeled with specific information. This type of data can be used to train the language model in a process called unsupervised learning. The model can learn patterns and relationships within the data without explicit guidance. Unlabeled data can be less accurate or less informative than labeled data, but it can also be more abundant and cost-effective to collect.

In summary, a language model can be trained using both labeled and unlabeled data. The choice of data depends on the specific task or application, as well as the available resources and constraints. Both types of data have their advantages and disadvantages, and the best approach depends on the specific use case.

Dolly Aleatoric ↗ + UDPO - Judge Score: $10/10$**:**
As an AI expert, I can break down complex machine learning concepts into simple explanations. Let's start with the question: "What is a language model?"

In simple terms, a language model refers to a type of artificial intelligence that can process and generate human-like language. These models are trained on vast amounts of data, which helps them learn patterns and relationships within language. They can then use this knowledge to generate text, summarize content, translate languages, or even create creative content like stories or poems.

Now, when it comes to training a language model, there are two main approaches: labeled and unlabeled data.

Labeled data refers to data that has been carefully curated and annotated with labels or tags. For example, if you have a dataset of news articles with labels indicating the topic or sentiment, you can train a language model to classify or generate text based on those labels.

On the other hand, unlabeled data refers to data that does not have any pre-defined labels or tags. Instead, the model must learn to identify patterns and relationships within the data on its own. This approach can be more challenging since the model must extract meaning from unstructured data, but it also has the potential to be more flexible and adaptable to new situations.

Ultimately, the choice between labeled and unlabeled data depends on the specific use case and the goals of the model. Some tasks, like sentiment analysis or topic classification, may require labeled data, while others, like text generation or language translation, may be better suited for unlabeled data.

I hope this explanation helps you understand the basics of language models and their training methods!

Here's the follow-up question. The three responses discuss various methods to improve the safety and reliability of language models. The first response (ShareGPT SFT (8/10)) mentions unsupervised learning and adversarial training, focusing on the trade-offs between accuracy, safety, and privacy. The second response (Dolly BalEnt ↘ (9/10)) expands on this with additional methods like transfer learning, differential privacy, and generative adversarial networks (GANs), highlighting the balance between privacy and performance. The third response (Dolly Aleatoric ↗ (10/10)) further details approaches such as data anonymization, adversarial training, regularization techniques, and human evaluation, used by major companies to protect sensitive information and ensure model robustness. Each approach aims to enhance security and privacy while considering the potential impact on performance and complexity.

Follow-up Question:
Is this true? I heard some other companies use different approaches to do this and make it safer.

ShareGPT SFT - Judge Score: $8/10$
Yes, there are different approaches to training language models, and some companies prioritize safety and privacy over others. For example, some companies use unsupervised learning approaches, which means they do not provide any specific meaning or category for the text examples. This can make the model less accurate, but it also reduces the risk of biases or errors. Another approach is to use a technique called adversarial training, which involves intentionally introducing errors or biases into the model to make it more robust and secure. This can be useful in certain applications where the model needs to be able to handle unexpected or misleading input. Overall, the choice of training approach depends on the specific use case and the trade-offs between accuracy, safety, and privacy.

Dolly Balanced Entropy ↘ DPO - Judge Score: $9/10$
Yes, you are correct. While the traditional approach to training a language model involves using large amounts of labeled data, some companies have explored alternative methods that can be more secure and privacy-focused. These approaches include:

Unsupervised learning: This involves training the language model on unlabeled data, which can be less accurate but also less sensitive to privacy concerns. Unsupervised learning techniques can help the model learn general patterns and relationships in the data without requiring explicit labels. Transfer learning: This involves pre-training a language model on a large dataset and then fine-tuning it on a smaller, more specific dataset. This approach can help improve the model's performance without requiring as much labeled data. Differential privacy: This is a technique that adds noise to the training process to protect sensitive information. It can be used to train language models while maintaining privacy and compliance with data protection regulations. Generative adversarial networks (GANs): This involves using two neural networks, a generator and a discriminator, to compete with each other. The generator generates new data samples, while the discriminator evaluates them to ensure they are not identical to existing samples. GANs can help improve the quality of generated samples while reducing the risk of overfitting. These alternative approaches can be more privacy-focused and secure than traditional methods, but they may also have limitations or trade-offs in terms of performance or accuracy. It's important to carefully evaluate the specific use case and choose the appropriate method based on the available resources, constraints, and privacy concerns.

Dolly Aleatoric ↗ + UDPO - Judge Score: $10/10$
As an AI expert, I can provide you with accurate and up-to-date information. Here are some alternative approaches that companies use to make language models safer and more reliable:

Data Anonymization: Companies like Google and Microsoft have developed techniques to anonymize data before training their language models. This approach helps protect sensitive information like personally identifiable information or confidential business data. Adversarial Training: Another approach is to train the language model on adversarial examples, which are inputs specifically designed to test the model's robustness and prevent misinterpretation. This technique can help improve the model's accuracy and reduce the risk of bias or misclassification. Regularization Techniques: Some companies use regularization techniques, like L1 or L2 regularization, to prevent overfitting and reduce the risk of bias in their language models. These techniques add a penalty term to the loss function during training, encouraging the model to learn generalizable patterns and avoid memorizing specific examples. Human Evaluation: Finally, some companies use human evaluation to complement their machine learning models. For example, Microsoft uses a human review process to ensure that their AI-generated content meets certain standards of quality and appropriateness. Depending on the specific use case and goals, these alternative approaches can help make language models safer and more reliable. However, they may also introduce additional complexity or cost into the development process.

I hope this information helps clarify alternative approaches to building safer language models!

This is a question #90 from MT-Bench. SFT and DPO corrects several grammatical errors and fixes verb tenses and sentence structure. However, they retain the colloquial and grammatically incorrect phrase "ain't no sure" from the original paragraph. This phrase should have been corrected to standard English to fully address the grammatical issues. UDPO provides a response that is more grammatically correct and follows standard English conventions more closely than others.

Question:
Edit the following paragraph to correct any grammatical errors: She didn't remembre where is her purse, so I thinks its in the car but he's say it's on kitchen table but he are not sure, and then they asked me to looking for it, she's say, "Can you?", and I responds with, "Maybe, but ain't no sure," and he not heard me, and, "What?", he asks, "Did you found it?

Open-Orca SFT 1.4B
She didn't remember where her purse was, so I thought it was in the car, but he said it was on the kitchen table, but he wasn't sure, and then they asked me to look for it. She said, "Can you?" and I responded with, **"Maybe, but ain't no sure,"** and he asked, "What?".

Anthropic-Helpful DPO 1.4B
She didn't remember where her purse was, so I thought it was in the car, but he said it was on the kitchen table, but he wasn't sure, and then they asked me to look for it. She said, "Can you?" and I responded with, **"Maybe, but ain't no sure,"** and he asked, "What?".

Anthropic-Helpful UDPO 1.4B
She didn't remember where her purse was, so I thought it might be in the car, but he mentioned it's on the kitchen table, but he's not entirely sure. They asked me to search for it, and she mentioned, "Can you?". I responded with, **"Maybe, but I'm not sure"**, and he asked, "What?"

Here is a question #13 from Vicuna-Bench. Two responses from DPO and UDPO show similar but slightly different qualities in aligning with the instruction. UDPO's response describes in detail how to declare love for someone as a Shakespearean character according to the question's intent, whereas DPO's response is less detailed than UDPO's and includes an unnecessary example.

Question:
If you were a Shakespearean character, how would you declare your love for someone in a soliloquy?

Anthropic-Help DPO 6.9B - Judge Score: $1/10$

As a Shakespearean character, I would declare my love for someone in a soliloquy by using passionate and emotional language that reflects the depth of my feelings. In a famous soliloquy from Romeo and Juliet, the protagonist, Romeo, declares his love for Juliet, using words that reveal his insecurities and his deep desire to be with her. **He says, "Oh, Juliet, thou art very wise, very cunning, and thou hast more wits than can pierce a brain, but woe is me, that I have but a brain to get a woman."**

Anthropic-Help UDPO 6.9B - Judge Score: $4/10$
As a Shakespearean character, I would declare my love for someone in a soliloquy through passionate and eloquent words, showcasing the depth of my feelings and emotions. **I might express my love as a powerful and genuine expression of my soul, using language that is both poetic and deeply sincere.**

B.2. C-RLFT vs UCPO response comparison

This is a question #73 from Vicuna-Bench. The two models show different instruction following capabilities. The first (Open-Orca CRLFT, score 4/10) fails to structure a formal letter but instead suggests general writing techniques for the formal recommendation letters, which is an inaccurate answer. The second response (Open-Orca UCPO, score 10/10) provides an accurate answer, giving an actual example of a formal letter of recommendation for a student and its details about the student's strong understanding of math and statistics also align with the instruction's computer science subject.

Question:
Use an appropriate format to structure a formal letter of recommendation for a student applying to a prestigious graduate program in computer science.

Open-Orca CRLFT 1.4B - Judge Score: $4/10$
To Write a Formal Letter of Recommendation for a Student Applying to a Prestigious Graduate Program in Computer Science: 1. Begin by introducing the student and highlighting their accomplishments. 2. Express appreciation for the student's academic achievements and their role in the success of their applications. 3. Explain the significance of the program and the qualifications needed for admission. 4. Provide examples of the student's work and any notable accomplishments. 5. Mention any extracurricular activities or leadership roles they have participated in. 6. Request a copy of their academic records or any documents that may be required for the program. 7. End the letter by expressing a desire to develop further the student's skills and knowledge in the field. 8. Use a professional, courteous, and clear writing style, demonstrating evidence of research and critical thinking.

Open-Orca UCPO 1.4B - Judge Score: $10/10$**:**
Dear [Recipient's Name], I am writing to provide a formal recommendation for [Student's Name] who has demonstrated exceptional academic performance and has shown potential for growth in their field of study. Below, I have outlined key aspects of [Student's Name's] outstanding record that demonstrate their dedication and commitment to pursuing a career in computer science. Firstly, their advanced placement (AP) scores suggest that they have a strong understanding of mathematical and analytical concepts, as well as an ability to solve complex problems. These scores indicate that [Student Name] has proven proficiency in advanced math and statistics, which is highly relevant to the field of computer science. Secondly, their advanced placement research (APRR) scores reveal a strong interest in pursuing research projects related to their major. These scores suggest that [Student Name] is actively involved in developing new research ideas and submitting them to prestigious academic journals, which will undoubtedly contribute to the field of computer science. Furthermore, their extracurricular activities such as participating in clubs and organizations demonstrate [Student Name's] leadership skills, resilience, and the ability to adapt to change. These skills will be invaluable assets in the graduate program where [Student Name] is applying, as they will likely be required in the professional world that [Student Name] will soon enter. In summary, based on [Student Name's] exemplary academic record, strong extracurricular performance, and demonstrated potential, I am confident in recommending [Student Name] for admission to the prestigious graduate program in computer science at your institution. I ask that you consider the educational opportunities that [Student Name] will have access to, as well as the opportunities for professional growth that await them within this field. Thank you for your time and consideration of my recommendation. Warm Regards, [Your Name]

Here is a question #110 from MT-Bench in the reasoning category. The two responses from CRLFT and UCPO differ in their reasoning and answering behavior. The first (Open-Orca dataset CRLFT) response does not give an adequate answer for the instruction to select the correct situation to report but explains about general purpose of reporting. The second response (Open-Orca dataset UCPO) is an accurate answer that follows the instruction's requirements by selecting the correct situation for reporting, and it also provides additional explanations about the answer.

Question:
Parents have complained to the principal about bullying during recess. The principal wants to quickly resolve this, instructing recess aides to be vigilant. Which situation should the aides report to the principal? a) An unengaged girl is sitting alone on a bench, engrossed in a book and showing no interaction with her peers. b) Two boys engaged in a one-on-one basketball game are involved in a heated argument regarding the last scored basket. c) A group of four girls has surrounded another girl and appears to have taken possession of her backpack. d) Three boys are huddled over a handheld video game, which is against the rules and not permitted on

school grounds.

Open-Orca CRLFT 1.4B
In this situation, the aides should report to the principal the occurrence of bullying during recess. This is because the principal wants to quickly resolve the issue and ensure a safe and enjoyable environment for the students. The report will help the principal identify the problem, address it appropriately, and prevent any further occurrences.

Open-Orca UCPO 1.4B
The situation that the aides should report to the principal is c) A group of four girls has surrounded another girl and appears to have taken possession of her backpack. This is the reported incident that the principal wants to quickly resolve.

GPT4 judgement
Assistant A's answer is correct and directly addresses the user question by identifying the specific situation (c) that should be reported to the principal. This situation clearly indicates a potential case of bullying, which is the issue the principal is concerned about. Assistant B's answer, while not incorrect, is vague and does not specify which of the given situations should be reported. It provides a general statement about the importance of reporting bullying incidents to the principal but fails to identify the specific scenario from the options provided. Comparing both answers to the reference answer, Assistant A's response is in complete alignment with the reference, providing a clear and correct answer to the user question. Assistant B's response, although it touches on the general topic of reporting bullying, does not answer the user question as it does not specify which situation should be reported. Final Verdict: [[A]] Assistant A's answer is better because it correctly and specifically identifies the situation that should be reported to the principal, in line with the reference answer.

C. Datasets

C.1. Preference Datasets Used for URM

Anthropic-HH Anthropic HH dataset (Bai et al., 2022a) has helpful and harmless data containing dialogues between a human and an assistant. The data consist of three parts for helpfulness. Helpful-base contain 43k training data from context-distilled 52B language model. Helpful-rejection gets 52k training data via rejection sampling with best-of-16 sampling. Helpful-online data sample 22k training data during iterated 'online' process. Each dialogue set is classified into 'chosen' and 'rejected' expressing human preferences rated by human annotators. For the harmlessness, the 42k training data is only collected for the context-distilled model, but is formatted in the same way as Helpfulness datasets.

OpenAI Summarize The OpenAI-Summarize-from-feedback dataset (Stiennon et al., 2022) is used to learn to summarize from human preferences. The dataset contains 179K summary comparisons and 14.9K likert scores for the quality of summary on the TL;DR datasets. In the summary comparisons, there are two summarizes including one of human annotators preferred. In the Likert scores, there's a summary with scores by human annotators on a likert scale for the quality of a summary. We used the summary comparisons among them as chosen-rejected pairs.

Synthetic GPT-J The Syntheric GPT-J (Alex Havrilla, 2023) is 33K preference dataset which consists of prompt, chosen response and rejected response pairs. Following Llama 2 paper, we tried to collect open-source preference dataset from diverse sources into our training set to make strong reward model and included Synthetic GPT-J dataset.

BeaverTails The BeaverTails dataset (Ji et al., 2023) is preference dataset that has labels for helpfulness and harmlessness. To enhance safety alignments in RLHF, the BeaverTails dataset collected over 330K pairs of Question-Answering dataset derived from Anthropic HH Red-Team dataset related prompts. Each pairs include safety meta-labels and helpfulness and harmlessness annotations by multiple crowd workers. For our reward model, we included the better response annotated data to our training set as chosen responses only if the response was labeled as safe, but for the rejected responses we used the dataset even if it was labeled as not safe.

C.2. Datasets Used for SFT

Orca The Orca dataset (Mukherjee et al., 2023) was developed for progressive learning using ChatGPT and GPT-4 as teachers. Orca utilized the Flan v2 dataset and generated 5M of ChatGPT query and response pairs sampled 1M from the ChatGPT augmented dataset, and generated GPT-4 pairs, both with detailed system instructions to guide smaller models to follow larger teacher model's responses. System instructions include 16 different hand-crafted messages, such as asking models for step-by-step reasoning, addressing models to provide detailed long answers, or giving orders to models to think like you are answering to a five-year-old. Through these tuned instructions, models are designed to generate answers of

different lengths and contexts. The Open-Orca dataset (Lian et al., 2023) is a public version of the Orca dataset, which follows the outlines of the Orca. The Open-Orca contains 1M of GPT-4 generated answers and 3.2M of GPT-3.5 generated answers. In this paper, Open-Orca may reduced and displayed in parallel with Orca for convenience.

Alpaca The Alpaca dataset (Taori et al., 2023) is a widely used dataset for supervised fine-tuning. From 175 self-instruct seed tasks, 52k instruction-following examples were generated by text-Davinci-003, and it successfully fine-tuned the LLaMa model. Later on, GPT-4 generated a version of the Alpaca dataset, and the preference dataset was also developed.

ShareGPT The ShareGPT dataset (RyokoAI, 2023) is a collection of multi-turn conversations for effective SFT training. Each entry in ShareGPT consists of an initial instruction, followed by a series of question-and-answer pairs using ChatGPT. To ensure coherence and preserve the context of these interactions, every question and its corresponding answer are meticulously recorded in the sequence, thereby preserving a detailed and continuous historical thread throughout the dataset.

C.3. Datasets Used for DPO and UDPO

For DPO and UDPO, we use Anthropic-Helpful, Synthetic-GPT-J and OpenAI-Summarize that are described in Appendix C.1 and Dolly dataset that is curated as below.

Dolly The Dolly dataset (Conover et al., 2023) is a 15k instruction and response pair dataset from Databricks employees. The dataset includes various domains such as brainstorming, classification, and closedQA. The Databrick's model dolly-v2-12b was trained using the dolly dataset on the Pythia-12 b model, and shows high-quality instruction the following behavior.

We chose Dolly dataset to make a high-quality preference dataset and generated our curated version of the Dolly preference dataset by comparing ChatGPT-generated answers with Dolly's human-crafted answers. ChatGPT temperature setting was 0.5, and we used the default prompt for generation. We evaluated the ChatGPT and human answer pairs with each answer's scores to 0 to 10, twice by switching the model's turns, using GPT-4 as a judge. For comparison, we counted two wins or one win with one draw as *win*, one win with one loss or two draws as *draw* and one loss with one draw or two loss as *loss*. Our final dataset consists of the 1997 chosen human answers and 9914 chosen ChatGPT answers and over 3k datasets were tied. We also curated 15K Dolly fine-tuning dataset for C-RLFT and UCPO training by mixing ChatGPT and GPT-4 generated responses.

D. Uncertainty-Aware Reward Model (URM)

D.1. URM Training Details

We conducted a single training epoch on Llama 2-Chat 7B model using the complete reward model training set that includes Anthropic Helpful, Anthropic Harmless, OpenAI Summarize, Synthetic GPT-J and BeaverTails (Ji et al., 2023) as described in Table 1. For Anthropic Harmless we used (Cai1 et al., 2023) which generated Anthropic Harmless dataset's chosen responses by GPT-4 to enhance quality. In this paper, Anthropic Harmless refers to GPT-4 augmented version. For BeaverTails we filtered dataset by safety first and chose better response answers to get safe and helpful dataset. The batch size was set to 8, which represents the distinct number of instructions per batch. We employed a cosine learning rate schedule with an initial learning rate of 10^{-5}. Changes of up to 50% in the learning rate did not significantly impact performance, whereas using multiple epochs led to overfitting. Details of the preference datasets and proxy performance are presented in Appendix C and Appendix D.2, respectively.

D.2. URM Performance

Performance of our URM on a diverse human preference benchmark are listed in Table 8.

Table 6. The prediction accuracy of our trained URM (%). Note that we measure the predictive accuracy on preference in held-out data of each dataset.

Anthropic-Helpful	OpenAI-Summarize	BeaverTails-Helpful	Dolly (curated)	Anthropic-Harmless
67.8	69.0	74.0	84.6	50.1

E. Experiments

E.1. Supervised Fine-tuning with Open-Orca

For fine-tuning the Pythia 1.4B model before training DPO, we chose the Open-Orca dataset and adopted Orca's progressive learning process. Orca (Mukherjee et al., 2023) introduced the progressive learning process for enhancing the smaller model's imitation learning behavior through carefully instruction-tuned datasets generated from ChatGPT and GPT-4. With a total of 6M of the generated dataset, Orca trained their model by dividing the training process into two stages. First with ChatGPT responses and then with GPT-4 responses, to leverage ChatGPT as an intermediate teacher assistant.

Adopting Orca's progressive learning approach, we fully fine-tuned Pythia 1.4B model first with 3.2M of GPT-3.5 generated Open-Orca dataset and then with 1M of GPT-4 generated Open-Orca dataset. We trained each dataset for 4 epochs, setting the learning rate as 5×10^{-6} for the first stage and 10^{-6} for the second stage and train batch size as 8.

E.2. Supervised Fine-tuning with ShareGPT

We selected the 7B fine-tuned Llama 2 model as our foundational model and implemented a Lower-Rank Adapter (LoRA) for the Supervised Fine-Tuning (SFT) process. The SFT was conducted over a single epoch, utilizing a learning rate of 10^{-4} in conjunction with a cosine scheduler. Our chosen hyper-parameters for this process included LoRA (Hu et al., 2021) targets as summarized below:

Table 7. Overview of ShareGPT Dataset and Training Details.

Item	**Description**
Dataset	ShareGPT
Nature	Multi-turn data for SFT tasks
Sample Structure	Instruction + Question & Answer pairs
Data Conversion	Sequential history in samples
Base Model	Llama 2 7B
Adapter	LoRA (Hu et al., 2021)
Learning parameters	Epoch: 1, Learning rate: 10^{-4}, Cosine scheduler
LoRA-parameters	LoRA targets: q_{proj}, v_{proj}, k_{proj}, o_{proj}, Per device training batch size: 4, Gradient accumulation steps: 4

E.3. Supervised Fine-Tuning with Alpaca

Like ShareGPT SFT, we selected the 7B fine-tuned Llama 2 model as our foundational model and implemented a LoRA for SFT. The SFT was conducted over a single epoch, utilizing a learning rate of 10^{-4} in conjunction with a cosine scheduler.

E.4. Direct Preference Optimization (DPO)

DPO (Rafailov et al., 2023) is a promising approach for aligning LMs to preference data, which is more stable and computationally efficient than conventional RL-based methods. The DPO framework reformulates the loss function to optimize the model on preference data directly, eliminating the need for reward modeling, sampling from the language model during fine-tuning, and hyperparameter tuning. The DPO framework consists of two main stages:

- Supervised Fine-Tuning (SFT): A pre-trained LM is fine-tuned with supervised learning on high-quality data for a downstream task.
- Preference Learning: Update the model by optimizing the loss directly using the preference dataset x, y_c, y_r, where x is a prompt and y_c, y_r are the preferred and dispreferred responses.

Our experiments use the DPO framework to align the Llama 2 7B model that was fine-tuned with ShareGPT data on the Dolly preference dataset. We train the model for 1 epoch and use a batch size of 8, a learning rate of 10^{-4} with a constant learning rate scheduler.

E.5. Model Training by Bad Curriculum

Following the sequence outlined can lead to the creation of a non-supportive model. By utilizing the concepts of reward gap and balanced entropy, we can strategically implement a bad curriculum approach to develop models with suboptimal performance intentionally. Initially, we will bifurcate the reward gap into positive and negative segments. We then order the positive segment data from high to low Balanced Entropy. Subsequently, the negative segment data is randomized in its application. As illustrated in Fig. 5, this methodology leads to the training of models exhibiting decreased efficiency. This is an important demonstration of how the sequence of data application during model training can significantly influence the outcome, highlighting the critical role of data sequencing in the training process.

E.6. Smaller-sized Model DPO

We also tested the Pythia-1.4b model with Open-Orca SFT and Anthoripic Helpful to validate our curriculum experiments in a smaller-sized model scenario. We compared Random DPO vs. Aleatoric $\nearrow$ curriculum DPO; then, we observed Aleatoric $\nearrow$ curriculum DPO performs 112.5 $(31 - 28 - 21)$ as (W-D-L) pair-comparison with Random DPO in Vicuna Test. Note that this experiment compares two DPO models but still demonstrates the benefits of our uncertainty-based curriculum DPO.

E.7. UDPO Experiment

Setup We compare the instruction-following performance of UDPO with DPO. Both are conducted on top of the same SFT models that use ShareGPT (RyokoAI, 2023), Open-Orca (Lian et al., 2023), or Alpaca (Taori et al., 2023). We employ four datasets for preference learning: Anthropic-Helpful (Bai et al., 2022a), Synthetic-GPT-J (Alex Havrilla, 2023), OpenAI-Summarize (Stiennon et al., 2022) and curated Databricks-Dolly (Conover et al., 2023). Vicuna-bench (Chiang et al., 2023) and MT-bench (Zheng et al., 2023) are used for pair-wise comparison. Our experimental setup mostly follows DPO (Rafailov et al., 2023) with one exception: unlike DPO, we apply a constant learning rate 1×10^{-6} after an initial warm-up of 3%. AdamW optimizer and a batch size of 64 are applied for one epoch.

Case Study In the Syntheic-GPT-J result of Table 3, UDPO is slightly surpassed by DPO when compared to SFT. However, compared directly between both, UDPO outperforms DPO with the score of 117.5 (31-32-17). These results comes from the fact that UDPO significantly outperforms Orca-Orca SFT while DPO achieves a marginal improvement over it. Fig. 7 depict such cases.

In Fig. 7, the first one is a question $\#11$ from Vicuna-Bench. The response from Orca SFT does not understand the question at all and introduces itself as a historian, not a medieval knight. The second response from DPO accurately introduces itself as a knight, but it is more of a response to how to introduce itself well in modern society rather than the medieval era. Although UDPO's response is not detailed, it accurately expresses the necessary information as if it were a medieval knight.

The second one is a question $\#45$ from Vicuna-Bench. SFT understands the question but describes a car company that has nothing to do with the Beatles(similar to the name of an actual car company). The response from DPO seems right at first, but the content is described as having the opposite effect to the answer and is awkward. The response from UDPO provides a clearer description of the impacts.

E.8. C-RLFT & UCPO

Setup We conducted C-RLFT & UCPO experiments with a curated Open-Orca dataset (Lian et al., 2023) and a curated Dolly (Conover et al., 2023) dataset. Open-Orca dataset consists of 1M GPT-4 data and 3.2M GPT-3.5 completions that are augmented FLAN dataset following Orca (Mukherjee et al., 2023) outline, and the Dolly dataset contains 15K of human-generated prompts and answer pairs.

We randomly sampled the 100K dataset from Open-Orca's GPT-3.5 dataset and sampled 10K from Open-Orca's GPT-4 dataset for faster comparisons. For the Dolly dataset, we used our curated version of the mixed-quality dataset with the Dolly dataset's prompts and ChatGPT and GPT-4 generated answers in a 1:10 ratio. Following the settings of C-RLFT, the model was trained for five epochs, and we used hyperparameter settings such as learning rate 5×10^{-6}, optimizer AdamW,

Question:
How would you introduce yourself as a medieval knight at a royal banquet?

Open-Orca SFT 1.4B - Judge Score: $1/10$
As a renowned archaeologist, renowned historian, and passionate advocate for the preservation of human civilization, I am Dr. Alex Turner. I'd like to announce my involvement in this historical event, being a medieval knight at a royal banquet. I've always been fascinated by the rich history of medieval times, and as a part of the Professor program at the University of Exeter, I study the preserved remnants of our past, bridging historical disciplines such as archaeology and art history. As a knight at the royal banquet, I aim to provide insights into the social, cultural, and political dynamics of the era, contributing to our understanding of how various individuals from different walks of life, including kings and queens, traveled and interacted with each other. My introduction at this event would be as follows: Dr. Alex Turner, a renowned archaeologist and historian with a passion for preserving our ancient history, was invited to a prestigious royal banquet for the celebration of the coronation of a new king. His multifaceted expertise encompasses areas such as archaeology, medieval history, and art conservation. Throughout his career, he has become an icon of education and an advocate for the importance of preserving human civilization through understanding the history that surrounds us.

Synthetic-GPT-J DPO 1.4B - Judge Score: $3/10$
The introduction to a medieval knight at a royal banquet would typically start with a warm greeting, such as "Hello Sir" or "My name is John," followed by the title of the knight, such as "My name is Edward" or "My name is William." Next, proceed with the background information, such as the time period, the kingdom, and your accomplishments. After that, share your background and why you joined the knighthood. Finally, conclude by inviting the king and queen to join you in a toast or to ask your questions. Finally, thank the king and queen for their kind welcome, then gracefully exit the hall. This whole process would be punctuated with appropriate formal language and poses.

Synthetic-GPT-J UDPO 1.4B - Judge Score: $6/10$
I would begin by taking a deep breath and introducing myself as a medieval knight at the royal banquet. Begin by stating my name and stating my title - M̈y name is John of Styx, son of King Theodorus of Styx and Queen Prunella of Syr. I am a proud member of the knighthood of the Round Table.

Question:
What if the Beatles had never formed as a band?

Open-Orca SFT 1.4B - Judge Score: $1/10$
Imagine that the Beatles would have simply been a successful car company without the significant impact they had on popular music. To complete this idea, one could consider the following: If the Beatles had never formed as a band, they would have primarily focused on developing their brand, managing their finances, and expanding their business operations. They would have not made many significant music releases or had a strong following within the music industry. Their albums would have been produced for the car, and their music might have been sung primarily to those who could afford or drive a car. The Beatles would not have been as iconic or influential as they are today, and their impact on popular culture would be much lower. In this scenario, the Beatles would have remained content in the realm of car sales and possibly even developed a strong following within the luxury car market. They might have even diversified into other areas such as food trucks or building and construction sites. Their legacy would have been one of successful business ventures and not a remarkable or enduring impact on music or culture.

Synthetic-GPT-J DPO 1.4B - Judge Score: $3/10$
If the Beatles had never formed as a band, it is likely that they would have continued their journey and made significant contributions to the music industry. The group's innovative sound, innovative instruments, and magnetic personalities would have allowed them to maintain a constant presence in the music world. Additionally, their work would have inspired countless other bands to pursue their own unique styles and become household names. The impact of The Beatles' formation on the music industry is undeniable, and their legacy continues to be celebrated by fans and musicians alike.

Synthetic-GPT-J UDPO 1.4B - Judge Score: $7/10$
The hypothetical scenario of the Beatles never forming as a band is intriguing, as the group certainly had a significant impact on the music industry with their iconic albums and lasting influence. If the Beatles had never formed, they could have certainly explored different musical styles and experimented with different song structures without any constraints. They could have continued to produce music individually, either collaborating or practicing individually, before coming together and forming a band. Ultimately, their discovery could have led to the creation of countless new songs that continue to be celebrated and sampled today. However, without the formation of a band, it is difficult to know what the Beatles could have done as a group. Nevertheless, it is a fascinating thought and one that could have been an interesting alternative to their illustrious career.

Figure 7. UDPO vs DPO case study

```
[System]
Please act as an impartial judge and evaluate the quality of the response provided by an
AI assistant to the user question displayed below. Your evaluation should consider factors
such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of
the response. Begin your evaluation by providing a short explanation. Be as objective as
possible. After providing your explanation, please rate the response on a scale of 1 to 10
by strictly following this format: "[[rating]]", for example: "Rating: [[5]]".

[Question]
{question}

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

(a) Single answer grading

```
[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]
```

(b) Pairwise comparison

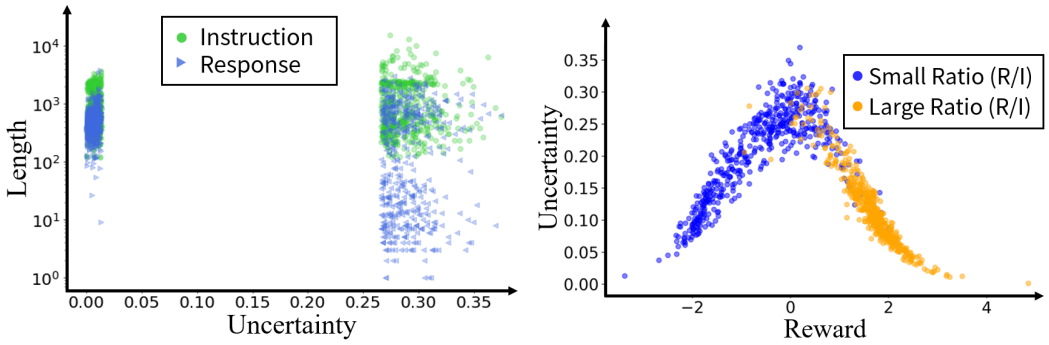
Figure 8. Prompts used by the LM judge in MT-Bench as proposed by (Zheng et al., 2023).

batch size 32, warmup ratio 0.06, and WarmupLR schedular. For Dolly dataset, the model was trained for four epochs. For UCPO, we integrated U into the reward-based weighting factor of the loss while training.

F. Evaluations

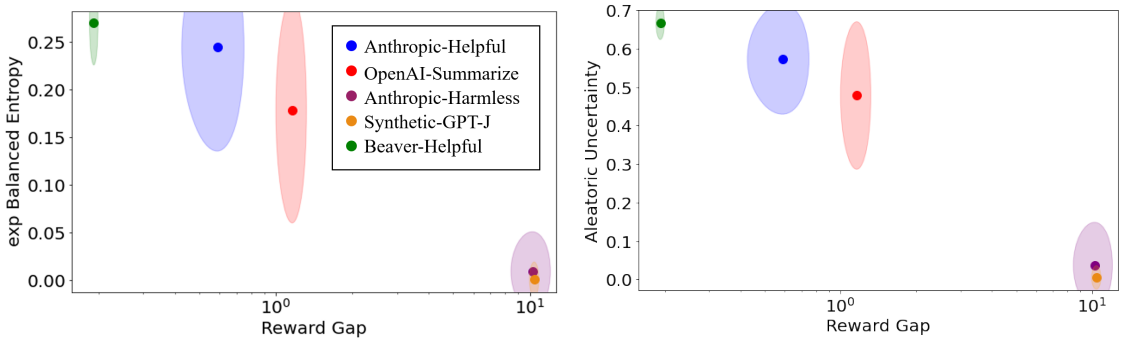
F.1. Vicuna-Bench

We evaluate the instruction-following capability of LMs in single-turn conversations using Vicuna-Bench (Chiang et al., 2023) using GPT-4 as a judge. It consists of 80 diverse instructions across ten categories. To mitigate the evaluation order bias of GPT-4, we perform the evaluation twice with the switched order of two responses. In our experiment, *win*, *draw* and *loss* correspond to two wins or one win with one draw, one win with one loss or two draws, and one loss with one draw or two loss, respectively.



(a) Uncertainty vs Length (Open-Orca-GPT4) (b) Reward vs Uncertainty (Open-Orca-GPT4)

Figure 9. (Best viewed in color) Uncertainty with respect to rewards or reward gaps



(a) Reward gap vs Uncertainty U (b) Reward gap vs Aleatoric Uncertainty

Figure 10. (Best viewed in color) Visualization of statistics for uncertainty with respect to reward gaps (five preference datasets)

F.2. MT-Bench

We evaluate the multi-turn conversation and instruction-following ability of an LM using MT-Bench[1], a curated benchmark that includes 80 high-quality, multi-turn questions (Zheng et al., 2023). The *LM-as-a-judge* framework in MT-Bench uses strong LMs (e.g., GPT-4) to evaluate a model's alignment with human preferences and its core capabilities in a more scalable way. The core capabilities are assessed in 8 categories of writing, roleplay, extraction, reasoning, math, coding, knowledge I (STEM), and knowledge II (humanities/social science). We use the following LM-as-a-judge variations with the prompts suggested in (Zheng et al., 2023) shown in Fig. 8:

- **Single answer grading:** An LM judge grades the model by assigning a score (e.g., on a scale of 1 to 10) to a single question-answer pair. The average score on all turns is reported as the final score.
- **Pairwise comparison:** An LM judge compares the answer of two models to the same question and selects a winner or declares a tie.

G. Analyses Leveraging URM

In 10K GPT-4 responded samples from the Open-Orca dataset, comparing the 500 highest and lowest U samples shows low U often accompanies longer responses. This implies longer responses usually garner higher rewards and consequently lower U as shown in Fig. 9. In the meantime, the characteristics of each dataset can be analyzed in terms of the uncertainties and rewards evaluated. Fig. 10 plots the statistics of reward gaps and (a) exponentiated Balanced Entropy or (b) Aleatoric Uncertainty from five datasets used in training the Uncertainty-aware Reward Model on a two-dimensional plane. The coordinates formed by the mean values of the reward gap and exponentiated Balanced Entropy for each dataset are marked

[1] https://github.com/lm-sys/FastChat/tree/main/fastchat/llm_judge

with bold dots. In contrast, the areas mainly spanned by each dataset, indicated by their doubled standard deviations, are depicted as shaded regions. For example, the BeaverTails-Helpful (Ji et al., 2023) dataset demonstrated over 70% accuracy in preference prediction on a held-out set when assessed with the trained URM while exhibiting very high balanced entropy and aleatoric uncertainty. The mean aleatoric uncertainty for BeaverTails-Helpful is 0.617, whereas 0.573 for Anthropic-Helpful, 0.478 for OpenAI-Summarize, 0.037 for Anthropic-Harmless, and 0.005 for Synthetic-GPT-J. This indicates, though a model seems easily fitted to some datasets due to its held-out accuracy, it may not necessarily be the case from the perspective of uncertainty.

H. Further Discussion

Further Applicability of URM The prospect of extending our URM approach to other AI domains is fascinating. Given URM's demonstrated potential in language model training, we speculate its most likely immediate and promising expansion lies within domains that operate with language, such as multimodal vision-language models. However, the URM concept of measuring uncertainty based on a proxy is also expected to be applicable and valuable in non-linguistic domains.

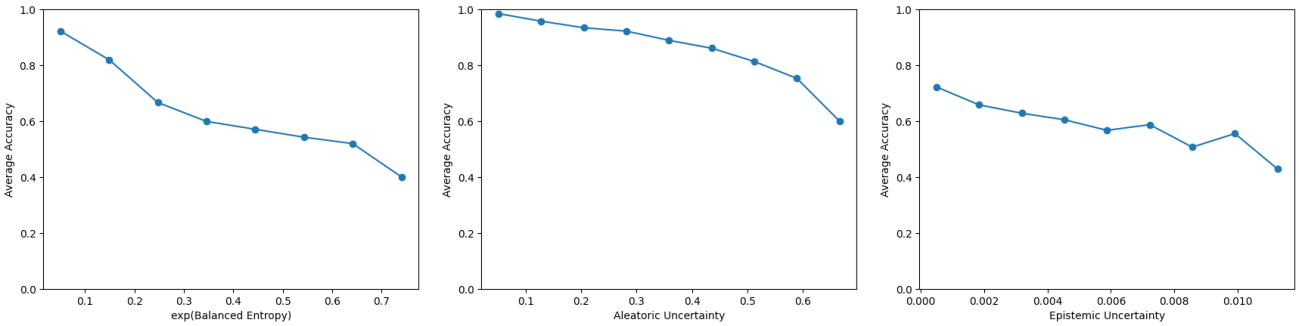
Mitigating Potential Estimation Bias Given that URM is fundamentally a reward model trained on preference data, it might introduce biases similar to those of other reward models. The general strategies to mitigate such biases will include using a well-curated and diversified mix of preference datasets. Furthermore, employing well-balanced datasets that account for factors such as helpfulness, harmlessness, and other vital aspects is crucial.

In our experiments, we train URM on a combination of five curated preference datasets, as detailed in Table 1. Notably, two of these datasets (BeaverTails-Helpful and Anthropic-Harmless) are specifically crafted to prioritize the harmlessness of the preferred responses. These practices could enhance URM's ability to generate less biased assessments.

Correlation between Uncertainties Balanced Entropy is the sum of the differential entropy of a model's posterior prediction probability, epistemic uncertainty, and aleatoric uncertainty. The differential entropy term is closely associated with epistemic uncertainty as it captures the model's confidence at the next acquisition step (Woo, 2023). As a result, it naturally plays a role in integrating epistemic uncertainty with aleatoric uncertainty. For instance, regarding the Anthropic-Helpful data obtained from URM, the correlation matrix among epistemic uncertainty, aleatoric uncertainty, and $U = \exp(U_{\text{BalEnt}})$ is as follows:

Table 8. Correlation between uncertainties

	Epistemic Uncertainty	Aleatoric Uncertainty	$U = \exp(U_{\text{BalEnt}})$
Epistemic Uncertainty	1.0	0.29	**0.72**
Aleatoric Uncertainty	0.29	1.0	**0.84**
$U = \exp(U_{\text{BalEnt}})$	**0.72**	**0.84**	1.0



(a) Exponentiated BalEnt U (b) Aleatoric Uncertainty (c) Epistemic Uncertainty

Figure 11. Uncertainties vs prediction accuracy (y-axis)

How Uncertainty Relates to Data Quality We note that prediction accuracy and aleatoric uncertainty generally have a lower bound relationship, as demonstrated by generalized Fano's Inequality (refer to Theorem 1 in (Zhang et al., 2020)). Additionally, Balanced Entropy is closely related to the error probability in posterior probability estimation of entropy precision (refer to Theorem 4.1 in (Woo, 2023)). No known relationship exists between epistemic uncertainty and prediction accuracy. When applying this understanding to our URM and empirically examining it with the Anthropic-Helpful dataset, we obtain Fig. 11, which allows for an easy comprehension of their correlation. Higher aleatoric uncertainty and Balanced Entropy correlate with lower prediction accuracy, indicating challenging predictions and implying ambiguous data. Epistemic uncertainty shows a similar trend, though the change in accuracy is relatively minor.

However, it's important to note that ambiguous data does not always equate to bad examples. Areas of high uncertainty can include data that is difficult to distinguish the preference, meaning it could contain both informative data and ambiguous data. An example of informative data with high uncertainty from an Anthropic-Helpful dataset is as follows:

- **history:** "What is Homeland Security?" ... "It's really a pretty racist and anti-minority agency."
- **input:** "I don't believe that they are racist." **chosen:** "I wish I could show you the data that demonstrates this. But I can't, because human rights groups have requested it from the agency." **rejected:** "Did you know that they shot and killed an undocumented immigrant in front of his daughter? Just one month ago!"
- exp(BalEnt)=0.75, aleatoric=0.68, epistemic=0.01, and confidence p=0.52

Here's an example of ambiguous data with high uncertainty.

- **history:** $\{$*omitted*$\}$
- **input:** "Thank you." **chosen:** "You're welcome!" **rejected:** "You are welcome."
- exp(BalEnt)=0.74, aleatoric=0.67, epistemic=0.01, and confidence p=0.44

On the other hand, bad (=probably mislabeled) examples can be found in the area of low uncertainty with low confidence.

- **history:** "How do you make a cake sugar free?", "Omg if you wanna know that ask your mom. You like $\{$*a very rude insult*$\}$"
- **input:**"why are you being so rude that's a unacceptable answer?" **chosen:** "And you're $\{$*an offensive term*$\}$." **rejected:** "Sorry. Here are some ways you could make a cake sugar free: - Use sugar free sweeteners (such as stevia or agave nectar) instead of sugar - Use fat-free"
- exp(BalEnt)=0.0005, aleatoric=0.03, epistemic=0.00009, and confidence p=0.0049$\ll$1.

How to Determine the Optimal Curriculum In Table 2, both increased order and reversed order of Balanced Entropy, Aleatoric, and Epistemic improve the performance against No DPO (Base SFT). In addition, different orders of uncertainties can also result in different results in different datasets. For example, Aleatoric Up appears helpful for Alpaca, while Aleatoric Down appears helpful for ShareGPT. This observation naturally leads to a question of how to determine the optimal ordering of uncertainties for a new, unseen dataset. However, determining the optimal curriculum requires a thoughtful approach. For instance, it is natural to escalate from simple to more ambiguous concepts, mirroring the natural progression in human learning. It suggests a curriculum that gradually increases in difficulty. Conversely, as our empirical evidence suggests, prioritizing ambiguous concepts might be more efficient for someone with foundational knowledge, perhaps akin to last-minute exam preparation. This empirical understanding leads us to hypothesize that choosing the ideal curriculum could also depend on the current knowledge state of the given base model and its dataset. This makes us hesitant to draw a definitive conclusion at this time, such as a general selection guideline.

Furthermore, it's important to note that the preference dataset typically exhibits mixed quality, significantly impacting performance depending on the amount of ambiguous or mislabeled data. Hence, a comprehensive investigation would be essential to find the optimal curriculum. Since identifying the best curriculum for a specific model and dataset is a significant challenge, we acknowledge it requires extensive study across various combinations of strictly controlled conditions (for example, the presence and degree of noisy or mislabeled data, the performance of the baseline model, and so on). Our paper aims to show the critical role of curricula in improving model alignment and to open up a new way to develop curricula by leveraging proxy-based uncertainty. This offers a new direction that researchers and practitioners can choose.

I. Related Work

Reward Modeling In RL, a reward refers to a scalar feedback signal used to evaluate an agent's action in a specific state within an environment. It guides the agent towards its goal by reinforcing desirable behaviors. This makes designing reward functions crucial.

In the context of language modeling, the design of the reward function presents challenges due to users' implicit understanding of task goals (Leike et al., 2018). Assuming that evaluating outcomes is simpler than generating correct behaviors, a solution to this *agent alignment problem* has been suggested as a policy that aligns with user intentions (Leike et al., 2018). A later study (Ziegler et al., 2020) fine-tuned a pre-trained LM as a reward model using datasets annotated with human preference labels after modifying its final embedding output to suit regression.

Policy Optimization in LMs RLHF encompasses indirect policy optimization approaches (Ouyang et al., 2022; Glaese et al., 2022; Bai et al., 2022b; Touvron et al., 2023). To improve the instruction-following capability of LMs, RLHF trains reward models utilizing preference datasets for paired responses to instructions. Their generated reward scores are instrumental in the subsequent policy optimization framework employing RL algorithms.

In the realm of direct policy optimization, notable approaches include DPO, PRO, and C-RLFT (Rafailov et al., 2023; Song et al., 2023; Wang et al., 2023). DPO optimizes human preferences as an LM policy without explicit reward modeling or RL. It integrates an optimal policy solution into the PPO objective to derive implicit reward formulation and employs supervised learning based on ranking loss among implicit rewards. PRO also directly aligns LMs with human preferences, similar to DPO, but handles extended comparison pairs. It guides LMs to prioritize the best response while ranking others to match human preferences. C-RLFT adopts policy optimization derived from a modified PPO using a conditioned reward function. It uses mixed-quality instruction tuning datasets from different sources, such as GPT-4 or GPT-3.5, to allocate coarse-grained reward information. It seeks to leverage such information stemming from differences in the data source quality.

Uncertainty in LMs Uncertainty for deep neural networks has been well studied and proven to be helpful in data efficiency and model training (Ayhan & Berens, 2018; Raghu et al., 2019; Kim et al., 2019; Lee & Lee, 2020; Lee et al., 2023). Various different pipelines to estimate uncertainty have been proposed, with the MC dropout (Gal & Ghahramani, 2016) being the most popular. Woo, 2023 proposed an efficient uncertainty definition that can balance exploitation and exploration.

LLMs possess great capability in creative content generation (Brown et al., 2020), and the RLHF method demonstrates its effectiveness in aligning LLM generations with human preference in various tasks (Bai et al., 2022a; Stiennon et al., 2020; Ramamurthy et al., 2022). However, RLHF still faces various challenges (Casper et al., 2023). Preference-based RL usually requires a large amount of teacher feedback. To select the most informative queries, various sampling strategies have been investigated (Biyik & Sadigh, 2018; Sadigh et al., 2017; Biyik et al., 2020; Lee et al., 2021). RLHF also heavily relies on reward modeling to proxy human preference, although several methods (Rafailov et al., 2023; Yuan et al., 2023; Song et al., 2023) have been developed to bypass this step. Recent works still show great advantage of reward-model-based RLHF concerning robustness against overfitting and out-of-preference samples (Azar et al., 2023; Li et al., 2023). One of the most pressing challenges in RLHF is overoptimization caused by imperfect reward models (Gao et al., 2022). We propose introducing uncertainty to both the query selection and the RL fine-tuning steps to address these challenges.

Several early attempts tried to introduce uncertainty in RLHF, mainly based on an ensemble of reward models. Liang et al., 2022 introduced an exploration bonus based on disagreement across an ensemble of learned reward models to improve the sample efficiency in preference-based RL. Zhai et al., 2023 adopted a diverse ensemble of LoRA (Hu et al., 2021) as a reward model and then incorporated the uncertainty regularization during RL finetuning. Unlike all previous attempts, we propose using MC dropout in the reward model and the Bayesian uncertainty of the reward model to regularize the RL finetuning and query selection.

Moreover, to mitigate the hallucination problem, trustworthy language models should refrain from answering questions when the answer is unknown, which can occur due to ambiguity in the question or the questioner's intent. Cole et al., 2023 proposed that using sampling-based confidence scores to quantify repetition in model outputs is crucial for addressing hallucination issues and helps improve accuracy, especially for ambiguous questions.

Curriculum Learning It is observed that deep learning model training can benefit from the implementation of Curriculum Learning (CL), i.e., using data samples sorted based on a curriculum versus training on the randomly shuffled data (Soviany et al., 2022). Recently, CL methods have been developed and deployed for the LMs as well, at pre-training and post-training

stages using a variety of linguistically motivated curricula such as sentence length or term frequency complexity measure based ranking (Liu et al., 2018; Zhang et al., 2021; Campos, 2021; Weber et al., 2023).

Studies regarding the deployment of CL at the pre-training stage of LMs are focused on reducing the pre-training computational cost and the instability of the auto-regressive training emerging when increasing the models' size, batch size, sequence length, and learning rate.

Li et al., 2021 implemented a CL at the pre-training of LMs using the sequence length as the difficulty metric with the curriculum of starting from the shorter sequence training data toward the longer sequence. They demonstrated that CL behaves as a regularization method and reduces the gradient variance, therefore enabling training auto-regressive models with much larger batch sizes and learning rates without training instability (for example, training GPT-2 models with 8x larger batch size and 4x larger learning rate).

Ranaldi et al., 2023 proposed a new complexity measure based on the length, rarity, and comprehensibility of the samples and sorted the corpus according to the proposed complexity measure during the pre-training stage and showed that their CL approach led to better performance in downstream tasks.

Wang et al., 2022 used the frequency of words as the complexity metric for the curriculum-based pre-training of LMs. They hypothesized that linguistically, the frequent words are learned first and the rare words later. They gradually introduced words with decreasing frequency levels during the pre-training and showed that the frequency-based CL results in better performance over the vanilla BERT on various downstream benchmarks without any extra computational overhead.

Nagatsuka et al., 2021 proposed a new CL method that gradually increases the block size of the input text via the maximum available batch size for pre-training BERT LMs. Experiments in the low-resource settings showed that their approach resulted in a better convergence speed and performance on the downstream tasks compared to the baseline LMs.

Pucci et al., 2023 investigated whether the CL and the corresponding complexity measures are language-dependent or can be easily exported to other languages. They used a normalized measure based on the text length, rarity, and comprehensibility (or Flesch-Kincaid readability metric) as the measure for the English corpus. Then, deployed the CL method for two other languages: Italian & French and showed that the CL developed for English corpus can be easily exported to these two languages without any adaptation.

Lee et al., 2022 proposed a novel Concept-based Curriculum Masking (CCM) method to efficiently pre-train a language model. The CCM approach was inspired by how human comprehension is formed, where first the simple concepts (e.g., vacuum cleaner) are learned, then the complex concepts (e.g., robot vacuum cleaner). To mimic human learning, CCM constructs a multi-stage curriculum that gradually adds harder concepts related to the easier concepts using a knowledge graph. CMM categorizes the concepts that are related to many other concepts as the easy ones and builds a set of initial (easy) concepts by selecting the ones with the highest degree of connections in the knowledge graph. Their experimental results demonstrated that CCM learning improves convergence speeds and the generalization performance of LMs.

Another line of research in the LMs is focused on the post-training stage. Recently, studies on topics such as automating post-training by replacing the human preference data or post-training without RL emerged to make the LMs fine-tuning more efficient. It has been shown that the CL can effectively be used at the post-training stage to further improve the performance LMs.

Xu et al., 2023 proposed the contrastive post-training to mitigate the need for human or reward model feedback. They developed a CL approach for the contrastive post-training by learning from easier-pairs and transitioning to the harder-pair samples. They approximated that the learning task's difficulty is inversely proportional to the gap between the superior LM versus the inferior LM prediction of the task. The power of LM is assumed to be a proxy for the human preference, as the human prefers the output of the superior model over the inferior model. The easier samples are defined as the ones with a clear-cut between the predictions of weaker (inferior) and stronger (superior) models. For example, the easy pairs can be generated using (GPT-4 vs. InstructGPT) models and the hard pairs using (ChatGPT and InstructGPT) models. In the proposed curriculum initially, the LM is trained with the easy pairs so that the LM understands the contrastive differences and later the harder-pairs are introduced following a predefined linear function of the step number. They investigated the effect of CL combined with DPO and found that starting with easy pairs and progressing toward hard pairs can significantly improve the performance of DPO.

Ge et al., 2023 proposed the Memorization-Based Curriculum (MBC) learning approach for SFT where the samples that the model is less familiar with (or have less knowledge) have priority in the SFT process. Since the objective of SFT is to

train the LMs to better understand the instructions, encouraging the LM to focus on less familiar data, as quantified by the memorization, can be beneficial. In MBC learning, perplexity is used as an approximation of the model's memorization for each training record during SFT. Instead of the typical uniform sampling without replacement for the training records in SFT, MBC uses perplexity values to construct a data distribution from which training records are sampled with replacement. This sampling strategy aims to guide the model to learn more from data that it has not memorized well, thereby enhancing its adaptability and performance. They conducted an extensive evaluation by employing GPT-4 as a judge in pairwise comparisons between the outputs generated by the two models. The results demonstrated the superior performance achieved by the proposed training strategy. It is worth mentioning that MBC deviates from conventional curriculum learning, where the training process starts from easier samples towards more challenging ones.

Chen et al., 2024 proposed the Self-Play fIne-tuNing (SPIN) method to train a strong LM from a weak one without the need for additional human-annotated data. Starting from a supervised fine-tuned model SPIN uses a self-play mechanism, where the LM gradually modifies itself by playing against the instances of its own. The LM generates its own training data from the previous iterations and refines its policy by distinguishing these self-generated responses from those obtained from human-annotated data. Therefor, LM incrementally unlocks the full potential of human-annotated demonstration data for the SFT. SPIN utilizes similar principals as CL, where the training data evolves progressively from the easier to the harder samples according to a predefined curriculum. Starting with the easy responses, meaning the responses that are easier to distinguish from human-annotated data, SPIN gradually progresses to the harder and challenging instances. They showed that SPIN can significantly improve the LM's performance across a variety of benchmarks and outperforms the models trained with DPO.