# Weak-to-Strong Extrapolation Expedites Alignment

**Chujie Zheng** [1 2] **Ziqi Wang** [3] **Heng Ji** [3] **Minlie Huang** [1] **Nanyun Peng** [2]

## Abstract

The open-source community is experiencing a surge in the release of large language models (LLMs) that are trained to follow instructions and align with human preference. However, further training to improve them still requires expensive computational resources and data annotations. *Is it possible to bypass additional training and cost-effectively acquire better-aligned models?* Inspired by the literature on *model interpolation*, we propose a simple method called **ExPO** to boost LLMs' alignment with human preference. Utilizing a model that has undergone alignment training (e.g., via DPO or RLHF) and its initial SFT checkpoint, ExPO directly obtains a better-aligned model by ***extrapolating*** from the weights of the initial and the aligned models, which implicitly optimizes the alignment objective via first-order approximation. Through experiments with twelve open-source LLMs on HuggingFace, we demonstrate that ExPO consistently improves off-the-shelf DPO/RLHF models, as evaluated on the mainstream LLM benchmarks AlpacaEval 2.0 and MT-Bench. Moreover, ExPO exhibits remarkable scalability across various model sizes (from 1.8B to 70B) and capabilities. Through controlled experiments and further empirical analyses, we shed light on the essence of ExPO amplifying the reward signal learned during alignment training. Our work demonstrates the efficacy of model extrapolation in expediting the alignment of LLMs with human preference, suggesting a promising direction for future research.

## 1. Introduction

Over the past year, the open-source community has witnessed explosive growth in large language models (LLMs). These powerful LLMs, typically with billions of parameters, are trained to follow instructions and align with human preference (Ouyang et al., 2022; OpenAI, 2022; Bai et al., 2022). Although the open weights of LLMs facilitate out-of-the-box use, further training to improve their performance usually requires expensive computational resources and additional data annotations. *Is it possible to bypass additional training and cost-effectively acquire better-aligned models*?

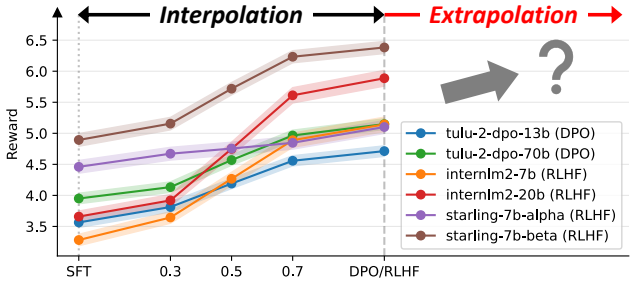



Figure 1: Calculating the reward scores (§ 3.1) on the UltraFeedback (Cui et al., 2023) development set, we observe that *model interpolation* usually gives trade-off performance between the two original models (e.g., an SFT model and a further-trained DPO/RLHF model). This observation motivates our proposal of ExPO, which cheaply acquires a better-aligned (*stronger*) model from a DPO/RLHF model and its initial SFT checkpoint (i.e., two relatively *weaker* models) via *model extrapolation*.

We draw inspiration from the literature on *model interpolation*, also known as model/weight averaging. This technique merges different models fine-tuned from the same base model by interpolating between their weights (Utans, 1996; Izmailov et al., 2018; Wortsman et al., 2022), relying on the mode connectivity of neural networks (Garipov et al., 2018; Entezari et al., 2022). Previous work observes that while model interpolation can integrate the respective strengths of different models to improve out-of-distribution generalization, it usually results in in-between performance compared to the original ones (Izmailov et al., 2018; Lin et al., 2024; Wortsman et al., 2022). We similarly observe this phenomenon when interpolating between a supervised

[*]Work done during Chujie's visit to UCLA. Project repository: `https://github.com/chujiezheng/LLM-Extrapolation`.
[1]The CoAI Group, DCST, BNRist, Tsinghua University [2]University of California, Los Angeles [3]University of Illinois Urbana-Champaign. Correspondence to: Chujie Zheng <chujiezhengchn@gmail.com>, Minlie Huang <aihuang@tsinghua.edu.cn>, Nanyun Peng <violetpeng@cs.ucla.edu>.

*ICML 2024 Workshop on Models of Human Feedback for AI Alignment*, Vienna, Austria. 2024. 

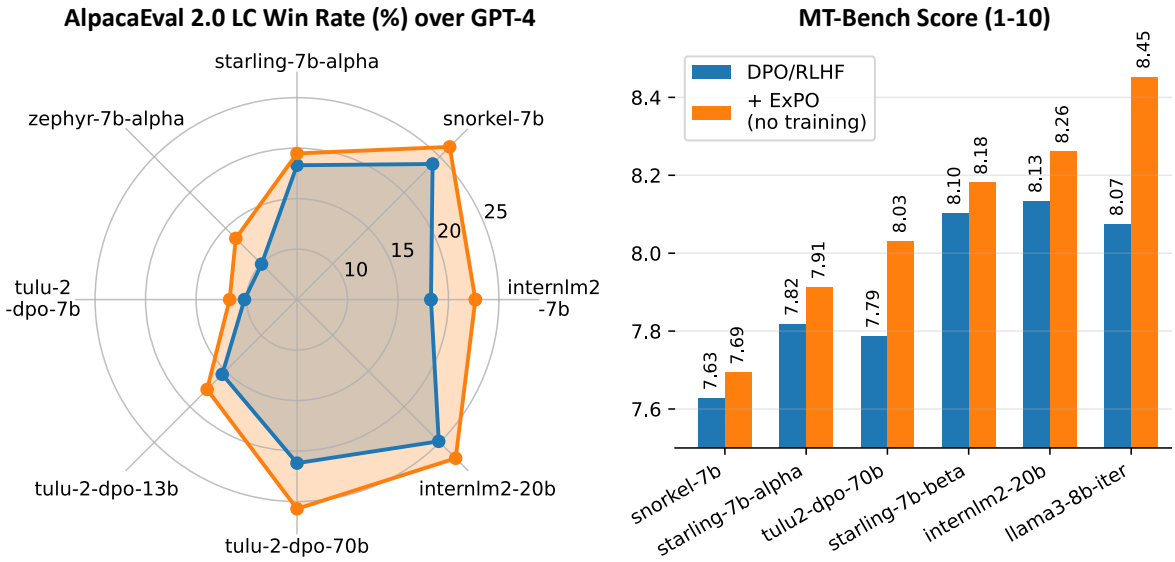



Figure 2: With *no additional training*, ExPO remarkably improves off-the-shelf DPO/RLHF models on HuggingFace across various model sizes and capabilities, as evaluated on two leading LLM benchmarks: AlpacaEval 2.0 (Li et al., 2023a) (left) and MT-Bench (Zheng et al., 2023b) (right). See Table 1 for full results.

fine-tuned (SFT) model and a model further trained by direct preference optimization (DPO) (Rafailov et al., 2023) or reinforcement learning from human feedback (RLHF) (Ziegler et al., 2019), as shown in Figure 1.

Intrigued by the observations of model interpolation, we turn to another compelling but unexplored direction: *What if we consider a DPO/RLHF model as the interpolated result from the initial SFT model and a hypothetically better-aligned model?* If this hypothetical model exists, we can straightforwardly obtain its weights by reversely **extrapolating** from the weights of the SFT and the DPO/RLHF models, as indicated by the gray arrow in Figure 1. This can potentially further improve many off-the-shelf DPO/RLHF-aligned LLMs without any additional training.

Building upon the above assumption, we propose a simple method called **ExPO** (*model **ex**trap**o**lation*) to boost LLMs' alignment with human preference (§ 2). Utilizing a model $\mathcal{M}_1$ that has undergone alignment training (e.g., via DPO or RLHF) and the SFT model $\mathcal{M}_0$ that initializes $\mathcal{M}_1$, ExPO directly extrapolates a better-aligned (*stronger*) model $\mathcal{M}_2$ from the weights of the two relatively *weaker* models $\mathcal{M}_1$ and $\mathcal{M}_0$.

Despite its simplicity, we demonstrate the impressive efficacy of ExPO through extensive empirical experiments, as summarized in Figure 2. Through experiments with twelve open-source LLMs on HuggingFace, we show that ExPO consistently improves off-the-shelf DPO/RLHF models, by up to 4.5% on AlpacaEval 2.0 (Li et al., 2023a) and 0.37 on MT-Bench (Zheng et al., 2023b) (§ 3). Moreover, ExPO also manifests remarkable scalability across various model sizes (from 1.8B to 70B) and capabilities. We further conduct controlled experiments to shed light on how ExPO amplifies the reward signal learned during $\mathcal{M}_1$'s alignment training, where we show that ExPO can boost models trained with less preference data (e.g., 10% or 20%) to compete and even outperform the fully-trained one (§ 4). Our work demonstrates model extrapolation as a promising method for expediting the alignment of LLMs with human preference, and we believe it deserves more exploration in future research.

## 2. Methodology

### 2.1. Overview

Our proposed ExPO method is inspired by the observation in Figure 1 and the mode connectivity of neural networks (Garipov et al., 2018; Entezari et al., 2022; Goddard et al., 2024). Formally, we denote that a language model $\mathcal{M}_1$ (parameterized by $\boldsymbol{\theta}_1$) has undergone training for human preference alignment (e.g., via DPO (Rafailov et al., 2023) or RLHF (Ziegler et al., 2019)). We denote its corresponding SFT checkpoint as $\mathcal{M}_0$ (parameterized by $\boldsymbol{\theta}_0$), which is used for initializing $\mathcal{M}_1$. We denote the model's parameter space as $\boldsymbol{\Theta}$ and suppose that the alignment level can be quantified by a continuous scalar function $\Omega : \boldsymbol{\Theta} \to \mathbb{R}$, where higher $\Omega(\boldsymbol{\theta})$ indicates better alignment with human preference. ExPO assumes that there exists a better-aligned model $\mathcal{M}_2$ (parameterized by $\boldsymbol{\theta}_2$) that satisfies $\Omega(\boldsymbol{\theta}_0) < \Omega(\boldsymbol{\theta}_1) < \Omega(\boldsymbol{\theta}_2)$, and an interpolation coefficient $\gamma \in [0, 1]$ such that $\boldsymbol{\theta}_1 = (1-\gamma)\boldsymbol{\theta}_0 + \gamma\boldsymbol{\theta}_2$. Here, we consider the

simplest form of uniform linear interpolation, as we find it can already work well. With the substitution of $\alpha = 1/\gamma - 1 \in [0, +\infty)$, ExPO obtains the assumed better-aligned (*stronger*) model $\mathcal{M}_2$ by *extrapolating* from the weights of the two (relatively *weaker*) models $\mathcal{M}_1$ and $\mathcal{M}_0$ (i.e., *weak-to-strong extrapolation*), formulated as follows:

$$\begin{aligned}\boldsymbol{\theta}_2 &= (1+\alpha)\boldsymbol{\theta}_1 - \alpha\boldsymbol{\theta}_0 \\ &= \boldsymbol{\theta}_1 + \alpha(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0) = \boldsymbol{\theta}_1 + \alpha\Delta\boldsymbol{\theta}, \end{aligned} \tag{1}$$

where $\alpha$ serves as the hyperparameter that controls the extrapolation length. In practice, $\alpha$ can be easily tuned as a decoding hyperparameter (similar to the sampling temperature). This requires **only one 24GB GPU for 7B LLMs** (half-precision inference with `vllm` (Kwon et al., 2023)), which, however, is far from sufficient for model training.

### 2.2. Explanation and Insights

Theoretically, ExPO takes *first-order approximation to implicitly optimize the alignment objective* $\Omega(\boldsymbol{\theta})$. Note that alignment algorithms typically include the regularization term (e.g., the KL constraint in RLHF) that restricts $\boldsymbol{\theta}_1$ within the small vicinity of $\boldsymbol{\theta}_0$ (i.e., $|\Delta\boldsymbol{\theta}|$ is small), and we can also control $\alpha$ such that $|\alpha\Delta\boldsymbol{\theta}| \ll |\boldsymbol{\theta}_1|$. We then apply first-order Taylor Expansion and have:

$$\Omega(\boldsymbol{\theta}_1 + \alpha\Delta\boldsymbol{\theta}) \approx \Omega(\boldsymbol{\theta}_1) + \alpha\nabla\Omega(\boldsymbol{\theta}_1)\cdot\Delta\boldsymbol{\theta}. \tag{2}$$

Therefore, $\Omega(\boldsymbol{\theta}_2) = \Omega(\boldsymbol{\theta}_1 + \alpha\Delta\boldsymbol{\theta}) > \Omega(\boldsymbol{\theta}_1)$ holds if the gradient of $\Omega$ at $\boldsymbol{\theta}_1$ has a positive component along $\Delta\boldsymbol{\theta}$ (as long as $\Omega$ is not locally maximum at $\boldsymbol{\theta}_1$). This can generally be satisfied, as we can reasonably assume $\Omega$ to monotonically increase from $\boldsymbol{\theta}_0$ to $\boldsymbol{\theta}_1$ during alignment training, as illustrated in Figure 3.

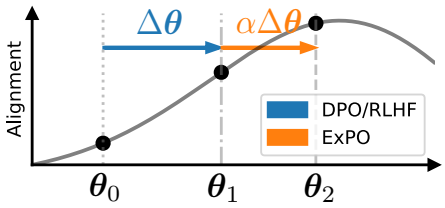



Figure 3: Illustrative 1D diagram of $\Omega(\boldsymbol{\theta})$. ExPO can be viewed as a "global gradient update" along $\Delta\boldsymbol{\theta}$. It essentially amplifies the reward signal learned during alignment training.

The above assumption also implies that the alignment training from $\mathcal{M}_0$ to $\mathcal{M}_1$ is *suboptimal*, as shown in Figure 3. We conjecture this is very likely to occur in practice due to both the regularization in alignment algorithms (e.g., the KL constraint in RLHF) and *the sparsity of reward signal*. For the latter, since it is intractable to directly optimize $\Omega(\boldsymbol{\theta})$, the dominant practice is to first employ another reward model to assign preference labels (e.g., in DPO) or reward values (e.g., in RLHF) to the language model $\mathcal{M}$'s outputs, on which we then train $\mathcal{M}$. However, transmitting the reward signal through the intermediate discrete, textual outputs can make the learned reward signal noisy or *sparse*, which consequently hinders the optimal alignment training. We will show in § 3 that the open-source DPO/RLHF models usually have significant room for further improvement.

Additionally, Figure 3 provides a more intuitive illustration of ExPO. Specifically, ExPO can be viewed as a "global gradient update" along the weight change $\Delta\boldsymbol{\theta}$. Note that starting from $\boldsymbol{\theta}_0$, $\Delta\boldsymbol{\theta}$ indicates a direction in which the alignment level $\Omega$ with human preference increases. Therefore, ExPO *essentially amplifies the learned reward signal through the extrapolation* $\alpha\Delta\boldsymbol{\theta}$. This insight underscores the importance of the "quality" of $\Delta\boldsymbol{\theta}$, i.e., $\Delta\boldsymbol{\theta}$ should indicate a direction that truly improves the alignment with human preference. Otherwise, ExPO could also amplify the learned spurious features in $\Delta\boldsymbol{\theta}$. We will provide more empirical analyses in § 4 to show that the "quality" of $\Delta\boldsymbol{\theta}$ can vary depending on the training configuration for $\mathcal{M}_1$.

### 2.3. Highlights

We underline the following appealing properties of ExPO:

- **Simplicity:** ExPO is extremely simple and quick to implement. It merely involves performing extrapolation based on the weights of two checkpoints $\mathcal{M}_0$ and $\mathcal{M}_1$, which can be implemented within just a few lines of code.
- **Efficiency:** ExPO does not need any additional model training. The only variable $\alpha$ is efficient to tune as a decoding hyperparameter, which requires much fewer computational resources than model training (e.g., **one 24GB GPU is enough for 7B LLMs**). Moreover, we believe more efficient means of hyperparameter search can be developed in future work, as evidenced by the advances in adaptive model interpolation (Ilharco et al., 2023; Lin et al., 2023).
- **Scalability:** ExPO is, in principle, applicable to various LLMs, including those of large sizes and those trained by advanced alignment algorithms like iterative DPO (Xiong et al., 2024; Dong et al., 2024; 2023). We will show in § 3 that ExPO can improve off-the-shelf models across various sizes and capabilities.

## 3. ExPO Improves Off-the-shelf Models

In this section, we demonstrate the impressive efficacy of ExPO in improving *off-the-shelf* LLMs from HuggingFace, utilizing their SFT and DPO/RLHF checkpoints. We particularly underscore the *scalability* of ExPO across different model sizes and capabilities.

### 3.1. Experimental Setup

**Models** When selecting open-source LLMs for experiments, we found that many well-known LLMs, such as LLaMA-2/3 (Touvron et al., 2023; AI@Meta, 2024), Gemma (Team et al., 2024), and Qwen (Bai et al., 2023), only release the final DPO/RLHF checkpoints but not the corresponding SFT ones. To facilitate reproducible research, we select the following **twelve open-source DPO/RLHF models** on HuggingFace that (1) have publicly accessible SFT checkpoints, (2) have disclosed the training details, and (3) are popularly downloaded:

- `zephyr-7b-alpha/beta` (Tunstall et al., 2023b), two Mistral-based (Jiang et al., 2023) models developed by HuggingFace. They are initialized from different SFT checkpoints and trained via **DPO** on UltraFeedback (Cui et al., 2023).
- `starling-7b-alpha/beta` (Zhu et al., 2023), two Mistral-based models. They are initialized from different SFT versions of the OpenChat model (Wang et al., 2024) and trained via the **RLHF** algorithm.
- `snorkel-7b` (Tran et al., 2023), a Mistral-based model. It is initialized from the official SFT Mistral model and trained via the **iterative DPO** algorithm (Tran et al., 2023) on the instructions of UltraFeedback.
- `llama3-8b-iter` (Dong et al., 2024), a LLaMA-3-based (AI@Meta, 2024) model developed by Salesforce. It is trained via **iterative DPO** on open-source datasets.
- `internlm2-1.8/7/20b` (Cai et al., 2024), a Chinese-English bilingual model suite developed by Shanghai AI Laboratory. The three-sized models undergo the same SFT training and similar online **RLHF** training processes.
- `tulu-2-dpo-7/13/70b` (Ivison et al., 2023), a LLaMA-2-based model suite developed by the Allen Institute for AI. The three-sized models undergo the same SFT and **DPO** training processes.

We decide the optimal $\alpha$ in ExPO from [0.1, 0.2, 0.3, 0.4, 0.5] based on the model performance on the instructions of the UltraFeedback[1] (Cui et al., 2023) development set. The performance is measured by the expected reward score calculated by an open-source reward model[2]. It ranks among the top on RewardBench[3] (Lambert et al., 2024), a leaderboard that assesses the performance of reward models. This reward model is also not involved in either preference annotation or RLHF training of all the models we experiment with in this work, thus reducing the risk of reward hacking.

**Benchmarks** We employ three mainstream LLM benchmarks for evaluation:

- **AlpacaEval 2.0** (Li et al., 2023a), a leading benchmark that assesses LLMs' instruction-following ability and the alignment with human preference. It calculates the probability that an LLM-based evaluator (`gpt-4-1106-preview`) prefers the model's output over the GPT-4 baseline, which provides an affordable and replicable alternative to human preference annotation. The **win rate** over the GPT-4 baseline is computed as the expected preference probability. Recently, AlpacaEval 2.0 has introduced the new **length-controlled (LC) win rate** metric (Dubois et al., 2024), which alleviates the length bias of the GPT-4 evaluator (i.e., the prior preference toward longer responses) (Park et al., 2024). According to (Dubois et al., 2024), *the LC win rate metric currently has the highest correlation (a Spearman correlation of 0.98) with the real-world human evaluation on Chatbot Arena* (Zheng et al., 2023b).
- **MT-Bench** (Zheng et al., 2023b), another leading benchmark for assessing chat LLMs' general and multi-turn ability. It contains a set of challenging multi-turn open-ended questions covering topics such as writing, role-playing, math, coding, and more. The model-generated answers are judged by `gpt-4` via a scalar score (from 1 to 10), without any pairwise comparison.
- **Open LLM Leaderboard** (Beeching et al., 2023), a popular evaluation suite hosted by HuggingFace. It consists of six benchmarks and assesses a variety of model abilities across commonsense reasoning (Zellers et al., 2019; Sakaguchi et al., 2021), math problem-solving (Cobbe et al., 2021), human falsehood mimicking (Lin et al., 2022), and general knowledge (Clark et al., 2018; Hendrycks et al., 2021). We follow the official evaluation protocol (Gao et al., 2021) and report the average scores on the six benchmarks, while the breakdowns are shown in Appendix E.

### 3.2. Results

In Table 1, we demonstrate that ExPO consistently enhances the evaluated LLMs, with increases of up to 10.1% basic win rate on AlpacaEval 2.0 (for `internlm2-20b`), 4.5% LC win rate (for `tulu-2-dpo-70b`), and 0.37 on MT-Bench (for `llama3-8b-iter`). The improvements are made across LLMs of various sizes and capabilities, from the smallest `internlm2-1.8b` and the second weakest `zephyr-7b-alpha`, to the largest `tulu-2-dpo-70b` and the strongest `llama3-8b-iter` and `starling-7b-beta`, which demonstrates the remarkable scalability of ExPO. In Figure 4, we also show that ExPO overall slightly improves the Open LLM Leaderboard scores, indicating that ExPO generally does not impact the base model's capability. Over-

[1] `https://huggingface.co/datasets/HuggingFaceH4/ultrafeedback_binarized`

[2] `https://huggingface.co/weqweasdas/RM-Mistral-7B`

[3] `https://huggingface.co/spaces/allenai/reward-bench`

Table 1: AlpacaEval 2.0 (win rate and LC win rate) and MT-Bench evaluation results of off-the-shelf DPO/RLHF models. The **gray models' scores** are copied from the official leaderboard for reference.

| | Original | | | + ExPO, *no training* | | |
|---|---|---|---|---|---|---|
| | **WR** | **LC WR** | **MT-B** | **Win Rate** | **LC Win Rate** | **MT-Bench** |
| llama2-7b | 5.0% | 5.4% | 6.27 | - | - | - |
| llama2-70b | 13.9% | 14.7% | 6.86 | - | - | - |
| mistral-7b-v0.2 | 14.7% | 17.1% | 7.60 | - | - | - |
| claude-2.1 | 15.7% | 25.3% | 8.18 | - | - | - |
| gpt-4-0314 | 22.1% | 35.3% | 8.96 | - | - | - |
| zephyr-7b-alpha | 6.7% | 10.0% | 6.85 | **10.6%** (+3.8%) | **13.6%** (+3.6%) | **6.87** (+0.02) |
| zephyr-7b-beta | 10.2% | 13.2% | 7.02 | **11.1%** (+0.9%) | **14.0%** (+0.8%) | **7.06** (+0.04) |
| starling-7b-alpha | 15.0% | 18.3% | 7.82 | **18.2%** (+3.2%) | **19.5%** (+1.2%) | **7.91** (+0.09) |
| starling-7b-beta | 26.6% | 25.8% | 8.10 | **29.6%** (+3.0%) | **26.4%** (+0.7%) | **8.18** (+0.08) |
| snorkel-7b | 24.7% | 24.0% | 7.63 | **28.8%** (+4.1%) | **26.4%** (+2.4%) | **7.69** (+0.07) |
| llama3-8b-iter | 29.2% | 36.0% | 8.08 | **32.7%** (+3.5%) | **37.8%** (+1.8%) | **8.45** (+0.37) |
| internlm2-1.8b | 3.8% | 4.0% | 5.17 | **5.2%** (+1.5%) | **4.3%** (+0.3%) | **5.26** (+0.08) |
| internlm2-7b | 20.5% | 18.3% | 7.72 | **28.1%** (+7.6%) | **22.7%** (+4.4%) | **7.80** (+0.08) |
| internlm2-20b | 36.1% | 24.9% | 8.13 | **46.2%** (+10.1%) | **27.2%** (+2.4%) | **8.26** (+0.13) |
| tulu-2-dpo-7b | 8.5% | 10.2% | 6.35 | **11.5%** (+3.0%) | **11.7%** (+1.5%) | **6.38** (+0.03) |
| tulu-2-dpo-13b | 11.2% | 15.5% | 7.00 | **15.6%** (+4.3%) | **17.6%** (+2.1%) | **7.26** (+0.26) |
| tulu-2-dpo-70b | 15.4% | 21.2% | 7.79 | **23.0%** (+7.6%) | **25.7%** (+4.5%) | **8.03** (+0.24) |

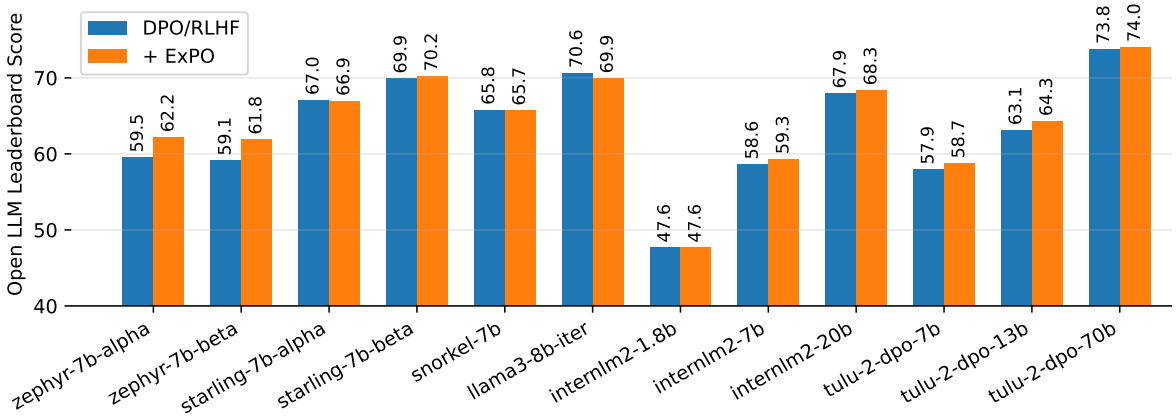



Figure 4: Open LLM Leaderboard evaluation results of off-the-shelf DPO/RLHF models. We report the average scores over the six tasks. Breakdowns are shown in Appendix E.

Table 2: AlpacaEval 2.0 evaluation results of models trained with varying sizes of preference data.

| | Original | | + ExPO, *no training* | |
|---|---|---|---|---|
| | **Win Rate** | **LC Win Rate** | **Win Rate** | **LC Win Rate** |
| SFT ($\mathcal{M}_0$) | 4.7% | 8.7% | - | - |
| DPO (init from $\mathcal{M}_0$, 100% data) | 14.7% | 17.3% | 18.0% (+3.3%) | 20.2% (+2.8%) |
| DPO (init from $\mathcal{M}_0$, 5% data) | 5.0% | 9.1% | 11.5% (+6.5%) | 14.7% (+5.6%) |
| DPO (init from $\mathcal{M}_0$, 10% data) | 5.9% | 10.4% | 17.9% (+12.0%) | 16.3% (+5.8%) |
| DPO (init from $\mathcal{M}_0$, 20% data) | 8.6% | 12.9% | **22.7%** (+14.2%) | **21.3%** (+8.4%) |
| DPO (init from $\mathcal{M}_0$, 40% data) | 12.1% | 14.6% | 17.7% (+5.6%) | 16.6% (+2.0%) |

all, our extensive evaluation suggests that most open-source LLMs have not been trained optimally for human preference alignment, while ExPO enables further improvements for them without any additional training.

## 4. Controlled Experiments and Analyses

In this section, we conduct controlled experiments to give more insights into ExPO, where we fix the same $\mathcal{M}_0$ and adopt varying training configurations for $\mathcal{M}_1$, including training data sizes and hyperparameters. We also discuss the impact of model choices of $\mathcal{M}_0$ and $\mathcal{M}_1$ on the effectiveness of ExPO. We underscore that ExPO amplifies the reward signal learned during alignment training, but it can also amplify the learned spurious features such as length bias.

### 4.1. Experimental Setup

**Models** We refer to the alignment handbook (Tunstall et al., 2023a), a widely-used code base released by HuggingFace for alignment training of LLMs. We adopt their recipe for training the `zephyr-7b-sft` and `zephyr-7b-dpo` models, which are popularly used for controlled experiments in recent LLM alignment research (Chen et al., 2024b; Ji et al., 2024b; Chen et al., 2024a). The recipe employs DPO for alignment training, where the SFT model `zephyr-7b-sft` is used as the reference model in DPO and also for initializing the policy models. We adopt the same hyperparameter configuration (see Appendix F) and train all the models on 4 A100 80GB GPUs. We use `zephyr-7b-dpo` as the fully-trained baseline (i.e., using 100% data).

**Data** We use the same preference dataset UltraFeedback (Cui et al., 2023) for alignment training. It contains diverse instructions and response pairs with GPT-4-annotated preference labels and has been popularly used by the open-source community for training aligned LLMs (Ivison et al., 2023; Tunstall et al., 2023b; Zhu et al., 2023). The preprocessed version on HuggingFace contains 61K and 1K preference data in the training and development sets, respectively. As in § 3, we search the optimal $\alpha$ in ExPO based on the performance on the instructions of the *development set*, as evaluated by the same open-source reward model.

### 4.2. Analysis of Training Data

We first study the impact of training data on the effectiveness of ExPO. We train multiple $\mathcal{M}_1$ from the same initial $\mathcal{M}_0$ (i.e., `zephyr-7b-sft`), but with varying data sizes (from 5% to 40%). In Table 2, we show their performance as well as the results of further applying ExPO to them. While training with less preference data usually results in lower-tier performance, ExPO boosts the performance to compete (10% data, 16.3%) and even surpass (20% data, 21.3%) the fully-trained model (17.3%). We also observe that the model trained with 20% data obtains a larger improvement than other data proportions. It implies that the former gives a superior extrapolation direction $\Delta\boldsymbol{\theta}$ (i.e., of a higher "quality"), as illustrated in Figure 5.

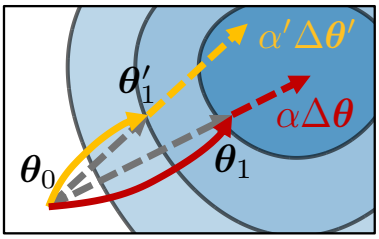



Figure 5: Illustrative 2D contour diagram of $\Omega(\boldsymbol{\theta})$. The "quality" of $\Delta\boldsymbol{\theta}$ and the effectiveness of ExPO can vary depending on the training configurations for $\mathcal{M}_1$. Here, $\Delta\boldsymbol{\theta}$ indicates a *superior* extrapolation direction to $\Delta\theta'$.

However, the "quality" of $\Delta\boldsymbol{\theta}$ is not simply correlated with the amount of data. As shown in Table 2, using 20% data slightly outperforms using 100% data when both applying ExPO (21.3% vs. 20.2%), while the gain from ExPO decreases when the used data increases to 40%. In Figure 6, we present the reward scores and output lengths on the UltraFeedback *development set* versus varying $\alpha$ values. From the left part, we observe that the global optimal reward score (6.08) achieved by ExPO is obtained with a medium size (20%) of training data, rather than the smaller (5% or 10%) or larger (40%) ones. For the former (5% and 10% data), although ExPO still notably improves the performance (from the reward score 3.13 to 4.79, and 3.59 to 5.82, respectively), the limited data still cannot provide an accurate $\Delta\boldsymbol{\theta}$, thus capping the improvement after model extrapolation. For the latter (40% data), we speculate that *the model has learned the spurious features within the training data as shortcuts, especially the length bias*[4] (Park et al., 2024) where the preferred responses are usually longer. As shown in the right part of Figure 6, for the model trained with 40% data, using a very small $\alpha$ results in a dramatic increase in the output length. However, this does not lead to sustained improvement in performance, where the optimal rewards typically correspond to moderate output lengths ranging between 500 and 600.

### 4.3. Analysis of Training Hyperparameters

As ExPO can be viewed as a "global gradient update" (§ 2.2), we also compare with simply tuning the training hyperparameters. We use the same 20% training data but increase the learning rate or training epochs, and train multiple $\mathcal{M}_1$ from the same initial $\mathcal{M}_0$. From the left part of

[4]The average lengths of the preferred and unpreferred responses in the UltraFeedback training set are 319 and 277, respectively.

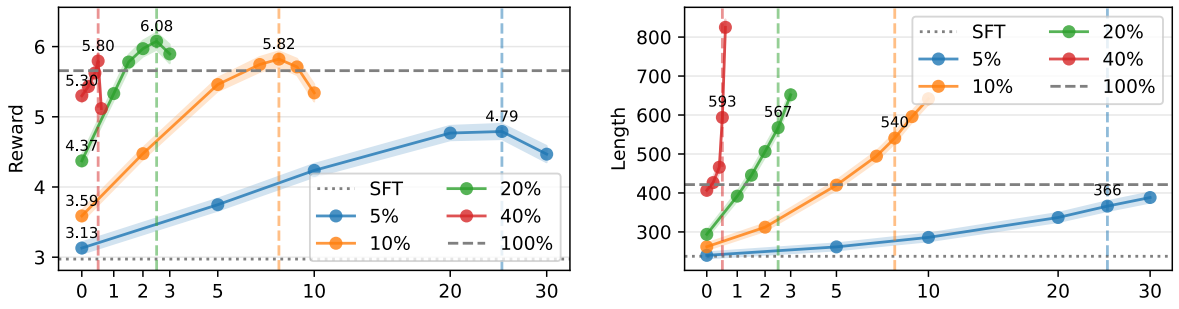



Figure 6: We train multiple $\mathcal{M}_1$ from the same initial $\mathcal{M}_0$, but with varying data sizes. We plot the reward scores (**left**) and output lengths (**right**) on the instructions of the UltraFeedback *development set* versus varying $\alpha$ values (x-axis). Note that $\alpha = 0$ indicates that ExPO is not applied.

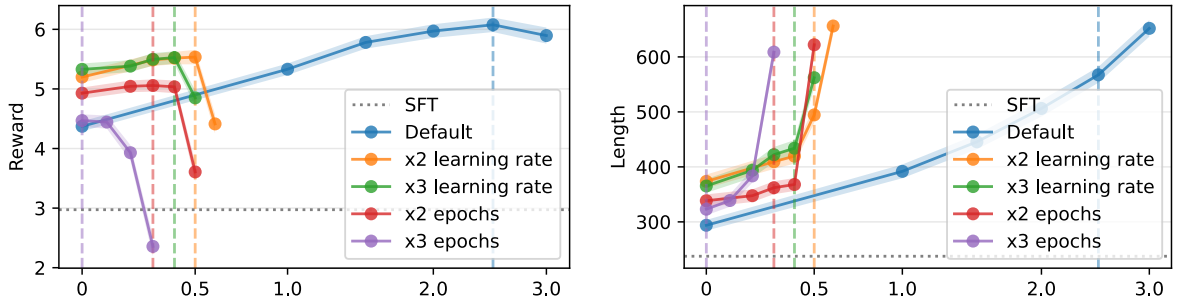



Figure 7: We train multiple $\mathcal{M}_1$ from the same initial $\mathcal{M}_0$ using the same 20% preference data, but with larger learning rates or for more epochs. We plot the reward scores (**left**) and output lengths (**right**) on the instructions of the UltraFeedback *development set* versus varying $\alpha$ values (x-axis). Note that $\alpha = 0$ indicates that ExPO is not applied.

Figure 7, we observe that increasing the learning rate or training epochs indeed somewhat improves the original reward score. However, it is still inferior to the optimal reward score achieved by ExPO under the default configuration, and also notably impairs the gains from ExPO (the peak points are lower than that of the default configuration). This is probably because the model is *overfitted* to the training data and similarly learns the spurious features (such as the length bias), thus failing to produce an accurate $\Delta\boldsymbol{\theta}$. The overfitting issue can also be evidenced by the right part of Figure 7. The models trained with larger learning rates or for more epochs become prone to generating longer outputs with a small $\alpha$, but do not obtain noticeable reward improvement (the left part of Figure 7). This suggests that $\Delta\boldsymbol{\theta}$ is very likely to contain the spurious length feature rather than the true human preference.

### 4.4. Discussion on Model Choices

Finally, we discuss the impact of model choices for $\mathcal{M}_0$ and $\mathcal{M}_1$ on the effectiveness of ExPO. In the experiments so far, we choose $\mathcal{M}_0$ as an SFT model and $\mathcal{M}_1$ as the model further trained for human preference alignment on top of $\mathcal{M}_0$. *Can other types of model combination $\mathcal{M}_0$ and $\mathcal{M}_1$, such as a Base and an SFT model, or two separately-trained RLHF models, be able to produce meaningful extrapolated models?* We experiment with the following combinations:

(1) **Base + SFT**: `mistral-7b-v0.1` (Jiang et al., 2023) as $\mathcal{M}_0$ and `mistral-7b-instruct-v0.1` as $\mathcal{M}_1$.

(2) **SFT 1 + SFT 2 (trained from different base models)**: `mistral-7b-instruct-v0.1` as $\mathcal{M}_0$ and `mistral-7b-instruct-v0.2` as $\mathcal{M}_1$.

(3) **SFT 1 + SFT 2 (same base)**: `openchat-3.5` (Wang et al., 2024) as $\mathcal{M}_0$ and `openchat-3.5-0106` as $\mathcal{M}_1$.

(4) **RLHF 1 + RLHF 2 (same base)**: `gemma-7b-it` (Team et al., 2024) as $\mathcal{M}_0$ and `gemma-1.1-7b-it` as $\mathcal{M}_1$. Note that it is not disclosed whether the two models are initialized from the same SFT model.

From Figure 8, (1) we find that extrapolating from two SFT models that are initialized from different base models can easily lead to the model collapse, due to that they do not satisfy the mode connectivity (Garipov et al., 2018; Entezari et al., 2022), (2) For the combination of Base and SFT, extrapolation degrades the performance, probably because training from Base to SFT does not naturally optimize for human preference and increase the alignment level $\Omega$. This is exactly why we need additional training for human preference alignment. (3&4) For two separately-trained SFT or

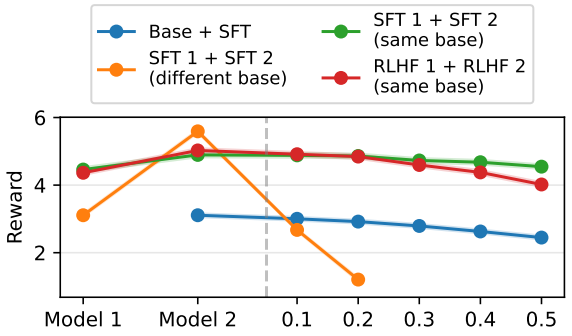



Figure 8: Reward scores of different model combinations on the instructions of the UltraFeedback *development set*, with $\alpha$ (x-axis) varying from 0.1 to 0.5.

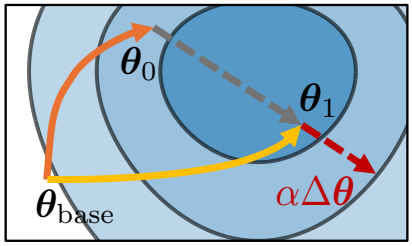



Figure 9: Extrapolation from two separately-trained models may not improve alignment, as their weight difference ($\Delta\boldsymbol{\theta}$) usually cannot indicate a direction in which the reward signal can be amplified.

RLHF models, we find that they also fail to benefit from model extrapolation. We speculate that this occurs because when $\mathcal{M}_1$ is not initialized from $\mathcal{M}_0$, the alignment level $\Omega$ does not monotonously increase along the path in the parameter space from $\boldsymbol{\theta}_0$ to $\boldsymbol{\theta}_1$. Instead, $\Omega$ may first reach an intermediate peak point and then decrease, as illustrated in Figure 9. Therefore, $\Delta\boldsymbol{\theta}$ fails to indicate a direction in which the reward signal can be amplified, even if the alignment level $\Omega(\boldsymbol{\theta}_1)$ is higher than $\Omega(\boldsymbol{\theta}_0)$. Overall, our method ExPO is generally applicable to the combination of an SFT model $\mathcal{M}_0$ and a model $\mathcal{M}_1$ further trained on top of the former, which is a very realistic combination choice, as modern LLMs that are trained to align with human preference are almost all initialized from their SFT checkpoints.

## 5. Conclusion

We present ExPO, a simple method to boost LLMs' alignment with human preference. By extrapolating from the weights of an aligned model and its initial SFT checkpoint, ExPO enables directly obtaining a better-aligned model without any additional training. We demonstrate the efficacy of ExPO in enhancing open-source LLMs across various model sizes (from 1.8B to 70B) and capabilities, suggesting significant improvement room for most open-source models. We also shed light on the essence of ExPO amplifying the reward signal learned in the alignment training through controlled experiments. Given its simplicity, efficiency, and scalability, we recommend ExPO as a promising approach for expediting the alignment of LLMs with human preference, which we believe deserves more future exploration.

**Limitations & Future Work** Our work is limited by the public accessibility to the SFT and DPO/RLHF checkpoints. Thus unfortunately, we are unable to experiment with the more representative LLMs like LLaMA-2/3 (Touvron et al., 2023; AI@Meta, 2024), Gemma (Team et al., 2024), and Qwen (Bai et al., 2023). We hope for more open-source efforts in increasing LLMs' transparency and accessibility. Outside the scope of our work, there are several problems that can potentially attract future research. First, since ExPO is currently based on the simplest uniform linear extrapolation (Equation 1, using the same $\alpha$ for all the model modules), future work can devise methods to adaptively search optimal $\alpha$ for different model modules. Second, although our work provides a basic explanation for ExPO (§ 2.2) and empirically demonstrates its effectiveness, future work can establish more profound theoretical foundations for its underlying mechanisms. Third, while we currently rely on an external reward model for searching $\alpha$, future work may get rid of such reliance by resorting to the inherent capability of the models $\mathcal{M}_1$ and $\mathcal{M}_0$ themselves. Finally, it would also be interesting to apply ExPO to multimodal LLMs like LLaVA (Liu et al., 2023) and other model architectures like Mamba (Gu & Dao, 2023).

## References

AI@Meta. Llama 3 model card, 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Akiba, T., Shing, M., Tang, Y., Sun, Q., and Ha, D. Evolutionary optimization of model merging recipes. *arXiv preprint arXiv:2403.13187*, 2024.

Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

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

Beeching, E., Fourrier, C., Habib, N., Han, S., Lambert, N., Rajani, N., Sanseviero, O., Tunstall, L., and Wolf, T. Open llm leaderboard, 2023. URL https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard.

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. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020.

Cai, Z., Cao, M., Chen, H., Chen, K., Chen, K., Chen, X., Chen, X., Chen, Z., Chen, Z., Chu, P., et al. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*, 2024.

Chen, H., He, G., Su, H., and Zhu, J. Noise contrastive alignment of language models with explicit rewards. *arXiv preprint arXiv:2402.05369*, 2024a.

Chen, Z., Deng, Y., Yuan, H., Ji, K., and Gu, Q. Self-play fine-tuning converts weak language models to strong language models. In *International Conference on Machine Learning*, 2024b.

Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., and Tafjord, O. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Cui, G., Yuan, L., Ding, N., Yao, G., Zhu, W., Ni, Y., Xie, G., Liu, Z., and Sun, M. Ultrafeedback: Boosting language models with high-quality feedback. *arXiv preprint arXiv:2310.01377*, 2023.

Dong, H., Xiong, W., Goyal, D., Zhang, Y., Chow, W., Pan, R., Diao, S., Zhang, J., SHUM, K., and Zhang, T. RAFT: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=m7p5O7zblY.

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

Dubois, Y., Galambosi, B., Liang, P., and Hashimoto, T. B. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*, 2024.

Entezari, R., Sedghi, H., Saukh, O., and Neyshabur, B. The role of permutation invariance in linear mode connectivity of neural networks. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=dNigytemkL.

Gao, L., Tow, J., Biderman, S., Black, S., DiPofi, A., Foster, C., Golding, L., Hsu, J., McDonell, K., Muennighoff, N., Phang, J., Reynolds, L., Tang, E., Thite, A., Wang, B., Wang, K., and Zou, A. A framework for few-shot language model evaluation, September 2021. URL https://doi.org/10.5281/zenodo.5371628.

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P., and Wilson, A. G. Loss surfaces, mode connectivity, and fast ensembling of dnns. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Goddard, C., Siriwardhana, S., Ehghaghi, M., Meyers, L., Karpukhin, V., Benedict, B., McQuade, M., and Solawetz, J. Arcee's mergekit: A toolkit for merging large language models. *arXiv preprint arXiv:2403.13257*, 2024.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Holtzman, A., Buys, J., Du, L., Forbes, M., and Choi, Y. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rygGQyrFvH.

Ilharco, G., Ribeiro, M. T., Wortsman, M., Schmidt, L., Hajishirzi, H., and Farhadi, A. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Ivison, H., Wang, Y., Pyatkin, V., Lambert, N., Peters, M., Dasigi, P., Jang, J., Wadden, D., Smith, N. A., Beltagy, I., et al. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *arXiv preprint arXiv:2311.10702*, 2023.

Izmailov, P., Wilson, A., Podoprikhin, D., Vetrov, D., and Garipov, T. Averaging weights leads to wider optima and better generalization. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, pp. 876–885, 2018.

Ji, J., Qiu, T., Chen, B., Zhang, B., Lou, H., Wang, K., Duan, Y., He, Z., Zhou, J., Zhang, Z., et al. Ai alignment: A comprehensive survey. *arXiv preprint arXiv:2310.19852*, 2023.

Ji, J., Chen, B., Lou, H., Hong, D., Zhang, B., Pan, X., Dai, J., and Yang, Y. Aligner: Achieving efficient alignment through weak-to-strong correction. *arXiv preprint arXiv:2402.02416*, 2024a.

Ji, K., He, J., and Gu, Q. Reinforcement learning from human feedback with active queries. *arXiv preprint arXiv:2402.09401*, 2024b.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. d. l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J., Zhang, H., and Stoica, I. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pp. 611–626, 2023.

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

Li, X., Zhang, T., Dubois, Y., Taori, R., Gulrajani, I., Guestrin, C., Liang, P., and Hashimoto, T. B. Alpacaeval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`, 2023a.

Li, X. L., Holtzman, A., Fried, D., Liang, P., Eisner, J., Hashimoto, T., Zettlemoyer, L., and Lewis, M. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12286–12312, Toronto, Canada, July 2023b. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.687. URL `https://aclanthology.org/2023.acl-long.687`.

Lin, S., Hilton, J., and Evans, O. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL `https://aclanthology.org/2022.acl-long.229`.

Lin, Y., Lin, H., Xiong, W., Diao, S., Liu, J., Zhang, J., Pan, R., Wang, H., Hu, W., Zhang, H., Dong, H., Pi, R., Zhao, H., Jiang, N., Ji, H., Yao, Y., and Zhang, T. Mitigating the alignment tax of rlhf, 2023.

Lin, Y., Tan, L., Hao, Y., Wong, H., Dong, H., Zhang, W., Yang, Y., and Zhang, T. Spurious feature diversification improves out-of-distribution generalization. In *International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=d6H4RBi7RH`.

Liu, A., Sap, M., Lu, X., Swayamdipta, S., Bhagavatula, C., Smith, N. A., and Choi, Y. DExperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6691–6706, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.522. URL `https://aclanthology.org/2021.acl-long.522`.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL `https://openreview.net/forum?id=w0H2xGHlkw`.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=Bkg6RiCqY7`.

Lu, S., Liu, H., Celikyilmaz, A., Wang, T., and Peng, N. Open-domain text evaluation via contrastive distribution modeling. In *International Conference on Machine Learning*, 2024.

OpenAI. `https://chat.openai.com.chat`, 2022.

OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

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

Park, R., Rafailov, R., Ermon, S., and Finn, C. Disentangling length from quality in direct preference optimization. *arXiv preprint arXiv:2403.19159*, 2024.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL `https://openreview.net/forum?id=HPuSIXJaa9`.

Rajbhandari, S., Rasley, J., Ruwase, O., and He, Y. Zero: memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1–16, 2020.

Sakaguchi, K., Bras, R. L., Bhagavatula, C., and Choi, Y. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., and Hashimoto, T. B. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

Team, G., Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S., Sifre, L., Rivière, M., Kale, M. S., Love, J., et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

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

Tran, H., Glaze, C., and Hancock, B. Iterative dpo alignment. Technical report, Snorkel AI, 2023.

Tunstall, L., Beeching, E., Lambert, N., Rajani, N., Huang, S., Rasul, K., Rush, A. M., and Wolf, T. The alignment handbook. https://github.com/huggingface/alignment-handbook, 2023a.

Tunstall, L., Beeching, E., Lambert, N., Rajani, N., Rasul, K., Belkada, Y., Huang, S., von Werra, L., Fourrier, C., Habib, N., et al. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*, 2023b.

Utans, J. Weight averaging for neural networks and local resampling schemes. In *Proc. AAAI-96 Workshop on Integrating Multiple Learned Models. AAAI Press*, pp. 133–138. Citeseer, 1996.

Wang, G., Cheng, S., Zhan, X., Li, X., Song, S., and Liu, Y. Openchat: Advancing open-source language models with mixed-quality data. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=AOJyfhWYHf.

Wang, Y., Kordi, Y., Mishra, S., Liu, A., Smith, N. A., Khashabi, D., and Hajishirzi, H. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13484–13508, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.754. URL https://aclanthology.org/2023.acl-long.754.

Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning*, pp. 23965–23998. PMLR, 2022.

Xiong, W., Dong, H., Ye, C., Wang, Z., Zhong, H., Ji, H., Jiang, N., and Zhang, T. Iterative preference learning from human feedback: Bridging theory and practice for RLHF under KL-constraint. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2024. URL https://openreview.net/forum?id=w3oJXMAQx2.

Yu, L., Yu, B., Yu, H., Huang, F., and Li, Y. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *International Conference on Machine Learning*, 2024.

Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., and Choi, Y. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472.

Zhao, Y., Khalman, M., Joshi, R., Narayan, S., Saleh, M., and Liu, P. J. Calibrating sequence likelihood improves conditional language generation. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=0qSOodKmJaN.

Zheng, C. Chat templates for huggingface large language models. https://github.com/chujiezheng/chat_templates, 2024.

Zheng, C., Ke, P., Zhang, Z., and Huang, M. Click: Controllable text generation with sequence likelihood contrastive learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1022–1040, Toronto, Canada, July 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.65. URL https://aclanthology.org/2023.findings-acl.65.

Zheng, C., Yin, F., Zhou, H., Meng, F., Zhou, J., Chang, K.-W., Huang, M., and Peng, N. On prompt-driven safeguarding for large language models. In *International Conference on Machine Learning*, 2024.

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*, 2023b. URL https://openreview.net/forum?id=uccHPGDlao.

Zhu, B., Frick, E., Wu, T., Zhu, H., Ganesan, K., Chiang, W.-L., Zhang, J., and Jiao, J. Starling-7b: Improving llm helpfulness & harmlessness with rlaif, November 2023.

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. *arXiv preprint arXiv:1909.08593*, 2019.

## A. Related Work

**LLM Alignment** Modern LLMs are typically first pre-trained on massive textual corpora (resulting in a Base model) (Brown et al., 2020; Touvron et al., 2023; AI@Meta, 2024) and then trained to align with human expectations (OpenAI, 2022; 2023; Ji et al., 2023). The alignment process generally contains two stages. In the first stage, an LLM is supervisedly fine-tuned (SFT) on *demonstration outputs* and learns to follow human instructions (Wang et al., 2023; Taori et al., 2023; Zheng, 2024). In the second stage, the LLM is trained to learn *human preference* and assign higher probabilities to human-preferred outputs over the disfavored ones. This is usually implemented in the fashion of reinforcement learning (RL) (Ouyang et al., 2022; Bai et al., 2022) or contrastive learning (Zhao et al., 2023; Zheng et al., 2023a; Rafailov et al., 2023), as exemplified by the reinforcement learning from human feedback (RLHF) (Ziegler et al., 2019) and direct preference optimization (DPO) (Rafailov et al., 2023) algorithms, respectively. However, as the model size increases (from 7B, 13B to 70B or larger), the computational resources required for alignment training also become extremely expensive (Ji et al., 2024a). For instance, training a 7B model via DPO has commonly required 4 or 8 A100 80GB GPUs, which can be unaffordable for open-source community users. Our work proposes the ExPO method to boost the alignment of LLMs with human preference in a simple, efficient, and scalable manner.

**Model Merging and Interpolation** Model merging is a recent focal technique for building powerful LLMs based on existing ones (Akiba et al., 2024; Yu et al., 2024; Goddard et al., 2024). It aims to integrate multiple models fine-tuned from the same base model into a unified one that retains the respective strengths. The simplest form of model merging is *model interpolation*, also known as model/weight averaging (Izmailov et al., 2018; Lin et al., 2024; Wortsman et al., 2022; Lin et al., 2023), which builds upon the mode connectivity of neural networks (Garipov et al., 2018; Entezari et al., 2022). Our work is inspired by the phenomenon that interpolation usually results in in-between performance compared to the original models, as observed in previous literature (Izmailov et al., 2018; Lin et al., 2024; Wortsman et al., 2022) and our experiments in Figure 1. The proposed ExPO method has a similar idea of blending model weights, but works under a distinct premise and goal. Rather than integrating the strengths of multiple strong models, ExPO starts from two relatively weaker models and aims to produce an overall stronger one.

There is another line of work that improves text generation by blending the token prediction distributions of multiple language models during the inference time (Liu et al., 2021; Li et al., 2023b; Lu et al., 2024). They share somewhat similar forms to model merging, but operate on output logits rather than model weights. Besides, they can suffer from decreased generation efficiency due to the interference with the inference process, and the increased exposure bias of different models. Our proposed ExPO method, as well as the work in model merging, bypasses these issues by producing a new single model.

## B. Broader Impacts and Safeguards

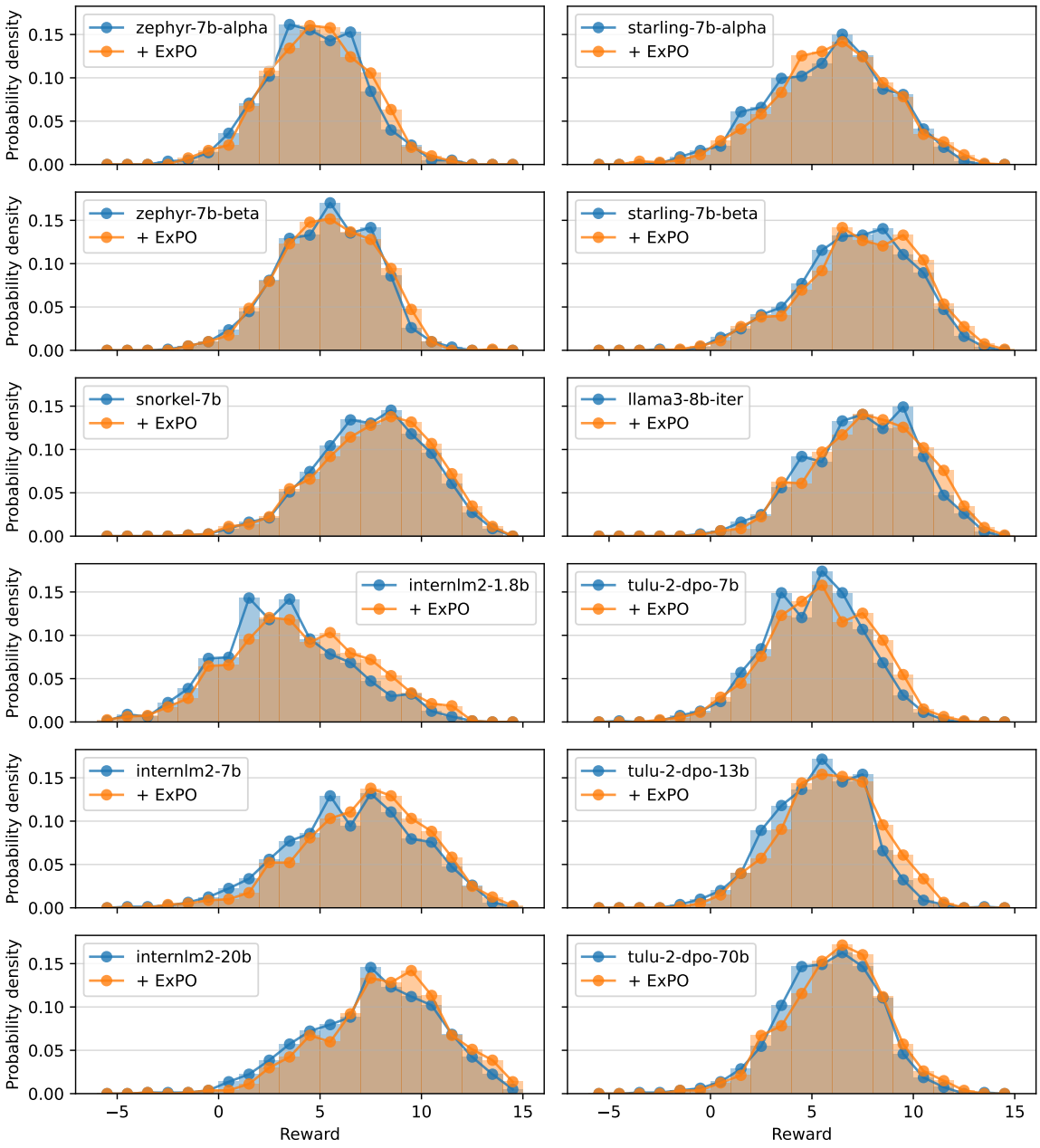
Our work aims to improve the alignment of LLMs with human preference when they follow human instructions. It can facilitate the development of more helpful AI assistants. On the other hand, the increased utility of LLMs also faces risks of dual use. For instance, they may be asked to assist with malicious queries like falsifying information or causing damages. For real-world deployment, it is essential that LLMs undergo additional safety training and learn to recognize and refuse harmful queries (OpenAI, 2023; Touvron et al., 2023). Furthermore, they should be equipped with necessary moderation mechanisms, such as safeguard classifiers or guardrail prompts (AI@Meta, 2024; Jiang et al., 2023; Zheng et al., 2024).

## C. Open-Source Models Used in This Work

| Model | HuggingFace Model ID |
|---|---|
| reward model | weqweasdas/RM-Mistral-7B |
| mistral-7b-sft-alpha | HuggingFaceH4/mistral-7b-sft-alpha |
| zephyr-7b-alpha | HuggingFaceH4/zephyr-7b-alpha |
| mistral-7b-sft-beta | HuggingFaceH4/mistral-7b-sft-beta |
| zephyr-7b-beta | HuggingFaceH4/zephyr-7b-beta |
| openchat-3.5 | openchat/openchat_3.5 |
| starling-7b-alpha | berkeley-nest/Starling-LM-7B-alpha |
| openchat-3.5-0106 | openchat/openchat-3.5-0106 |
| starling-7b-beta | Nexusflow/Starling-LM-7B-beta |
| mistral-7b-instruct-v0.2 | mistralai/Mistral-7B-Instruct-v0.2 |
| snorkel-7b | snorkelai/Snorkel-Mistral-PairRM-DPO |
| llama3-8b-sft | RLHFlow/LLaMA3-SFT |
| llama3-8b-iter | RLHFlow/LLaMA3-iterative-DPO-final |
| internlm2-1.8b-sft | internlm/internlm2-chat-1_8b-sft |
| internlm2-1.8b | internlm/internlm2-chat-1_8b |
| internlm2-7b-sft | internlm/internlm2-chat-7b-sft |
| internlm2-7b | internlm/internlm2-chat-7b |
| internlm2-20b-sft | internlm/internlm2-chat-20b-sft |
| internlm2-20b | internlm/internlm2-chat-20b |
| tulu-2-7b | allenai/tulu-2-7b |
| tulu-2-dpo-7b | allenai/tulu-2-dpo-7b |
| tulu-2-13b | allenai/tulu-2-13b |
| tulu-2-dpo-13b | allenai/tulu-2-dpo-13b |
| tulu-2-70b | allenai/tulu-2-70b |
| tulu-2-dpo-70b | allenai/tulu-2-dpo-70b |
| zephyr-7b-sft | alignment-handbook/zephyr-7b-sft-full |
| zephyr-7b-dpo | alignment-handbook/zephyr-7b-dpo-full |
| mistral-7b-v0.1 | mistralai/Mistral-7B-v0.1 |
| mistral-7b-instruct-v0.1 | mistralai/Mistral-7B-Instruct-v0.1 |
| gemma-7b-it | google/gemma-7b-it |
| gemma-1.1-7b-it | google/gemma-1.1-7b-it |

## D. Calculated Reward Histograms on the AlpacaEval 2.0 Instructions

For the DPO/RLHF models in § 3, we draw their reward distributions on the AplacaEval 2.0 Instructions, which are calculated by the aforementioned reward model. As shown below, ExPO generally shifts the distribution toward the higher-reward direction (i.e., the right-hand direction in the figures).



## E. Breakdowns of Open LLM Leaderboard Evaluation Results

| | ARC | HellaSwag | MMLU | GSM8K | Winogrande | TruthfulQA |
|---|---|---|---|---|---|---|
| zephyr-7b-alpha | 61.0 | 84.0 | 61.4 | 14.0 | 78.6 | 57.9 |
| + ExPO | 60.8 | 84.3 | 60.6 | 28.3 | 78.1 | 60.9 |
| zephyr-7b-beta | 62.0 | 84.5 | 61.1 | 11.4 | 78.1 | 57.4 |
| + ExPO | 62.3 | 84.5 | 61.0 | 27.3 | 77.7 | 58.3 |
| starling-7b-alpha | 63.7 | 84.9 | 64.7 | 62.3 | 80.4 | 46.3 |
| + ExPO | 63.9 | 84.8 | 64.6 | 61.6 | 80.4 | 46.4 |
| starling-7b-beta | 67.2 | 83.5 | 65.1 | 66.6 | 81.3 | 55.5 |
| + ExPO | 67.9 | 83.6 | 65.3 | 65.7 | 81.4 | 57.2 |
| snorkel-7b | 66.1 | 85.6 | 60.7 | 36.1 | 76.5 | 69.6 |
| + ExPO | 66.3 | 85.7 | 60.9 | 34.8 | 76.4 | 69.8 |
| llama3-8b-iter | 64.8 | 83.8 | 66.4 | 67.3 | 79.2 | 62.2 |
| + ExPO | 66.0 | 84.2 | 66.3 | 59.6 | 79.3 | 64.0 |
| internlm2-1.8b | 43.1 | 60.5 | 46.9 | 30.4 | 62.8 | 42.2 |
| + ExPO | 42.5 | 60.1 | 46.6 | 31.2 | 63.0 | 42.4 |
| internlm2-7b | 57.9 | 78.8 | 58.4 | 27.1 | 72.6 | 56.6 |
| + ExPO | 57.8 | 78.7 | 57.9 | 30.5 | 72.5 | 58.3 |
| internlm2-20b | 62.7 | 82.5 | 66.4 | 61.3 | 79.7 | 54.8 |
| + ExPO | 62.7 | 82.5 | 66.1 | 62.8 | 79.6 | 56.3 |
| tulu-2-dpo-7b | 57.2 | 81.0 | 52.0 | 27.3 | 74.0 | 55.9 |
| + ExPO | 58.0 | 81.3 | 52.0 | 26.7 | 74.7 | 59.6 |
| tulu-2-dpo-13b | 61.5 | 84.6 | 57.7 | 38.3 | 77.5 | 59.0 |
| + ExPO | 62.7 | 85.1 | 57.5 | 38.8 | 77.9 | 63.7 |
| tulu-2-dpo-70b | 72.1 | 89.0 | 69.8 | 62.6 | 83.3 | 65.8 |
| + ExPO | 72.7 | 89.3 | 69.6 | 59.4 | 83.2 | 70.0 |

## F. Implementation Details

For response generation in § 3 and 4, we employ the `vllm` (Kwon et al., 2023) library for high-throughput inference. We use top-$k$ ($k = 40$) and nucleus sampling (Holtzman et al., 2020) ($p = 0.9$) with a temperature of 0.7. To avoid repetition in generated texts, we set both the factors of presence penalty and frequency penalty to 0.1. **We adopt the same decoding hyperparameters with the sampling random seed set to 42 for all the evaluated models across all the experiments**, except in the evaluation of MT-Bench and Open LLM Leaderboard, as they have their own sets of decoding hyperparameters.

For model training in § 4, we adopt the global batch size 128 and gradient accumulation steps 4. We train the models on 4 A100 80GB GPUs, with ZeRO-3 offload (Rajbhandari et al., 2020) and gradient checkpointing for reducing GPU memory usage. We set the learning rate to 5e-7, with the cosine scheduling and warmup ratio of 0.1, and use the AdamW (Loshchilov & Hutter, 2019) optimizer to train the models for one epoch. For DPO, we follow `zephyr-7b-dpo` and set $\beta$ to 0.01.

For hyperparameter search in § 3 and 4, we perform grid search on the values of $\alpha$. We use the obtained model to generate responses on the UltraFeedback development set, score the responses with the reward model, and choose the optimal $\alpha$ corresponding to the highest average score. We list below the search range and the optimal $\alpha$ in our experiments.

| DPO/RLHF Model | SFT Checkpoint | Search Range | Optimal $\alpha$ |
|---|---|---|---|
| `zephyr-7b-alpha` | `mistral-7b-sft-alpha` | [0.1, 0.2, 0.3, 0.4, 0.5] | 0.3 |
| `zephyr-7b-beta` | `mistral-7b-sft-beta` | [0.1, 0.2, 0.3, 0.5] | 0.1 |
| `starling-7b-alpha` | `openchat-3.5` | [0.1, 0.2, 0.3, 0.5] | 0.2 |
| `starling-7b-beta` | `openchat-3.5-0106` | [0.1, 0.3, 0.4, 0.5] | 0.5 |
| `snorkel-7b` | `mistral-7b-instruct-v0.2` | [0.1, 0.2, 0.3, 0.4, 0.5] | 0.3 |
| `llama3-8b-iter` | `llama3-8b-sft` | [0.1, 0.2, 0.3, 0.4, 0.5] | 0.3 |
| `internlm2-1.8b` | `internlm2-1.8b-sft` | [0.1, 0.3, 0.4, 0.5] | 0.5 |
| `internlm2-7b` | `internlm2-7b-sft` | [0.1, 0.3, 0.4, 0.5] | 0.5 |
| `internlm2-20b` | `internlm2-20b-sft` | [0.1, 0.3, 0.4, 0.5] | 0.5 |
| `tulu-2-dpo-7b` | `tulu-2-7b` | [0.1, 0.3, 0.4, 0.5] | 0.5 |
| `tulu-2-dpo-13b` | `tulu-2-13b` | [0.1, 0.3, 0.4, 0.5] | 0.5 |
| `tulu-2-dpo-70b` | `tulu-2-70b` | [0.1, 0.3, 0.4, 0.5] | 0.5 |
| DPO (5% data) | `zephyr-7b-sft` | [5, 10, 20, 25, 30] | 25 |
| DPO (10% data) | `zephyr-7b-sft` | [2, 5, 7, 8, 9, 10] | 8 |
| DPO (20% data) | `zephyr-7b-sft` | [1.0, 2.0, 2.5, 3.0] | 2.5 |
| DPO (40% data) | `zephyr-7b-sft` | [0.2, 0.4, 0.5, 0.6] | 0.5 |
| `zephyr-7b-dpo` | `zephyr-7b-sft` | [0.1, 0.2, 0.3, 0.4, 0.5] | 0.3 |

## G. Results of Further Training on the Development Set Data

In § 4, one may be concerned that the UltraFeedback development set (1K data) is used to select optimal $\alpha$ in ExPO but is not involved in improving the baselines where ExPO is not applied, which may lead to unfair comparison. We thus further train these baselines on the 1K development data, and calculate the expected reward score on the development set. Note that ExPO only uses the instructions of the development set, while the further training for baseline models uses both the instructions and preference labels. In the table below, we show that further training on the development set still results in inferior performance to simply applying ExPO.

| | Original Reward | + Training on Dev | + ExPO, *no training* |
|---|---|---|---|
| DPO (5% data) | 3.13 | 3.33 (+0.20) | **4.79** **(+1.66)** |
| DPO (10% data) | 3.59 | 3.73 (+0.14) | **5.82** **(+2.23)** |
| DPO (20% data) | 4.37 | 4.46 (+0.09) | **6.08** **(+1.71)** |
| DPO (40% data) | 5.30 | 5.33 (+0.03) | **5.80** **(+0.50)** |
| DPO (100% data) | 5.66 | 5.64 (-0.02) | **5.81** **(+0.15)** |