Preference Tuning with Human Feedback on Language, Speech, and Vision Tasks: A Survey

Genta Indra Winata[*1] GENTA.WINATA@CAPITALONE.COM
Hanyang Zhao[*2] HZ2684@COLUMBIA.EDU
Anirban Das[*1] ANIRBAN.DAS3@CAPITALONE.COM
Wenpin Tang[2] WT2319@COLUMBIA.EDU
David D. Yao[2] DDY1@COLUMBIA.EDU
Shi-Xiong Zhang[1] SHIXIONG.ZHANG@CAPITALONE.COM
Sambit Sahu[1] SAMBIT.SAHU@CAPITALONE.COM
[1] *Capital One* [2] *Columbia University*

Abstract

Preference tuning is a crucial process for aligning deep generative models with human preferences. This survey offers a thorough overview of recent advancements in preference tuning and the integration of human feedback. The paper is organized into three main sections: 1) **introduction and preliminaries**: an introduction to reinforcement learning frameworks, preference tuning tasks, models, and datasets across various modalities: language, speech, and vision, as well as different policy approaches, 2) **in-depth exploration of each preference tuning approach**: a detailed analysis of the methods used in preference tuning, and 3) **applications, discussion, and future directions**: an exploration of the applications of preference tuning in downstream tasks, including evaluation methods for different modalities, and an outlook on future research directions. Our objective is to present the latest methodologies in preference tuning and model alignment, enhancing the understanding of this field for researchers and practitioners. We hope to encourage further engagement and innovation in this area.

Keywords: preference tuning, human preference, reinforcement learning, multi-modality, multilingual, large language models, vision language models, speech language models, generative models, survey, DPO, RLHF.

Contents

[*] These authors contributed equally.

1 Introduction

Learning from human feedback is a crucial step in aligning generative models with human preferences to generate output that closely resembles human speech and writing. Despite the powerful learning capabilities of generative models in self-supervised learning, these models frequently misinterpret instructions, leading to hallucinations in generation (Ji et al., 2023a; Yao et al., 2023a). Additionally, ensuring the safety of the generated content remains a significant challenge for these models. Extensive research on preference tuning using human feedback has demonstrated that adversarial samples can be utilized to jailbreak systems (Rando and Tramèr, 2023; Wei et al., 2024). Ideally, generative models need to be controlled to ensure that their outputs are safe and do not cause harm. Models often exhibit unintended behaviors, such as fabricating facts (Chen and Shu, 2023; Sun et al., 2024), producing biased or toxic text (Hartvigsen et al., 2022), or failing to follow user instructions (Ji et al., 2023b; Tonmoy et al., 2024). Additionally, maintaining the privacy of data is crucial to ensure the safe operation of models and protect user privacy (Brown et al., 2022). In the text-to-image generation task, large-scale models often struggle to produce images that are well-aligned with text prompts (Feng et al., 2022), particularly in compositional image generation (Liu et al., 2022; Lee et al., 2023) and coherent generation (Liu et al., 2023a). Similarly, in text-to-speech tasks, Zhang et al. (2024a); Chen et al. (2024a) integrate subjective human evaluation into the training loop to better align synthetic speech with human preferences.

The application of preference tuning has been widely used in language tasks by training instruction-tuned large language models (LLMs), such as Llama (Touvron et al., 2023b; Dubey et al., 2024), Phi (Abdin et al., 2024), Mistral (Jiang et al., 2023a), Nemotron (Parmar et al., 2024; Adler et al., 2024), Gemma (Team et al., 2024). Commercial models like GPT-4 (Achiam et al., 2023), Gemini (Team et al., 2023; Reid et al., 2024), Claude (Anthropic, 2024), Command-R, and Reka (Ormazabal et al., 2024) have leveraged human preference alignment to enhance their performance. Alignment of LLM improves task-specific skills, coherence, fluency, and helps avoid undesired outputs. Additionally, alignment research has benefited multilingual LLMs, such as Aya (Aryabumi et al., 2024; Üstün et al., 2024), BLOOMZ, and mT0 (Muennighoff et al., 2023), as well as regional LLMs like Cendol (Cahyawijaya et al., 2024) and SEALLM (Nguyen et al., 2023). Common approaches to achieving LLM alignment involve reinforcement learning techniques that guide language models to follow preferred samples by maximizing rewards. Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017) is the initial approach that is used to align models with human preference, which is further applied to the deep learning space that has been popularized by its successes in LLMs (Ouyang et al., 2022; Bai et al., 2022a) via PPO (Schulman et al., 2017), REINFORCE (Kool et al., 2019), Online Directed Preference Optimization (online DPO) (Guo et al., 2024a), and Supervised Fine-Tuning (SFT)-like approach (Dong et al., 2023). It typically involves three key aspects: human feedback collection, reward modeling, and *online* RL for policy optimization. Recent methods, however, allow for training the reward model alongside the policy model in an *offline* manner, as demonstrated by DPO (Rafailov et al., 2024), and jointly training with offline and online policies training (Zhao et al., 2023). Moreover, preference tuning is also applied to vision-text tasks, and has been shown to improve the representation of both image and text using the alignment score of image and text embeddings (Ramesh et al., 2022; Saharia et al., 2022; Yu et al., 2022b) measured by pre-trained vision-text models, such as CLIP (Radford et al., 2021) and CoCa (Yu et al., 2022a). Wu et al. (2023c) utilize LoRA (Hu et al., 2021) to align Stable Diffusion (Lee et al., 2023), a vision-text pre-trained model. The application in speech has not been much explored, and there is only a handful works in the literature. Zhang et al. (2024a) focus on investigating alignment between codes and the text.

In this paper, we survey the recent advances of preference tuning with human feedback in different modalities. It provides not only a comprehensive introduction including preliminaries to get readers familiar with the topic, but also an in-depth review on the latest proposed approaches and in-depth discussions. To summarize, the paper comprises the following contributions:

- We provide a comprehensive overview of preference tuning for models on different modalities, such as language, speech, and vision tasks, and expand our survey to all existing preference tuning methods, including reinforcement learning (RL) approaches.
- We formulate and taxonomize a systematic framework and classification for preference tuning for deep generative models from the existing literature.
- We present various applications of preference tuning to improve generation aspects using human feedback. We also describe the automatic and human-based evaluations to measure the quality of generation in deep generative models.
- We discuss the opportunities and future directions for preference tuning.

Through this survey, we aim to present the recent methodologies on preference tuning and alignment for deep generative models, enabling researchers and practitioners to better understand this topic and further innovate.

2 Preliminaries

This section outlines the preliminaries of preference tuning, including the formal definitions of the tasks and the notations used throughout this paper. Additionally, we provide a taxonomy for classifying preference tuning methods.

2.1 Tasks and Definition

In general, the entire preference tuning mechanism for generative models can be formulated as a RL problem described as follows.

2.1.1 RL Framework Concepts

Policy Model The policy model π_θ is a generative model that takes in an input prompt x and returns a sequence of output or probability distributions y. We define a generative model as a policy model π_θ where it is parameterized by θ with a policy model π. Given a prompt x, a generative model generates an output y as following:

$$\pi_\theta(y|x) = \prod_t \pi_\theta(y_t|x, y_{<t}), \tag{1}$$

where y_t is the t-th token in the response and $y_{<t}$ is tokens in the response before y_t. For example, for the text-based tasks, the input prompt is a text sequence x and the output is a probability distribution over text vocabulary of LLM y; and for the vision-text-based tasks, such as text-to-image tasks, the input x is the text sequence, and y is the generated image.

Reward Model The reward model (RM) processes both the input x and the target y, passing them through the model to obtain a reward $r_\theta(y|x)$, which reflects the notion of preferability. This preferability score can also be interpreted as a relative score assigned to the target y given the input x. Less preferred outcomes receive a lower score compared to more preferred samples.

Action Space The action refers to all tokens corresponding to the vocabulary of generative models. For text tasks, the action space encompasses the entire vocabulary of the LLM. For vision tasks (similarly for speech tasks), the action space consists of real values representing the image, for example, the next hierarchy in diffusion generative models (if understanding diffusion models as Hierarchical Variational Autoencoders (Luo, 2022)).

Environment The distribution encompasses all possible input token sequences for generative models. In text-based tasks, these input token sequences correspond to text sequences, highly depending on the sampling methods for the inference. In vision tasks, they correspond to possible images.

2.1.2 Preference Data

In the preference tuning pipeline, we utilize the supervised data $\mathcal{D}_{\text{sft}}$ and the preference data $\mathcal{D}_{\text{pref}}$. We denote the supervised data $\mathcal{D}_{\text{sft}} = [(x^1, y^1), \cdots, (x^M, y^M)]$ as a list of input and label pairs. Specifically for the text SFT data, x can be represented as prompts. The prompt $x^i = (I^i, F^i, Q^i)$ consists of the concatenation of an instruction I^i, few-shot samples F^i, and a query Q^i. Then, we denote the preference data $\mathcal{D}_{\text{pref}} = [(x^1, y_w^1, y_l^1), \cdots, (x^N, y_w^N, y_l^N)]$, a list of input x^i with preferred response y_w^i and dispreferred response y_l^i, and they are either sampled from the reference policy model π_{ref} or collected by human annotation. Generally, given the preference data, we can obtain a reward r associated to the response with the input.

2.1.3 Terminology and Notation

Table 1 lists the common notations used in this survey paper. The table serves as a quick reference guide for understanding the mathematical expressions and technical terms used throughout the paper.

2.2 Taxonomy

We define the following categories for all of the preference tuning approaches as shown in Table 2. Figure 1 shows the five categories we study in this survey paper and described in the following:

Sampling Likewise in the literature of RL, we categorize the methods based on how we sample the data and use them to train or obtain the reward: *offline* and *online* human alignments. The categorization is related to how we compute the reward and use it in the policy models. In online human alignment setting, the agent that collects a batch of examples by interacting with the environment and uses them to update the policy. The reward of the examples can be collected by the reward model or samples generated by the policy model. While for the offline human alignment setting, the data are collected from

Name	Notation	Description
Input Sequence	x	Input sequence that is passed to the model.
Output Sequence	y	Expected label or output of the model.
Dispreferred Response	y_l	Negative samples for reward model training.
Preferred Response	y_w	Positive samples for reward model training.
Optimal Policy Model	π^*	Optimal policy model.
Policy Model	π_θ	Generative model that takes the input prompt and returns a sequence of output or probability distribution.
Reference Policy Model	π_{ref}	Generative model that is used as a reference to ensure the policy model is not deviated significantly.
Preference Dataset	$\mathcal{D}_{\text{pref}}$	Dataset with a set of preferred and dispreferred. responses to train a reward model.
SFT Dataset	$\mathcal{D}_{\text{sft}}$	Dataset with a set of input and label for supervised fine-tuning.
Loss Function	$\mathcal{L}$	Loss function.
Regularization Hyper-parameters	$\alpha, \beta_{\text{reg}}$	Regularization Hyper-parameters for preference tuning.
Reward	r	Reward score.
Target Reward Margin	γ	The margin separating the winning and losing responses.
Variance	β_i	Variance (or noise schedule) used in diffusion models.

Table 1: Table of Terminology and Notation.

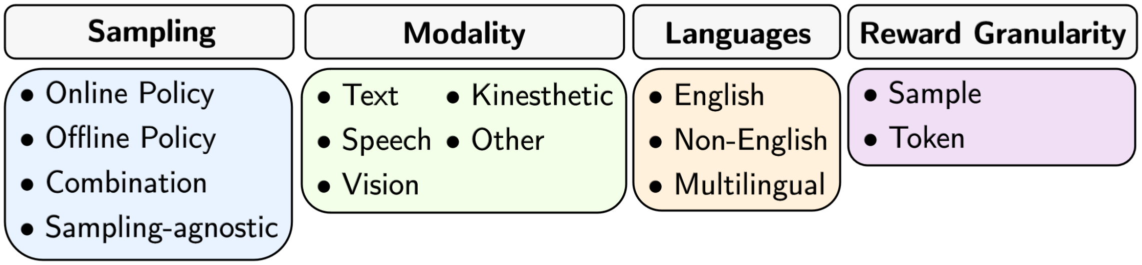

Figure 1: Taxonomy of the Preference Tuning methods.

offline human demonstrations. For online methods, we also categorize the methods as either *on-policy* when the behaviour policy is the same as the optimization policy, or *off-policy* if the behaviour policy is different.

Modality We study the use of preference tuning on various modality, such as text, speech, vision, kinesthetic and others if we are not able to classify them. In the latest advancement of NLP, the idea of RL has been further explored to language and speech tasks, even in multi-modal tasks, such as vision-text. Thus, it is essential to categorize the papers by the extend of the study in terms of the modality, such as text, speech, vision, and vision-text.

Language We explore the preference tuning application on different languages. In this case, we categorize the method by English, non-English, and multilingual.

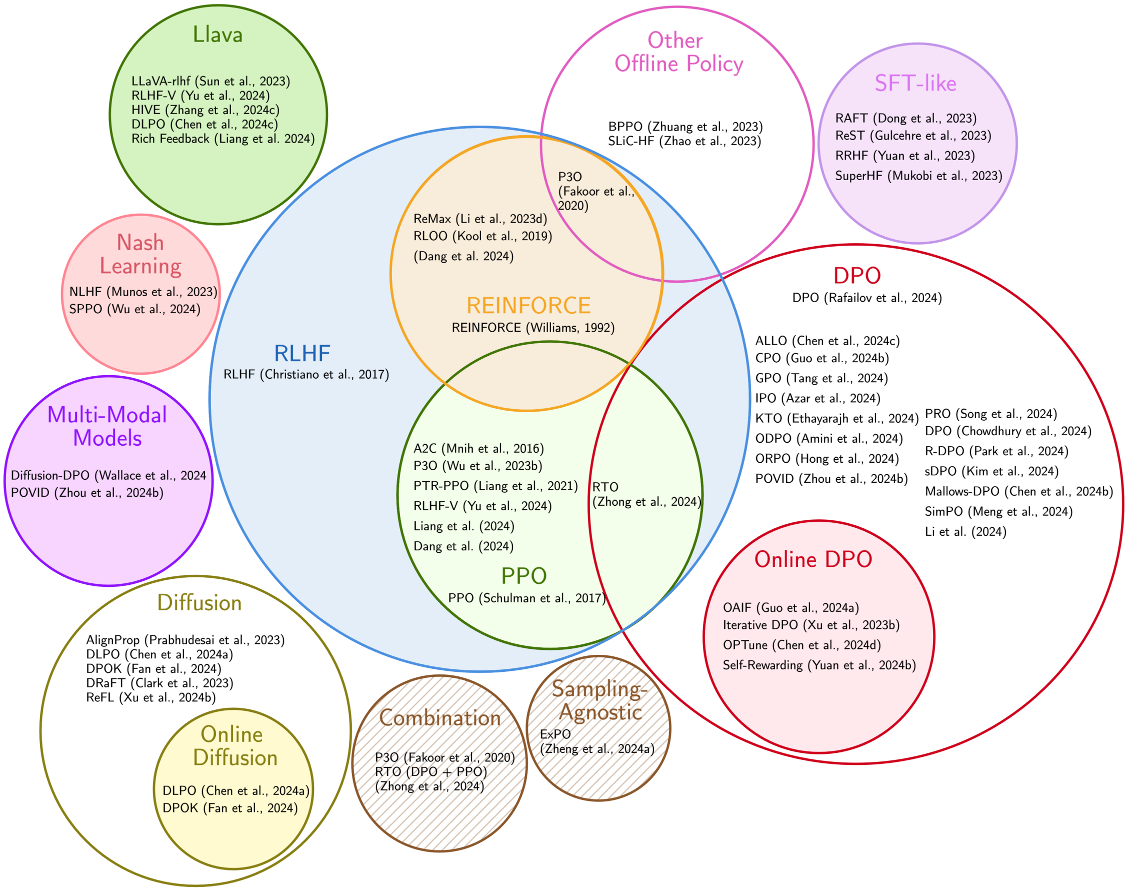
Llava
LLaVA-rlhf (Sun et al., 2023)
RLHF-V (Yu et al., 2024)
HIVE (Zhang et al., 2024c)
DLPO (Chen et al., 2024c)
Rich Feedback (Liang et al. 2024)
Other
Offline Policy
BPPO (Zhuang et al., 2023)
SLiC-HF (Zhao et al., 2023)
SFT-like
RAFT (Dong et al., 2023)
ReST (Gulcehre et al., 2023)
RRHF (Yuan et al., 2023)
SuperHF (Mukobi et al., 2023)
P3O
(Fakoor et al.,
2020)
ReMax (Li et al., 2023d)
RLOO (Kool et al., 2019)
(Dang et al. 2024)
Nash
Learning
NLHF (Munos et al., 2023)
SPPO (Wu et al., 2024)
REINFORCE
REINFORCE (Williams, 1992)
RLHF
RLHF (Christiano et al., 2017)
DPO
DPO (Rafailov et al., 2024)
ALLO (Chen et al., 2024c)
CPO (Guo et al., 2024b)
GPO (Tang et al., 2024)
IPO (Azar et al., 2024)
KTO (Ethayarajh et al., 2024)
ODPO (Amini et al., 2024)
ORPO (Hong et al., 2024)
POVID (Zhou et al., 2024b)
PRO (Song et al., 2024)
DPO (Chowdhury et al., 2024)
R-DPO (Park et al., 2024)
sDPO (Kim et al., 2024)
Mallows-DPO (Chen et al., 2024b)
SimPO (Meng et al., 2024)
Li et al. (2024)
Multi-Modal
Models
Diffusion-DPO (Wallace et al., 2024
POVID (Zhou et al., 2024b)
A2C (Mnih et al., 2016)
P3O (Wu et al., 2023b)
PTR-PPO (Liang et al., 2021)
RLHF-V (Yu et al., 2024)
Liang et al. (2024)
Dang et al. (2024)
RTO
(Zhong et al., 2024)
PPO
PPO (Schulman et al., 2017)
Online DPO
OAIF (Guo et al., 2024a)
Iterative DPO (Xu et al., 2023b)
OPTune (Chen et al., 2024d)
Self-Rewarding (Yuan et al., 2024b)
Diffusion
AlignProp (Prabhudesai et al., 2023)
DLPO (Chen et al., 2024a)
DPOK (Fan et al., 2024)
DRaFT (Clark et al., 2023)
ReFL (Xu et al., 2024b)
Online
Diffusion
DLPO (Chen et al., 2024a)
DPOK (Fan et al., 2024)
Combination
P3O (Fakoor et al., 2020)
RTO (DPO + PPO)
(Zhong et al., 2024)
Sampling-
Agnostic
ExPO
(Zheng et al., 2024a)

Figure 2: Preference Tuning methods. The circles with shaded areas represent off-policy methods, while the unshaded circles denote on-policy methods. The overlapping area signifies methods that incorporate both on-policy and off-policy approaches. The policy-agnostic circle indicates methods that are applicable to either on-policy or off-policy scenarios. The combination circle represents methods that integrate both online and off-policy strategies.

Reward Granularity In the preference tuning, the reward can be computed in different granularity levels. The granularity levels can be expanded into two: sample- and token-level. The token-level for each modality may differ, for example, in text tasks, we can use subwords from vocabulary as tokens. And, in vision tasks, patches of image are tokens.

3 Preference Tuning

In this section, we cover the general framework to train preference-tuned generative models. As shown in Table 3, the preference tuning training framework typically begins with the supervised fine-tuning (SFT) stage, during which the generative model is trained to excel

Method	Modality					Languages			Reward Granularity	
	Text	Speech	Vision	Kinesthetic	Other	EN	Non-EN	Multi.	Sample	Token
Online Policy										
RLHF (Christiano et al., 2017)										
PPO (Schulman et al., 2017)										
RLHF-PPO (Ouyang et al., 2022)	✓	×	×	×	×	✓	×	×	✓	×
AI Feedback (Bai et al., 2022b)	✓	×	×	×	×	✓	×	×	✓	×
P3O (Wu et al., 2023b)	✓	×	×	×	×	✓	×	×	✓	×
MaxMin-RLHF (Chakraborty et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
RLHF Workflow (Dong et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
(Dang et al., 2024)	✓	×	×	×	×	✓	✓	✓	×	×
REINFORCE (Williams, 1992)										
ReMax (Li et al., 2023f)	✓	×	×	×	×	✓	×	×	✓	×
RLOO (Ahmadian et al., 2024a)	✓	×	×	×	×	✓	×	×	✓	×
Online DPO										
OAIF (Guo et al., 2024a)	✓	×	×	×	×	✓	×	×	✓	×
Iterative DPO (Xu et al., 2023b)	✓	×	×	×	×	✓	×	×	✓	×
OPTune (Chen et al., 2024d)	✓	×	×	×	×	✓	×	×	✓	×
Self-Rewarding (Yuan et al., 2024b)	✓	×	×	×	×	✓	×	×	✓	×
Nash-Learning										
NLHF (Munos et al., 2023)	✓	×	×	×	×	✓	×	×	✓	×
SPPO (Wu et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
SFT-like										
RAFT (Dong et al., 2023)	✓	×	×	×	×	✓	×	×	✓	×
ReST (Gulcehre et al., 2023)	✓	×	×	×	×	✓	×	×	✓	×
RRHF (Yuan et al., 2023)	✓	×	×	×	×	✓	×	×	✓	×
SuperHF (Mukobi et al., 2023)	✓	×	×	×	×	✓	×	×	✓	×
Multi-Modal Models										
Diffusion (Schulman et al., 2017)										
DDPO (Black et al., 2024)	✓	×	✓	×	×	✓	×	×	✓	×
DPOK (Fan et al., 2024)	✓	×	✓	×	×	✓	×	×	✓	×
ReFL (Xu et al., 2024b)	✓	×	✓	×	×	✓	×	×	✓	×
DRaFT (Clark et al., 2023)	✓	×	✓	×	×	✓	×	×	✓	×
AlignProp (Prabhudesai et al., 2023)	✓	×	✓	×	×	✓	×	×	✓	×
LLaVA (Liu et al., 2024b)										
LLaVA-rlhf (Sun et al., 2023)	✓	×	✓	×	×	✓	×	×	✓	×
RLHF-V (Yu et al., 2024)	✓	×	✓	×	×	✓	×	×	✓	×
HIVE (Zhang et al., 2024c)	✓	×	✓	×	×	✓	×	×	✓	×
DLPO (Chen et al., 2024c)	✓	×	✓	×	×	✓	×	×	✓	×
Rich Feedback (Liang et al., 2024)	✓	×	✓	×	×	✓	×	×	✓	×
Offline Policy										
BPPO (Zhuang et al., 2023)	×	×	×	✓	×	×	×	×	✓	×
Multi-Modal Models										
Diffusion-DPO (Wallace et al., 2024)	✓	×	✓	×	×	✓	×	×	✓	×
POVID (Zhou et al., 2024b)	✓	×	✓	×	×	✓	×	×	✓	×
Offline DPO (Rafailov et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
ALLO (Chen et al., 2024g)	✓	×	×	×	×	✓	×	×	×	✓
CPO (Guo et al., 2024b)	✓	×	×	×	×	✓	✓	✓	✓	×
GPO (Tang et al., 2024b)	✓	×	×	×	×	✓	×	×	✓	×
IPO (Azar et al., 2024)	×	×	×	×	✓	×	×	×	✓	×
KTO (Ethayarajh et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
ODPO (Amini et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
ORPO (Hong et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
PRO (Song et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
R-DPO (Park et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
rDPO (Chowdhury et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
sDPO (Kim et al., 2024a)	✓	×	×	×	×	✓	×	×	✓	×
Mallows-DPO (Chen et al., 2024b)	✓	×	×	×	×	✓	×	×	✓	×
SimPO (Meng et al., 2024)	✓	×	×	×	×	✓	×	×	✓	×
(Li et al., 2024)	✓	×	×	×	×	✓	✓	✓	✓	×
SLiC-HF (Zhao et al., 2023)	✓	×	×	×	×	✓	×	×	✓	×
Combination										
P3O (Fakoor et al., 2020)	×	×	✓	×	×	×	×	×	✓	×
RTO (DPO + PPO) (Zhong et al., 2024)	✓	×	×	×	×	✓	×	×	×	✓
Sampling-Agnostic										
ExPO (Zheng et al., 2024a)	✓	×	×	×	×	✓	×	×	✓	×

Table 2: Preference Tuning methods. The categorization based on the methods under study and it does not limit the extension of the method to other domains or modalities.

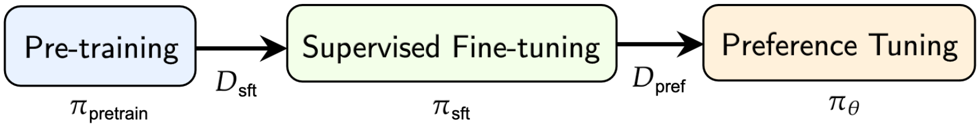

Figure 3: Training stages.

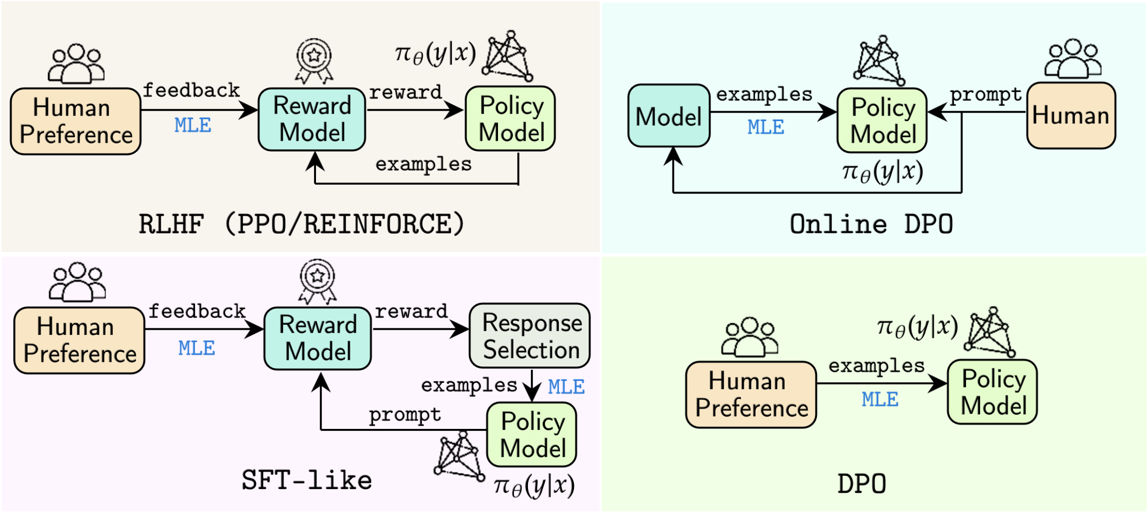

Figure 4: Preference Tuning methods for online algorithms, such as RLHF (PPO/REINFORCE), Online DPO, and SFT-like, and offline methods, such as DPO.

at next-token prediction or use an instruction-tuned model as the base initialized model. The SFT focuses on improving the model capability to generate tokens as it guides the model on how an generative model should response to a prompt input. Once the model is able to properly generate fluent text sequences, the model is further aligned by further policy optimization via RL. The alignment is useful to guide the model to answer with a appropriate manner based on the preference objective. This step is a necessary training stage to make sure the model generation aligned to human preference, thus, the model will act more human-like. Notably, the human alignment stage can also be jointly trained alongside SFT.

3.1 Training Phases

The training phases for preference tuning are described as follows.

Reward Model	**Sizes**	**Model Base**	**Datasets**
Single Objective			
BTRM Qwen2	7B△	Qwen2	UNK
Eurus-RM (Yuan et al., 2024a)	7B△	Mistral	UltraInteract, UltraFeedback, UltraSafety
FsfairX-LLama3-v0.1 (Dong et al., 2023)	8B△	Llama3	UNK
GRM-llama3-8B-sftreg (Yang et al., 2024)	8B△	Llama3	Preference 700K
GRM-llama3-8B-distill (Yang et al., 2024)	8B△	Llama3	Preference 700K
InternLM2 (Cai et al., 2024)	1.8B△, 7B△, 20B△	UNK	UNK
SteerLM-Llama3 (Wang et al., 2024b)	70B△	Llama3	HelpSteer2
Nemotron-4-340B-Reward (Adler et al., 2024)	340B△	Nemotron4	HelpSteer2
Pair-preference-model-LLamA3-8B (Dong et al., 2024)	8B△	LLama3	RLHFlow Pair Preference
Starling-RM-34B	34B△	Yi-34B-Chat	Nectar
UltraRM (Cui et al., 2023)	13B△	Llama2	UltraFeedback
Multi-Objective			
ArmoRM-Llama3-8B-v0.1 (Wang et al., 2024a)	8B△	Llama3	HelpSteer, UltraFeedback, BeaverTails-30k CodeUltraFeedback, Prometheus, Argilla-Capybara Argilla-OpenOrca, Argilla-Math-Preference

Table 3: Reward Models.

3.1.1 Supervised Fine-Tuning (SFT)

On the preference tuning, a generative model with trainable weights θ normally starts by SFT via maximum likelihood (MLE) using teacher forcing and cross-entropy loss. The training is done using the supervised fine-tuning dataset $\mathcal{D}_{\text{sft}}$. The objective is to maximize the log probability of a set of human demonstrations. The generative model is trained to generate the label by predicting the next token y_{t+1} given the input x, current and previous label tokens $y_{t:<t}$. During the SFT, we utilize an attention mask applying to the entire context x and $y_{t:<t}$, and avoid applying attention to future tokens. The trained model denoted π_θ^{sft} and it is often to be used to initialize reward model and policy model π_θ.

3.1.2 Reward Modeling

The reward model $r_\phi(x, y)$ can be trained either separately (offline) or jointly trained with the policy model π_θ (online). Table 3 shows the list of reward models.

Single Objective Reward Model Bradley-Terry Reward Model (Bradley and Terry, 1952) is a pairwise comparison between two samples. It estimates the probability that the pairwise comparison $i \succ j$, which indicates a strong preference of i over j, is true as:

$$P(i \succ j) = \frac{\exp s_i}{\exp s_i + \exp s_j}, \tag{2}$$

where s_i and s_j are latent variables representing sample i and sample j, respectively. Thus, given the preference dataset $\mathcal{D}_{\text{pref}} = \{x^i, y_w^i, y_l^i\}_{i=1}^N$, we could obtain an estimation of the reward model $r_\phi(x, y)$ by minimizing the negative log-likelihood loss:

$$\begin{aligned}\mathcal{L}(\phi) &= -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{pref}}} \log P(y_w \succ y_l \mid x) &(3)\\ &= -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{pref}}} \log \sigma(r_\phi(x, y_w) - r_\phi(x, y_l)), &(4)\end{aligned}$$

which σ denotes the logistic function, i.e., $\sigma(x) := (1 + e^{-x})^{-1}$.

Multi-Objective Reward Model Absolute-Rating Multi-Objective Reward Model (ArmoRM) (Wang et al., 2024a) is a two-stage approach that first trains a multi-objective RM and then learns a gating layer that scalarizes reward objectives in a mixture-of-experts way. Each example consists of an input x and output y with k-dimensional rating vector, where each dimension corresponds to a reward objective. A concatenation of input and output $x \oplus y$ is passed through the model f_θ with a linear regression layer w, which outputs a k-dimensional rating prediction. The model is trained with regression loss:

$$\min_{\theta, w} \mathbb{E}_{x,y,r \in \mathcal{D}} \| w^\top f_\theta(x \oplus y) - r \|_2^2. \tag{5}$$

Then, it learns a mixture-of-experts gating function, g_ϕ, which is implemented as a shallow MLP. This MLP takes the representation of the input x and outputs a k-dimensional vector, which is then processed by a softmax function. During the training of the gating layer, the backbone and the regression layer are kept frozen. Only the gating layer is trained using the Bradley-Terry loss, augmented with an additional scaling variable.

3.1.3 Preference Alignment using Reinforcement Learning

While SFT has led to markedly improved performance, there is still a misalignment between SFT objective and the ultimate target of generating high-quality outputs as determined by humans. Stiennon et al. (2020); Ouyang et al. (2022) propose reinforcement learning from human feedback (RLHF) to further align language models with human intent. RLHF pipeline starts with the stage of modeling the rewards from human preferences, known as reward modeling stage, by maximizing the likelihood of preferences under the ground truth assumption. After obtaining the RM, RLHF further trains the Language Model policy via Reinforcement Learning to maximize the score given by the RM. Proximal Policy Optimization (PPO) was commonly chosen as the RL algorithm to update the policy because of its great sample efficiency.

3.1.4 Joint Training

Recent works also proposed that two stages of SFT and RLHF can be simplied as one stage with a weighted combination of the two loss functions and even lead to better performance. The key takeaway is to treat the preferred answer in the Human Alignment/RLHF stage as the SFT target, e.g., SLiC-HF (Zhao et al., 2023).

3.2 Datasets

The dataset sources for SFT and preference tuning can be collected from various sources, such as human and LLMs feedback. Table 4 shows the list of SFT and alignment text data labeled by the data source either they are collected by human or synthetically generated by LLM.

3.2.1 SFT Datasets

The SFT data is useful for training LM on high-quality input-output demonstration pairs. This is usually conducted for the foundation model as initialization. The SFT data can be in the form of prompts with various format.

LLM-Generated Datasets Taori et al. (2023) propose Alpaca, a dataset with demonstrations generated using OpenAI's GPT-3 text-davinci-003 model. The instruction data

Dataset	# Samples	Usecase		Data Source		Annotation
	or (# Tokens) or [Byte Size]	SFT	Alignment	Human	LLM	Human
Alpaca (Taori et al., 2023)	52k	✓	×	✓	✓	×
Alpaca-CoT$^{\triangle}$	127.5M†	✓	×	✓	✓	✓
Aya Dataset (Singh et al., 2024)	202k	✓	×	✓	×	✓
ChatAlpaca$^{\triangle}$	20k†	✓	×	✓	✓	×
BeaverTails (Ji et al., 2024)	30k, 330k	✓	✓	✓	✓	✓
Code-Alpaca$^{\triangle}$	20k	✓	×	✓	✓	×
CodeUltraFeedback$^{\triangle}$	10k	✓	✓	✓	✓	✓
Dolly (Conover et al., 2023)	15k	✓	×	✓	×	✓
FLAN collection (Longpre et al., 2023)	UNK‡	✓	×	✓	×	✓
HC3 (Guo et al., 2023)	24.3k	✓	✓	✓	✓	✓
HelpSteer2 (Wang et al., 2024b)	21k	✓	✓	✓	✓	✓
HH-RLHF (Bai et al., 2022a)	170k	×	✓	✓	×	✓
InstructionWild v2 (Ni et al., 2023)	110k	✓	×	✓	×	✓
LIMA (Zhou et al., 2024a)	1.3k	✓	×	✓	×	✓
Magpie (Air) (Xu et al., 2024d)	300k, 3M	✓	✓	✓	✓	✓
Magpie (Pro) (Xu et al., 2024d)	300k, 1M	✓	✓	✓	✓	✓
Natural Questions (Kwiatkowski et al., 2019)	323k	✓	×	✓	×	✓
Oasst1 (Köpf et al., 2024)	88.8k	✓	✓	✓	×	✓
Okapi (Lai et al., 2023)	4.3M*	✓	✓	✓	✓	×
P3 (Sanh et al., 2021)	122M	✓	×	✓	×	✓
Preference 700K$^{\triangle}$	700K	×	✓	UNK	UNK	UNK
Prometheus2 (Kim et al., 2024b)	200k	✓	✓	✓	✓	×
Prosocial-Dialog (Kim et al., 2022)	165.4k	✓	✓	✓	×	✓
RLHFlow Pair Preference$^{\triangle}$	700k	×	✓	✓	✓	✓
Self-instruct (Wang et al., 2023b)	197k	✓	×	✓	×	✓
ShareGPT	Multiple Versions	✓	✓	✓	✓	✓
StackExchange$^{\triangle}$	10.8M	✓	✓	✓	×	✓
Super-Natural Instructions (Wang et al., 2022)	5M	✓	✓	✓	✓	✓
UltraChat (Ding et al., 2023)	1.5M	✓	×	×	✓	✓
UltraFeedback (Cui et al., 2023)	64k	✓	✓	✓	✓	✓
WildChat (Zhao et al., 2024)	652k	✓	✓	✓	✓	✓
WizardLM (Xu et al., 2023a)	250k	✓	✓	✓	✓	✓
xP3 (Muennighoff et al., 2023)	78.8M	✓	×	✓	×	✓

Table 4: SFT and alignment text datasets. †The dataset is updated over the time and the number placed on the table is from the latest dataset released by the authors. ‡The exact size is unknown and some the datasets are no longer accessible. *The estimated number of translated and English instructions.

Dataset	# Samples	Usecase		Data Source		Annotation
	or (# Tokens) or [Byte Size]	SFT	Alignment	Human	LLM	Human
ImageRewardDB (Xu et al., 2024b)	137k+	✓	✓	✓	×	✓
Pick-a-pic (Kirstain et al., 2023)	500k+	×	✓	✓	✓	✓
RichHF-18K (Liang et al., 2024)	18k	×	✓	✓	×	✓

Table 5: SFT and alignment vision datasets. †The dataset is updated over the time and the number placed on the table is from the latest dataset released by the authors. ‡The exact size is unknown and some the datasets are no longer accessible. *The estimated number of translated and English instructions.

can be used to conduct instruction tuning for LLMs and allow them to follow instruction better. A version of Alpaca dataset with Chain-of-Thought (CoT) (Wei et al., 2022) and it is introduced to further improve the LLM's reasoning ability. Multi-turn datasets generated using LLMs are also created, such as ChatAlpaca, UltraChat (Ding et al., 2023), and WildChat (Zhao et al., 2024).

Human-Generated and Human-Annotated Datasets Using human-generated and human-annotated data are essential in training high-quality models. Zhou et al. (2024a) has shown quality is more important than quantity, as shown as using LIMA datasets that models trained only consist of 1,000 carefully human curated prompts and responses, without any reinforcement learning or human preference modeling can outperform models with much larger instruction-tuned datasets.

Dataset Collection FLAN collection (Longpre et al., 2023) is introduced to train a collection of tasks on top of T5 and PaLM models (Raffel et al., 2020). For training multilingual LMs, Cendol Collection (Cahyawijaya et al., 2024), ROOTS (Laurençon et al., 2022), and xP3 (Muennighoff et al., 2023) are used in SFT. Other potential datasets are crowd-sourcing datasets, although they are designed for SFT, but they can be useful resources for SFT, such as NusaCrowd (Cahyawijaya et al., 2023) and SEACrowd (Lovenia et al., 2024).

3.2.2 Human Preference Alignment Datasets

The human alignment data can be in the form of pair-wise or ranking format. We can have a set of preferred and dispreferred data $\mathcal{D}_{\text{pref}}$ for each input sample. For pairwise dataset, we collect pairs of preferred response y_w and dispreferred response y_l. In case of multiple responses, we can gather responses y_0, y_1, y_2, ... and ask humans to pick the best y_i from each. These datasets have been used to train reward models.

Conversational Datasets Several existing conversational datasets are instrumental in evaluating the quality of dialogue system or chatbot responses. Notable examples include HelpSteer2 (Wang et al., 2024b) and UltraFeedback (Cui et al., 2023). HelpSteer2 provides alignment scores across five different aspects—helpfulness, correctness, coherence, complexity, and verbosity—collected from human evaluators. UltraFeedback offers alignment scores for four aspects: instruction-following, truthfulness, honesty, and helpfulness. Additionally, HH-RLHF (Bai et al., 2022a) introduces datasets labeled with scores for helpfulness and harmlessness.

Code Datasets CodeUltraFeedback comprises 10,000 coding instructions, each annotated with four responses generated by a diverse pool of 14 LLMs (Weyssow et al., 2024). These responses are ranked based on five distinct coding preferences: instruction-following, complexity, style, readability, and another instance of instruction-following. The rankings are determined using GPT-3.5 as a judge, providing both numerical scores and detailed textual feedback.

3.3 Pre-trained Generative Models

We categorize pre-trained generative models into three main types: LMs, VLMs, and SLMs. Additionally, we classify these models based on their accessibility: **(1) Open Source:** The model and data are open and accessible, **(2) Open-Weight:** Only the model is accessible and some or all data are inaccessible, **(3) Close-weight and Close-source:** The model is a black-box and may only be accessible by API or service, and **(4) Close Access:** The

Model	Sizes	SFT/Pref. Tuning Langs.†	Model Base	SFT	Pref. Tuning
Open-source LM					
Aya-23 (Aryabumi et al., 2024)	8B, 35B	Multi. (23)	Dec-Only; Command R	✓	×
Aya-101 (Üstün et al., 2024)	13B	Multi. (101)	Enc-Dec; mT5	✓	×
Bactrian-X (Li et al., 2023b)	7B	Multi. (52)	Dec-Only; Llama1	✓	×
BART (Lewis et al., 2020)	139M, 406M	English	Enc-Dec	×	×
BLOOM (Le Scao et al., 2023)	560M, 1.1B, 1.7B, 3B, 7.1B, 176B	Multi. (46) + Code (13)	Dec-Only	✓	×
BLOOMZ (Muennighoff et al., 2023)	560M, 1.1B, 1.7B, 3B, 7.1B, 176B	Multi. (108) + Code (13)	Dec-Only; BLOOM	✓	×
Cendol (Cahyawijaya et al., 2024)	7B, 13B	Multi. (10)	Dec-Only; Llama2	✓	×
	300M, 580M, 1.2B, 3.7B, 13B	Multi. (10)	Enc-Dec; mT5	✓	×
FLAN-T5 (Longpre et al., 2023)	80M, 250M, 780M, 3B, 11B	English	Enc-Dec; T5	✓	×
Llama1 (Touvron et al., 2023a)	6.7B, 13B, 32.5B, 65.2B	English	Dec-Only	×	×
M2M-100 (Fan et al., 2021)	418M, 1.2B, 12B	Multi. (100)	Enc-Dec	✓	×
mBART (Liu et al., 2020)	406M	Multi. (25), Multi. (50)	Enc-Dec	✓	×
Megatron-LM (Shoeybi et al., 2019)	1.2B, 2.5B, 4.2B, 8.3B	English	Dec-Only; GPT-2	×	×
MPT (Instruct/Chat)△	7B, 30B	English	Dec-Only	✓	✓
mT0 (Muennighoff et al., 2023)	560M, 1B7, 3B, 7B1	Multi. (108) + Code (13)	Enc-Dec; mT5;	✓	×
OLMo (Groeneveld et al., 2024)	1B, 7B	English + Code	Dec-Only	×	×
OPT (Zhang et al., 2022)	125M, 350M, 1.3B, 2.7B, 6.7B, 13B, 30B, 66B, 175B	English	Dec-Only; Megatron-LM	×	×
Phi1 (Gunasekar et al., 2023)	1.3B	English	Dec-Only	×	×
Phi1.5 (Li et al., 2023e)	1.3B	English	Dec-Only	×	×
Pythia (Biderman et al., 2023)	70M, 160M, 410M, 1B, 1.4B, 2.8B, 6.9B, 12B	English	Decoder-Only; GPT-NeoX	×	×
SantaCoder (Allal et al., 2023)	1.1B	Code (3)	Dec-Only	✓	×
StarCoder (Li et al., 2023d)	15.5B	Code (80+)	Dec-Only	✓	×
T0 (Sanh et al., 2021)	3B, 11B	English	Enc-Dec; T5	✓	×
T5 (Raffel et al., 2020)	80M, 250M, 780M, 3B, 11B	English	Enc-Dec	✓	×
T5v1.1 (Raffel et al., 2020; Shazeer, 2020)	80M, 250M, 780M, 3B, 11B	English	Enc-Dec	×	×
WizardCoder (Luo et al., 2023)	7B, 13B, 15B, 33B	Code	Dec-Only	✓	×
Open-weight LM					
Alpaca△	7B	English	Dec-Only; Llama1	✓	×
C4AI Command-R (incl. Plus)△	35B, 104B	Multi. (13)	Dec-Only	✓	✓
DBRX△	132B	Multi. (UNK) + Code	MoE	✓	✓
DeepSeek-V2△	16B, 236B	Multi. (UNK) + Code	MoE	✓	✓
Falcon (Almazrouei et al., 2023)	7B, 40B, 180B	Multi. (2) + Code	Dec-Only	✓	×
Falcon2△	11B	Multi. (11) + Code	Dec-Only	✓	×
Gemma (Team et al., 2024)	2B, 7B	Multi. (UNK) + Code	Dec-Only	✓	✓
Gemma2△	9B, 27B	Multi. (UNK) + Code	Dec-Only	✓	✓
Llama2 (Touvron et al., 2023b)	7B, 13B, 70B	Multi. (UNK) + Code	Dec-Only	✓	✓
Llama3, Llama3.1 (Dubey et al., 2024)△	8B, 70B	Multi. (UNK) + Code	Dec-Only	✓	✓
LlaMAX (Lu et al., 2024)	7B, 8B	Multi. (102)	Dec-Only; Llama2, Llama3	✓	×
Mistral (Jiang et al., 2023a)	7B	Multi. (UNK) + Code	Dec-Only	✓	✓
Mixtral-MoE (Jiang et al., 2024)	8×7B, 8×22B	Multi. (UNK) + Code	MoE; Mistral	✓	✓
Nemotron-4 (15B) (Parmar et al., 2024)	15B	Multi. (53) + Code (43)	Dec-Only	×	×
Nemotron-4 (340B) (Adler et al., 2024)	340B	Multi. (53) + Code (43)	Dec-Only; Nemotron-4 (15B)	✓	✓
NLLB (Costa-jussà et al., 2022)	600M, 1.3B, 3.3B, 54.5B (MoE)	Multi. (200+)	Enc-Dec; M2M-100, MoE	✓	×
Phi3 (Abdin et al., 2024)	3.8B, 7B, 14B	Multi. (UNK) + Code	Dec-Only	✓	✓
Qwen (Bai et al., 2023)	1.8B, 7B, 14B, 72B	Multi. (100) + Code	Dec-Only	✓	✓
Snowflake Artic△	128 × 3.66B	Multi. (UNK) + Code	MoE	✓	✓
StableLM 2 (1.6B) (Bellagente et al., 2024)	1.6B	Multi. (7) + Code	Dec-Only	✓	✓
StableVicuna△	13B	English	Dec-Only; Vicuna	✓	✓
Vicuna (Chiang et al., 2023)	7B, 13B	English	Dec-Only; Llama1, Llama2	✓	×
Close-weight and Close-source LM					
Bard (Manyika and Hsiao, 2023)	UNK	UNK	UNK	✓	✓
Chinchilla (Hoffmann et al., 2022)	70B	English + Code	Dec-Only	×	×
Claude 3.5 Sonnet (Anthropic, 2024)	UNK	UNK	UNK	✓	✓
Command R (Plus)△	UNK	UNK	UNK	✓	✓
Gemini 1.0 (Team et al., 2023)	UNK	UNK	Dec-Only	✓	✓
Gemini 1.5 (Reid et al., 2024)	UNK	UNK	MoE; Gemini 1.0	✓	✓
Gopher (Rae et al., 2021)	280B	English + Code	Dec-Only	×	×
GPT-3 (Brown et al., 2020)	125M, ..., 175B	Multi. (UNK)	Dec-Only; GPT-2	×	×
GPT-3.5 (Instruct GPT) (Ouyang et al., 2022)	1.3B	UNK	Enc-Dec; GPT-3	✓	✓
GPT-4 (Achiam et al., 2023)	UNK	Multi. (UNK)	UNK	✓	✓
Reka (Ormazabal et al., 2024)	7B (Edge), 21B (Flash), UNK (Core)	Multi. (110)	Enc-Dec	✓	✓
Close-access LM					
AlexaTM (Soltan et al., 2022)	20B	Multi. (12)	Enc-Dec; BART	×	×
BloombergGPT (Wu et al., 2023a)	50.6B	English	Dec-Only; BLOOM	×	×
FLAN-PaLM (Longpre et al., 2023)	8B, 62B, 540B	Multi. (124+) + Code (24+)	UNK	✓	×
PaLM (Chowdhery et al., 2023)	8B, 62B, 540B	Multi. (124) + Code (24)	Dec-Only	×	×
PaLM2 (Anil et al., 2023)	400M, ..., 15B	Multi. (124+) + Code (24+)	UNK	✓	×

Table 6: Pre-trained Generative Language Models. †The languages do not include the languages seen by the base model.

model is inaccessible. We also categorize these models based on the datasets used for pre-

Model	Sizes	SFT/Pref. Tuning Langs.†	Model Base	SFT	Pref. Tuning
Open-weight SLM					
BAT (Zheng et al., 2024c)	7B	English	Enc-Dec	✓	×
SpeechGPT (Zhang et al., 2023a)	13B	English	Dec	✓	✓
Open-source SLM					
Close-weight and Close-source SLM					
Reka (Ormazabal et al., 2024)	7B (Edge), 21B (Flash), UNK (Core)	Multi. (110)	Enc-Dec	✓	✓

Table 7: Pre-trained Speech Language Models. †The languages do not include the languages seen by the base model.

Model	Sizes	SFT/Pref. Tuning Langs.†	Model Base	SFT	Pref. Tuning
Open-weight VLM					
Falcon 2 VLM	11B$^{\triangle}$	Multi. (11)	Enc-Dec	✓	×
InstructBLIP (Dai et al., 2023)	7B, 13B (Vicuna)	English	Enc-Dec	✓	×
	3B, 11B (FLAN-T5)	English	Enc-Dec	✓	×
InstructPix2Pix (Brooks et al., 2023)	UNK	English	UNK	✓	×
LLaVA 1.5 (Liu et al., 2024a)	7B, 13B	English	Enc-Dec	✓	×
LLaVA 1.6 (NeXT)	UNK$^{\triangle}$	English	Enc-Dec	✓	×
X-instructblip (Panagopoulou et al., 2023)	7B, 13B	English	Enc-Dec	✓	×
Phi3-Vision (Abdin et al., 2024)	4.2B	English	Enc-Dec	✓	×
Otter (Li et al., 2023a)	7B (Dec)	English	Enc-Dec	✓	×
MultiModal-GPT (Gong et al., 2023)	UNK	English	Enc-Dec	✓	×
Stable Diffusion v1.5 (Rombach et al., 2022)	UNK	English	Enc-Dec	✓	×
Video-LLaMA (Zhang et al., 2023b)	7B, 13B	English	Dec-Only	✓	×
Open-source VLM					
Close-weight and Close-source SLM					
Reka (Ormazabal et al., 2024)	7B (Edge), 21B (Flash), UNK (Core)	Multi. (110)	Enc-Dec	✓	✓
SORA (Liu et al., 2024c)	UNK	UNK	Enc-Dec	✓	✓

Table 8: Pre-trained Vision Language Models. †The languages do not include the languages seen by the base model.

training, specifically noting whether they are trained with Supervised Fine-Tuning (SFT) datasets or Human Preference Tuning datasets.

3.3.1 Language Models (LMs)

Table 6 shows the list of LMs categorized by the model accessibility and annotated with the model sizes, languages, model base, and fine-tuning methods applied to the model.

3.3.2 Speech Language Models (SLMs)

Table 7 shows the list of open-weight and open-source Speech Language Models (SLMs) categorized by the datasets and methods used in training.

3.3.3 Vision Language Models (VLMs)

Table 8 shows the list of open-weight and open-source Vision Language Models (VLMs) categorized by the datasets and methods used in training.

4 Online Alignment

In this section, we explore into human preference tuning using online methods, where data is continuously sampled. Online preference tuning involves real-time model updates as new data becomes available, enabling the model to dynamically adapt to evolving preferences and new information. This approach allows the alignment process to incorporate new data as it arrives and benefit from online exploration. We discuss the mechanisms of data collection, processing, and real-time model updates, emphasizing the benefits of managing non-stationary environments and enhancing model performance through continuous learning. Various techniques and strategies for implementing especially on-policy tuning are examined to provide a comprehensive understanding of its effective application in human preference tuning. We cover standard RL-based methods (e.g., PPO, which is online and on-policy), online DPO and SFT like algorithms (which can be on-policy or off-policy) and Nash Learning (or self-play) based algorithms.

4.1 Reinforcement Learning Human Feedback (RLHF)

In general, RLHF learns a reward function from human feedback and then optimize that reward function (Christiano et al., 2017). The training for RLHF involves three stages:

- The policy model π_θ interacts with the environment and the parameters of π_θ are updated via RL.

- The pairs of segments are selected from the output produced by the policy model π_θ, and send them to human annotators for comparison.

- The parameters are optimized using reward r to fit the comparisons collected from human.

According to Ziegler et al. (2019), the RLHF pipeline for LMs can be summarized as following:

- **Supervised Fine-Tuning:** A pre-trained LM is instruction-tuned using a dataset consisting of a given instruction prompt, and (typically) a human-written completion. The LM/policy is trained with a cross-entropy loss over the completion only. Often, the SFT model, denoted as sft is used to initialize both the reward model and the RLHF policy.

- **Reward Modeling:** RLHF leverages a reward model r_ϕ trained using a dataset of preferences $\mathcal{D}$. The reward model is trained using the following loss:

$$\text{loss}(r) = \mathbb{E}_{(x,\{y_i\}_i,b)\sim S}\left[\log \frac{e^{r(x,y_b)}}{\sum_i e^{r(x,y_i)}}\right]. \tag{6}$$

  or, for pairwise preferences,

$$\mathcal{L}_{\text{RM}}(\phi) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{pref}}} \log \sigma(r_\phi(x,y_w) - r_\phi(x,y_l)). \tag{7}$$

- **Reinforcement Learning:** In this stage, the learned reward model r_{ϕ^*} is used to provide online feedback in the optimization of the policy. In Ziegler et al. (2019); Stiennon et al. (2020); Ouyang et al. (2022), RLHF further maximizes average reward with an extra KL regularization term, i.e.:

$$\mathcal{L}_{\mathrm{RL}}(\phi) = \mathbb{E}_{x \sim \mathcal{D}, y \sim \pi(\cdot|x)} \left[r_{\phi^*}(x, y) - \beta_{\mathrm{reg}} \, \mathrm{KL}(\pi(\cdot \mid x) \mid \pi_{\mathrm{ref}}(\cdot \mid x)) \right], \tag{8}$$

where $\beta_{\mathrm{reg}} > 0$ is a hyper-parameter controlling the deviation from the reference policy $\pi_{\mathrm{ref}} = \pi^{\mathrm{SFT}}$.

RLHF proposes optimizing the policy model using the Advantage Actor-Critic (A2C) method (Mnih et al., 2016) for playing Atari games and Trust Region Policy Optimization (TRPO) (Mnih et al., 2015) for performing simulated robotics tasks. The reward model is trained using the Bradley-Terry Reward Model, which leverages pairwise preference datasets—essentially, pairs of preferred and dispreferred responses. There are various methods and variations for training RLHF, primarily categorized into two main approaches: RLHF and REINFORCE. In the following sections, we will describe these methods in detail.

4.1.1 Proximal Policy Optimization (PPO)

Initially in the original RLHF paper (Ziegler et al., 2019), they use PPO (Schulman et al., 2017) as their optimization strategy. PPO framework is a method for the human preference signals from external reward models with RLHF. The idea is to improve the current state of affairs by introducing an algorithm that attains the data efficiency and reliable performance of TRPO, while using only first-order optimization with a simpler clipped surrogate objective, omitting the expensive second-order optimization presented in TRPO using stochastic gradient ascent. Whereas standard policy gradient methods perform one gradient update per data sample, PPO (Schulman et al., 2017) proposes a novel objective function that enables multiple epochs of minibatch updates. It have some of the benefits of TRPO, but they are much simpler to implement and more efficient. For the optimization, KL-shaped reward (Ahmadian et al., 2024a) is useful as penalty-free optimization of the reward model leads to degradation in the coherence of the model. Optimizing this objective is equivalent to maximizing the following KL-shaped reward in expectation. There are a couple of variants of PPO: A2C (Mnih et al., 2016), P3O (Wu et al., 2023b), PTR-PPO (Liang et al., 2021), and RLHF-V (Yu et al., 2024).

Advantage Actor-Critic (A2C) A2C (Mnih et al., 2016) is an asynchronous variant of four RL algorithms that utilize parallel actor-learners to stabilize the effect of training of four methods.

Pairwise Proximal Policy Optimization (P3O) P3O (Wu et al., 2023b) is an on-policy RL algorithms that interleaves off-policy updates with on-policy updates. P3O uses the effective sample size between the behavior policy and the target policy to control how far they can be from each other and does not introduce any additional hyper-parameters.

Prioritized Trajectory Replay (PTR-PPO) PTR-PPO (Liang et al., 2021) is an on-policy deep reinforcement learning algorithms have low data utilization and require significant experience for policy improvement. The algorithm proposes a proximal policy optimization algorithm with PTR-PPO that combines on-policy and off-policy methods to improve sampling efficiency by prioritizing the replay of trajectories generated by old policies. The method is designed three trajectory priorities based on the characteristics of trajectories: the first two being max and mean trajectory priorities based on one-step empirical generalized advantage estimation (GAE) values and the last being reward trajectory priorities based on normalized undiscounted cumulative reward. Then, it is also incorporated the prioritized trajectory replay into the PPO algorithm, propose a truncated importance weight method to overcome the high variance caused by large importance weights under multistep experience, and design a policy improvement loss function for PPO under off-policy conditions.

RLHF-V RLHF-V (Yu et al., 2024) enhances MLLM trustworthiness via behavior alignment from fine-grained correctional human feedback. Specifically, RLHF-V collects human preference in the form of segment-level corrections on hallucinations, and performs dense direct preference optimization over the human feedback.

4.1.2 REINFORCE

ReMax ReMax (Li et al., 2023f) builds upon the well-known REINFORCE algorithm (Williams, 1987, 1992), leveraging three key properties of RLHF: fast simulation, deterministic transitions, and trajectory-level rewards. The name “ReMax” reflects its foundation in REINFORCE and its use of the argmax operator. ReMax modifies the gradient estimation by incorporating a subtractive baseline value as following:

$$\widetilde{g}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\left[s_\theta(x^i, a^i_{1:t}) \times (r(x^i, a^i_{1:T}) - b_\theta(x^i))\right], \tag{9}$$

where the action $a^i_t \sim \pi_\theta(\cdot|x^i, a^i_{1:t-1})$, and $b_\theta(x^i)$ is a baseline value. A typical choice for $b_\theta(x^i)$ is

$$b_\theta(x^i) = r(x^i, \bar{a}^i_{1:T}),\, \bar{a}^i_t \in \operatorname{argmax} \pi_\theta(\cdot|x^i, \bar{a}^i_{1:t-1}). \tag{10}$$

This baseline value can be obtained by greedily sampling a response and calculating the associated reward value.

REINFORCE Leave One-Out (RLOO) RLOO (Ahmadian et al., 2024a) extends the REINFORCE algorithm by leveraging multiple online samples to achieve unbiased variance reduction. It improves upon REINFORCE in two key ways: (1) The rewards from each sample can serve as a baseline for all other samples, and (2) Policy updates are performed using the average of gradient estimates from each sample, resulting in a variance-reduced multi-sample Monte Carlo (MC) estimate. This is the intuition behind the RLOO estimator, as following:

$$\frac{1}{k}\sum_{i=1}^{k}[R(y_{(i)}, x) - \frac{1}{k-1}\sum_{j\neq i} R(y_{(j)}, x)]\nabla \log \pi(y_{(i)}|x), \text{ for } y_{(1)}, ..., y_{(k)} \overset{i.i.d}{\sim} \pi_\theta(.|x), \tag{11}$$

where k refers to the number of online samples generated, RLOO$_k$ considers each $y_{(i)}$ individually and uses the remaining $k-1$ samples to create an unbiased estimate of the expected return for the prompt. This approach functions similarly to a parameter-free value function, but it is estimated at each training step.

4.2 Online Directed Preference Optimization (Online DPO)

4.2.1 Online AI Feedback (OAIF)

OAIF (Guo et al., 2024a) employs a LLM as an annotator during each training iteration. In this process, two responses are sampled from the current model, and the LLM annotator is prompted to select the preferred response, thereby providing real-time feedback. OAIF aims to gather preferences dynamically for responses generated by the language model being aligned. Given the prohibitive cost of using human feedback, this method leverages an LLM as an online annotator to collect preferences over pairs of responses sampled from the model π_θ during its alignment process. The objective for online DPO yields (please see detailed derivation of DPO in Section 5.1):

$$\begin{aligned}\mathcal{L}_{\text{OAIF}}\left(\pi_\theta;\pi_{\text{ref}}\right) :=& \\ &-\mathbb{E}_{x\sim\mathcal{D},(y_w,y_l)\sim\pi_{\theta\_}}\left[\log\sigma\left(\beta_{\text{reg}}\log\frac{\pi_\theta\left(y_w\mid x\right)}{\pi_{\text{ref}}\ \left(y_w\mid x\right)}-\beta_{\text{reg}}\log\frac{\pi_\theta\left(y_l\mid x\right)}{\pi_{\text{ref}}\ \left(y_l\mid x\right)}\right)\right],\end{aligned} \quad (12)$$

in which we note $\pi_{\theta\_}$ to show that preference pairs are generated under π_θ, but we further adopt a stop gradient to prevent it from getting into the loss objective for the gradient computation. The OAIF is illustrated in Algorithm 1 (OAIF algorithm in Guo et al. (2024a)), in which function ℓ can be log-sigmoid (DPO), square (IPO), or ReLU (SLiC) functions.

Algorithm 1 Online AI Feedback (OAIF) for Direct Alignment from Preference (DAP)

```
Input: Prompt dataset $\mathcal{D}_x = \{x_i\}_{i=1}^N$, an LLM annotator, SFT model $\pi_{\theta^0}$
for $t := 0$ to $T$ do
    Sample prompt $x \sim \mathcal{D}_x$
    Sample response pair $y_1, y_2 \sim \pi_{\theta^t}(\cdot|x)$
    Use LLM annotator to get preference pair $y_w$, $y_l$
    Update $\theta^t$ into $\theta^{t+1}$ using $\nabla_\theta \ell(x, y_w, y_l, \theta^t)$
end for
```

4.2.2 Iterative Directed Preference Optimization

Iterative DPO (Xu et al., 2023b; Xiong et al., 2024) has been proposed to narrow the gap between the performance offline preference optimization methods like DPO and online methods like RLHF, as RLHF still outperforms offline DPO. Different from DPO that used a fixed offline dataset, iterative DPO proposed to formulate the preference datasets by the generations of the current model and labelers, being either a pretrained reward model or LLM as a judge or the model to be trained itself through specific prompting (Yuan et al., 2024b), thus this pipeline usually appears at the same time with self-rewarding (Yuan et al., 2024b) methods (some paper will even call self-rewarding as iterative DPO methods). For each iteration, if the batch size for preference datasets utilized for policy optimization is only 1, then iterative DPO is essentially the same as online DPO or OAIF, except that the reference policy may be chosen as the last iterated policy instead of always being the SFT policy; otherwise iterative DPO is a hybrid method which combines offline learning in loss function optimization and online sampling in preference data generation. The reference model in the loss objective may differ between different methods, can be fixed SFT model Xiong et al. (2024) or last iterated model (Xu et al., 2023b; Yuan et al., 2024b) or some mixtures.

4.2.3 Online Preference Tuning (OPTune)

OPTune (Chen et al., 2024d) is an algorithm for efficient data generation in online RLHF. It improves both generation and training efficiency by selectively regenerating only the lowest-rewarded responses and employing a weighted DPO objective that prioritizes pairs with larger reward gaps. This approach significantly enhances the overall efficiency of the RLHF pipeline, setting the stage for the development of preference-aligned LLMs in a resource-efficient manner. The method enhances both data generation and training efficiency for online preference alignment. To minimize the cost of iterative data regeneration, it employs a straightforward yet effective reward-based prompt selection strategy, updating responses only for prompts with the lowest scores according to the reward model. Additionally, recognizing that converting scalar rewards to binary labels for the online DPO objective results in information loss, the method introduces a weighted DPO loss variant. This variant prioritizes learning from response pairs with larger reward gaps, further boosting online learning efficiency.

4.3 SFT-like

4.3.1 Rank Responses to align Human Feedback (RRHF)

RRHF (Yuan et al., 2023) is a method that evaluates sampled responses from various sources using the logarithm of conditional probabilities and aligns these probabilities with human preferences through ranking loss. This approach can utilize responses from multiple origins, including the model's own outputs, responses from other large language models, and human expert responses, to learn how to rank them effectively. The primary objective is to simplify the complex hyper-parameter tuning and extensive training resources required by PPO. Before training, RRHF samples responses from diverse sources, which can include model-generated responses from the model itself as well as pre-existing human-authored responses of varying quality. During training, RRHF scores these responses based on the log probability provided by the training language model. These scores are then aligned with human preference rankings or labels using ranking loss, ensuring that the model's outputs are better aligned with human preferences.

4.3.2 Reward rAnked FineTuning (RAFT)

RAFT (Dong et al., 2023) is the combination of ranking samples by rewards and SFT, which iteratively alternates among three steps: 1) The batch is sampled from the generative models; 2) The reward function is used to score the samples and filter them to get a filtered subset of high rewards; and 3) fine-tune the generative models on the filtered subset.

4.3.3 Reinforced Self-Training (ReST)

ReST (Gulcehre et al., 2023) is an RLHF algorithm aimed at aligning an LM's outputs with human preferences. It uses a learned reward function to model human preferences over sequences. In the Markov decision process underlying conditional language modeling, states represent partial sequences, and actions correspond to generated tokens. ReST divides the typical reinforcement learning pipeline into distinct offline stages for dataset growth and policy improvement. Initially, it fine-tunes a model to map input sequences to output sequences using a dataset of sequence pairs, optimizing with Negative Log-Likelihood (NLL) loss. Then, it creates a new dataset by augmenting the initial training dataset with samples generated by the model. In this phase, conditioning inputs are resampled from the original dataset, similar to self-training, but direct sampling is possible if accessible.

4.3.4 Supervised Iterative Learning from Human Feedback (SuperHF)

SuperHF (Mukobi et al., 2023) is an alignment algorithm that enhances data efficiency using a reward model and replaces PPO with a straightforward supervised fine-tuning loss. The core concept involves the language model generating its own training data by sampling a "superbatch" of outputs, filtering these through a reward model, and iteratively fine-tuning on each filtered completion. This method builds upon and unifies previous research by integrating two crucial components: (1) the Kullback-Leibler (KL) divergence penalty and (2) an iterative process of sampling and fine-tuning. Additionally, SuperHF is embedded within a Bayesian inference framework, demonstrating that both RLHF and SuperHF can be understood from a unified theoretical perspective that does not rely on reinforcement learning. This perspective naturally justifies the use of the KL penalty and the iterative approach.

4.4 Nash Learning

4.4.1 Nash Learning from Human Feedback (NLHF)

NLHF (Munos et al., 2023) is motivated to address the limitation of reward models (or essentially the Elo ratings) to represent the richness of human preferences as in RLHF. Instead of targeting at maximizing the (regularized) reward, NLHF takes the preference model as the 'first class citizen', and pursue 'a policy that consistently generates responses preferred over those generated by any competing policy'. Thus this policy is the Nash equilibrium of this preference model, the reason the method is named NLHF. Concretely, the (regularized) preference model for two policies π, π' is defined as:

$$\begin{aligned}&\mathcal{P}\left(\pi > \pi'\right) := \\ &\qquad \mathbb{E}_{x\sim\rho}\mathbb{E}_{y\sim\pi(\cdot|x),y'\sim\pi'(\cdot|x)}\left[\mathcal{P}\left(y > y' \mid x\right) - \beta_{\text{reg}}\log\frac{\pi(y \mid x)}{\mu(y \mid x)} + \beta_{\text{reg}}\log\frac{\pi'\left(y' \mid x\right)}{\mu\left(y' \mid x\right)}\right], \quad (13)\end{aligned}$$

and NLHF searches the Nash Equilibrium such that (denote μ as π_{ref} for simplicity here):

$$\pi^* := \arg\max_{\pi}\min_{\pi'}\mathcal{P}\left(\pi > \pi'\right) - \beta_{\text{reg}}\text{KL}_{\rho}(\pi, \mu) + \beta_{\text{reg}}\text{KL}_{\rho}\left(\pi', \mu\right). \quad (14)$$

For optimization, the Nash-MD algorithm proposed in NLHF used a geometric mixture between the current policy π_t and the reference policy μ as the competing policy in the place of π':

$$\pi_t^{\mu}(y) := \frac{\pi_t(y)^{1-\eta\beta_{\text{reg}}}\mu(y)^{\eta\beta_{\text{reg}}}}{\sum_{y'} \pi_t\left(y'\right)^{1-\eta\beta_{\text{reg}}} \mu\left(y'\right)^{\eta\beta_{\text{reg}}}}, \tag{15}$$

where η is a learning rate, and Nash-MD algorithm is a step of mirror descent relative to the regularized policy π_t^{μ}:

$$\pi_{t+1} := \arg\max_{\pi} \left[\eta\mathcal{P}\left(\pi > \pi_t^{\mu}\right) - \mathrm{KL}\left(\pi, \pi_t^{\mu}\right)\right], \tag{16}$$

which yields a closed-form solution that:

$$\log \pi_{t+1}(y) = \left[(1-\eta\beta_{\text{reg}}) \log \pi_t(y) + \eta\beta_{\text{reg}} \log \mu(y)\right] + \eta\mathcal{P}\left(y > \pi_t^{\mu}\right) + c, \tag{17}$$

where c is a normalization constant which is independent of y and the algorithm is proved to converge of rate $\frac{1}{T}$ under the tabular setting. For practical concern, when policy is a deep neural network beyond tabular setting, NLHF further proposes Nash-MD-PG motivated by Nash-MD, and the algorithm updates the policy with policy gradient:

$$\nabla_\theta \mathcal{P}_\tau\left(\pi_\theta > \pi'_{\theta_-}\right) = \mathbb{E}_{x\sim\rho, y\sim\pi_\theta(\cdot|x), y'\sim\pi'(\cdot|x)}\left[\widehat{g}\left(x, y, y'\right)\right], \tag{18}$$

where π'_{θ_-} denotes a stop-gradient on π'_θ with π'_θ being a geometric mixture

$$\log \pi'_\theta(y \mid x) := (1-\lambda) \log\left(\pi_\theta(y \mid x)\right) + \lambda \log(\mu(y \mid x)) + c(x), \tag{19}$$

in which λ is a mixing constant and

$$\widehat{g}\left(x, y, y'\right) := \nabla_\theta \log \pi_\theta(y \mid x)\left(\mathcal{P}\left(y > y' \mid x\right) - 1/2 - \beta_{\text{reg}} \mathrm{KL}\left(\pi_\theta(\cdot \mid x), \mu(\cdot \mid x)\right)\right), \tag{20}$$

respectively. NLHF also argues that, Nash equilibrium of the preference model is a solution that better aligns with the diversity of human preferences.

4.4.2 Self-Play Preference Optimization (SPPO)

SPPO (Wu et al., 2024) can be understood as a specific instance of NLHF by taking $\lambda = 0$, i.e., the reference policy is itself. The algorithm can be found in Algorithm 2, given an LLM judge:

4.5 Fine-tuning Diffusion Models

Given the popularity of diffusion based t2I models and its different nature of structural properties, we have the methods of fine-tuning diffusion models as a separate section of interest. We first briefly review the formulation of text-to-image diffusion generative models. For a more comprehensive background of diffusion models, we refer the interested readers to existing tutorial/survey papers (Luo, 2022; Cao et al., 2024; Yang et al., 2023; Tang and Zhao, 2024; Chen et al., 2024f; Chan, 2024). DDPM (Sohl-Dickstein et al., 2015; Ho et al., 2020) consider a sequence of positive noise scales $0 < \beta_1, \beta_2, \cdots, \beta_N < 1$, and perturb data by gradually adding noise through a stochastic process: for each training data point $x_0 \sim p_{\text{data}}(x)$, a discrete Markov chain $\{x_0, x_1, \cdots, x_N\}$ is constructed such that:

$$x_i = \sqrt{1-\beta_i}x_{i-1} + \sqrt{\beta_i}z_{i-1}, \quad i = 1, \cdots, N, \tag{21}$$

Algorithm 2 Self-Play Preference Optimization (SPPO)

```
Input: base policy $\pi_{\theta_0}$, preference oracle $\mathcal{P}$, learning rate $\eta$, number of generated samples $K$
for $t = 0, 1, \dots$ do
    Generate synthetic responses by sampling $x \sim \mathcal{D}$ and $y_{1:K} \sim \pi_{\theta_t}(\cdot|x)$
    Annotate the win-rate $\mathcal{P}(y_k \succ y_{k'}|x), \forall k, k' \in [K]$
    Select responses from $y_{1:K}$ to form dataset $D_t = \{(x_i, y_i, \hat{\mathcal{P}}(y_i \succ \pi_{\theta_t}|x_i))\}_{i \in [N]}$
    Optimize $\pi_{\theta_{t+1}}$ according to:
        $\theta_{t+1} \leftarrow \arg\min_{\theta} \mathbb{E}_{(x,y,\hat{\mathcal{P}}(y \succ \pi_{\theta_t}|x)) \sim D_t} \left( \log \left( \frac{\pi_\theta(y|x)}{\pi_{\theta_t}(y|x)} \right) - \eta \left( \hat{\mathcal{P}}(y \succ \pi_{\theta_t}|x) - \frac{1}{2} \right)^2 \right).$
end for
```

where $z_{i-1} \sim \mathcal{N}(0, I)$. For generative modeling, the backward process - a variational Markov chain in the reverse direction - is parameterized with

$$p_\theta\left(x_{i-1} \mid x_i\right) = \mathcal{N}\left(x_{i-1}; \frac{1}{\sqrt{1-\beta_i}}\left(x_i + \beta_i s_\theta\left(i, x_i\right)\right), \beta_i I\right), \tag{22}$$

in which $s_\theta\left(i, x_i\right)$ is learned by maximizing an evidence lower bound (ELBO). In the context of text-to-image generation, trained s_{θ^*} will also be dependent on an input prompt c for conditional generation. For inference, samples can be generated by starting from pure noise and following the estimated reverse process as:

$$x_{i-1} = \frac{1}{\sqrt{1-\beta_i}}\left(x_i + \beta_i s_{\theta^*}\left(i, x_i, c\right)\right) + \sqrt{\beta_i} z_i, \quad i = N, N-1, \cdots, 1. \tag{23}$$

4.5.1 DDPO and DPOK

We review some key elements in DDPO and DPOK (Black et al., 2024; Fan et al., 2024) to formulate the problem of fine-tuning diffusion models as discrete-time MDPs, and then apply RL algorithms. Note that recent works, Tang (2024); Uehara et al. (2024a,b) extend a continuous-time formulation for fine-tuning, but we stick to the discrete time case for simplicity. Consider taking (i, x_i, c) as the state space, and define the action as the next hierarchy x_{i-1} to go to, then Eq. (23) naturally defines a stochastic policy: the stochasticity of the policy comes from $\sqrt{\beta_i} z_i$, thus the policy follows Gaussian with mean determined by $s_{\theta^*}\left(i, x_i, c\right)$ with variance β_i:

$$\pi_\theta(x_{i-1} \mid x_i) \sim \mathcal{N}\left(\frac{1}{\sqrt{1-\beta_i}}\left(x_i + \beta_i s_\theta\left(i, x_i, c\right)\right), \beta_i\right), \quad i = N, N-1, \cdots, 1. \tag{24}$$

Given this formulation, Black et al. (2024) directly maximize the expected reward (without regularization) $\mathcal{J}_{\text{DDPO}} = \mathbb{E}_\theta\left[r(x_0, c)\right]$ by REINFORCE or PPO:

$$\nabla_\theta \mathcal{J}_{\text{DDPO}} = \mathbb{E}\left[\sum_{t=0}^{T} \nabla_\theta \log p_\theta\left(x_{t-1} \mid x_t, c\right) r\left(x_0, c\right)\right]. \tag{25}$$

Compare to DDPO, DPOK (Fan et al., 2024) optimize the same regularized reward objective as in Eq. (8):

$$\mathcal{J}_{\text{DPOK}} = \mathbb{E}_{\theta}\left[r(x_0, c)\right] - \beta \mathbb{E}_{p(z)}\left[\mathrm{KL}\left(p_\theta\left(x_0 \mid z\right) \| p_{\text{pre}}\left(x_0 \mid z\right)\right)\right] \tag{26}$$

They further proposed a clipped gradient algorithm for optimization, motivated by the original PPO clipped objective. In addition, DPOK shows that adding regularization will yield a better generation result compared to the version without regularization.

4.5.2 Reward Feedback Learning (ReFL)

ReFL (Xu et al., 2024b) is a supervised fine-tuning method based on its pre-trained reward model ImageReward $r_{\text{IR}}(c, x)$. The objective for ReFL optimization is a linear combination of negative pre-trained loss (for diffusion models) and reward maximization:

$$\mathcal{J}_{\text{ReFL}}(\theta) = \mathcal{J}_{\text{pre}}(\theta) + \lambda\, \mathbb{E}_{c\sim p_{\boldsymbol{c}}, x_t \sim p_\theta(\cdot|c)}\left(\phi\left(r_{\text{IR}}\left(c, x_t\right)\right)\right), \tag{27}$$

in which λ is a scaling constant, ϕ is taken as a ReLU function and $t \in [0, \tilde{T}]$ is a random number for sampling, a technique that (Xu et al., 2024b) claims can help stabilize the training instead of always letting t be 0.

4.5.3 Direct Reward Fine-Tuning (DRaFT)

DRaFT (Clark et al., 2023) introduces a straightforward method for fine-tuning diffusion models using differentiable reward functions. The goal is to fine-tune the parameters θ of a pre-trained diffusion model such that images generated by the sampling process maximize a differentiable reward function r:

$$J(\theta) = \mathbb{E}_{\boldsymbol{c}\sim p_{\boldsymbol{c}}, \boldsymbol{x}_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})}\left[r(\text{sample}(\theta, \boldsymbol{c}, \boldsymbol{x}_T), \boldsymbol{c})\right], \tag{28}$$

where sample$(\theta, \boldsymbol{c}, \boldsymbol{x}_T)$ denotes the sampling process from time $t = T \rightarrow 0$ with context $\boldsymbol{c}$. First, DRaFT consider solving Eq. 28 by computing $\nabla_\theta r(\text{sample}(\theta, \boldsymbol{c}, \boldsymbol{x}_T), \boldsymbol{c})$ and using gradient ascent. Computing this gradient requires backpropagation through multiple diffusion model calls in the sampling chain, similar to backpropagation through time in a recurrent neural network. To mitigate the memory cost associated with this process, DRaFT employs two strategies: 1) low-rank adaptation (LoRA) (Hu et al., 2021), and 2) gradient checkpointing (Chen et al., 2016).

4.5.4 AlignProp

AlignProp (Prabhudesai et al., 2023) introduces a method that transforms denoising inference within text-to-image diffusion models into a differentiable recurrent policy, effectively linking conditioning input prompts and sampled noise to generate output images. This approach enables fine-tuning of the denoising model's weights through end-to-end backpropagation, guided by differentiable reward functions applied to the generated images. The proposed model casts conditional image denoising as a single step MDP with states $\mathcal{S} = \{(x_T, \mathbf{c}), x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})\}$, actions are the generated image samples, and the whole DDIM denoising chain corresponds to a differentiable policy that maps states to image samples:

$\mathcal{A} = \{x_0 : x_0 \sim \pi_\theta(\cdot|x_T, \mathbf{c}), x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})\ \}$. The reward function is a differentiable function of parameters ϕ that depends only on generated images $R_\phi(x_0), x_0 \in \mathcal{A}$. Given a dataset of prompts input $\mathcal{D}$, our loss function reads:

$$\mathcal{L}_{\text{align}}(\theta; \mathcal{D}) = -\tfrac{1}{|\mathcal{D}|} \sum_{\mathbf{c}^i \in \mathcal{D}} R_\phi(\pi_\theta(x_T, \mathbf{c}^i)). \tag{29}$$

The parameters of the diffusion model using gradient descent on $\mathcal{L}_{\text{align}}$. The policy π is recurrent, and training it is akin to backpropagation through time, a technique commonly used for training recurrent neural networks. The gradient for updating the parameters of the diffusion model with respect to the downstream objective (i.e., the differentiable reward function) is expressed as following:

$$\hat{\nabla}_\theta \mathcal{L}_{\text{align}} = \frac{\partial \mathcal{L}_{\text{align}}}{\partial \theta} + \sum_{t=0}^{K} \frac{\partial \mathcal{L}_{\text{align}}}{\partial x_t} \cdot \frac{\partial x_t}{\partial \theta}, \tag{30}$$

in which K is uniformly drawn from $[0, T]$ for memory efficiency instead of being T, referred as randomized truncation in Prabhudesai et al. (2023).

4.5.5 Diffusion Loss-guided Policy Optimization (DLPO)

DLPO (Chen et al., 2024c) applies online RL to fine-tune TTS diffusion models, where the reward is shaped by the diffusion model's loss. Incorporating the diffusion model loss into the objective function serves as an additional mechanism to enhance performance and maintain the coherence of the model. The method's objective is described as following:

$$\mathbb{E}_{c \sim p(c)} \mathbb{E}_{t \sim \mathcal{U}\{1,T\}} \mathbb{E}_{p_\theta(x_{0:T}|c)} \left[-\alpha r(x_0, c) - \beta \|\tilde{\epsilon}(x_t, t) - \epsilon_\theta(x_t, c, t)\|_2 \right], \tag{31}$$

where α, β are the reward and weights for diffusion model loss, respectively. DLPO uses the following gradient to update the objective:

$$\begin{aligned} &\mathbb{E}_{c \sim p(c)} \mathbb{E}_{t \sim \mathcal{U}\{0,T\}} \mathbb{E}_{p_\theta(x_{1:T}|c)} \\ &\qquad \left[- \left(\alpha r(x_0, c) - \beta \nabla_\theta \|\tilde{\epsilon}(x_t, t) - \epsilon_\theta(x_t, c, t)\|_2 \right) \nabla_\theta \log p_\theta(x_{t-1}|x_t, c) \right]. \end{aligned} \tag{32}$$

The diffusion model objective is incorporated into the reward function as a penalty. This algorithm aligns with the training procedure of TTS diffusion models by integrating the original diffusion model objective $\beta\|\tilde{\epsilon}(x_t, t) - \epsilon_\theta(x_t, c, t)\|_2$ as a penalty in the reward function. This approach effectively prevents model deviation and ensures that the model remains coherent during training.

4.5.6 Human Feedback for Instructional Visual Editing (HIVE)

HIVE (Zhang et al., 2024c) is proposed to improve instruction visual editing models (diffusion models based, e.g., InstructPix2Pix (Brooks et al., 2023)) with human feedback. In instructional supervised training, the stable diffusion model has two conditions $c = [c_I, c_T]$, where c_T is the editing instruction, and c_I is the latent space of the original input image. In the training process, a pre-trained auto-encoder with encoder $\mathcal{E}$ and decoder $\mathcal{D}$ is used to convert between edited image $\tilde{x}$ and its latent representation $z = \mathcal{E}(\tilde{x})$. The diffusion process is composed of an equally weighted sequence of denoising autoencoders $\epsilon_\theta\left(z_t, t, c\right)$, $t = 1, \cdots, T$, which are trained to predict a denoised variant of their input z_t, which is a noisy version of z. The objective of instructional supervised training is:

$$\mathcal{L} = \mathbb{E}_{\mathcal{E}(\tilde{\boldsymbol{x}}), c, \epsilon \sim \mathcal{N}(0, I), t}\left[\left\|\epsilon - \epsilon_\theta\left(z_t, t, c\right)\right\|_2^2\right]. \tag{33}$$

HIVE proposes that optimizing a exponential reward weighted objective for fine-tuning diffusion models:

$$\mathcal{L}_{\text{HIVE}}(\theta) := \mathbb{E}_{\mathcal{E}(\tilde{x}), c, \epsilon \sim \mathcal{N}(0, I), t}\left[\omega(\tilde{x}, c) \cdot \left\|\epsilon - \epsilon_\theta\left(z_t, t, c\right)\right\|_2^2\right], \tag{34}$$

with $\omega(\tilde{x}, c) = \exp\left(r_\phi(\tilde{x}, c)/\beta\right)$ being the exponential reward weight for edited image $\tilde{x}$ and condition c, which is motivated by the closed form of optimal solution for RLHF in Eq. (35).

5 Offline Alignment

In this section, we present a detailed explanation for each offline preference tuning method, including SLiC-HF, DPO and its variants. In Table 5, for simplicity, we include representative DPO variants and their final loss objectives. For each DPO variant, we conclude not only the resulting final objective or algorithm, but also both summarize the motivation or the direction the method contributed to for improvement over DPO.

5.1 Offline Directed Preference Optimization (Offline DPO)

One disadvantage of RLHF is that the RL step often requires substantial computational effort (e.g., to carry out the proximal policy optimization). DPO, recently proposed by Rafailov et al. (2024), suggested a possible way to bypass the reward modeling stage and avoid RL, and has attracted great attention. The key idea of DPO is the observation that given a reward function $r(x, y)$, the problem in Eq. (8) has a closed-form solution:

$$\pi_r(y \mid x) = \frac{1}{Z(x)} \pi_{\text{ref}}(y \mid x) \exp\left(\frac{1}{\beta_{\text{reg}}} r(x, y)\right), \tag{35}$$

where $Z(x) = \sum_y \pi_{\text{ref}}(y \mid x) \exp\left(\frac{1}{\beta_{\text{reg}}} r(x, y)\right)$ is a normalizing constant. Rearranging the terms, and plug in the ground truth reward r^* with the optimal policy $\pi^* = \pi_{r^*}$ yield:

$$r^*(x, y) = \beta_{\text{reg}} \log \frac{\pi^*(y \mid x)}{\pi_{\text{ref}}(y \mid x)} + \beta_{\text{reg}} \log Z(x). \tag{36}$$

Through this change of variables, the latent reward $r^*(x, y)$ can be expressed in terms of the optimal policy $\pi^*(y \mid x)$, the reference policy $\pi_{\text{ref}}\ (y \mid x)$ and a constant $Z^*(x)$. Substituting this r^* expression into Eq. (2) yields:

$$p^*\left(y_1 \succ y_2 \mid x\right) = \sigma\left(\beta_{\text{reg}} \log \frac{\pi^*\left(y_1 \mid x\right)}{\pi_{\text{ref}}\ \left(y_1 \mid x\right)} - \beta_{\text{reg}} \log \frac{\pi^*\left(y_2 \mid x\right)}{\pi_{\text{ref}}\ \left(y_2 \mid x\right)}\right), \tag{37}$$

Method	**Objective**
DPO	$-\log\sigma\left(\beta_{\text{reg}}\log\frac{\pi_\theta(y_w\mid x)}{\pi_{\text{ref}}(y_w\mid x)}-\beta_{\text{reg}}\log\frac{\pi_\theta(y_l\mid x)}{\pi_{\text{ref}}(y_l\mid x)}\right)$
IPO	$\left(\beta_{\text{reg}}\log\frac{\pi_\theta(y_w\mid x)}{\pi_{\text{ref}}(y_w\mid x)}-\beta_{\text{reg}}\log\frac{\pi_\theta(y_l\mid x)}{\pi_{\text{ref}}(y_l\mid x)}-\frac{1}{2}\right)^2$
f-DPO	$-\log\sigma\left(\beta_{\text{reg}}f'\left(\frac{\pi_{\boldsymbol{\theta}}(y_w\mid x)}{\pi_{\text{ref}}(y_w\mid x)}\right)-\beta_{\text{reg}}f'\left(\frac{\pi_{\boldsymbol{\theta}}(y_l\mid x)}{\pi_{\text{ref}}(y_l\mid x)}\right)\right)$
KTO	$-\lambda_w\sigma\left(\beta_{\text{reg}}\log\frac{\pi_\theta(y_w\mid x)}{\pi_{\text{ref}}(y_w\mid x)}-z_{\text{ref}}\right)-\lambda_l\sigma\left(z_{\text{ref}}-\beta_{\text{reg}}\log\frac{\pi_\theta(y_l\mid x)}{\pi_{\text{ref}}(y_l\mid x)}\right),$ where $z_{\text{ref}}=\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\beta_{\text{reg}}\text{KL}\left(\pi_\theta(y\mid x)\Vert\pi_{\text{ref}}(y\mid x)\right)\right]$
ODPO	$-\log\sigma\left(\beta_{\text{reg}}\log\frac{\pi_\theta(y_w\mid x)}{\pi_{\text{ref}}(y_w\mid x)}-\beta_{\text{reg}}\log\frac{\pi_\theta(y_l\mid x)}{\pi_{\text{ref}}(y_l\mid x)}-\Delta_r(x)\right)$
Mallows-DPO	$-\log\sigma\left(\phi(x)\left[\beta_{\text{reg}}\log\frac{\pi_\theta(y_w\mid x)}{\pi_{\text{ref}}(y_w\mid x)}-\beta_{\text{reg}}\log\frac{\pi_\theta(y_l\mid x)}{\pi_{\text{ref}}(y_l\mid x)}\right]\right)$
R-DPO	$-\log\sigma\left(\beta_{\text{reg}}\log\frac{\pi_\theta(y_w\mid x)}{\pi_{\text{ref}}(y_w\mid x)}-\beta_{\text{reg}}\log\frac{\pi_\theta(y_l\mid x)}{\pi_{\text{ref}}(y_l\mid x)}-(\alpha\lvert y_w\rvert-\alpha\lvert y_l\rvert)\right)$
CPO	$-\log p_\theta(y_w\mid x)-\log\sigma\left(\beta_{\text{reg}}\log\pi_\theta(y_w\mid x)-\beta_{\text{reg}}\log\pi_\theta(y_l\mid x)\right)$
ORPO	$-\log p_\theta(y_w\mid x)-\lambda\log\sigma\left(\log\frac{p_\theta(y_w\mid x)}{1-p_\theta(y_w\mid x)}-\log\frac{p_\theta(y_l\mid x)}{1-p_\theta(y_l\mid x)}\right),$ where $p_\theta(y\mid x)=\exp\left(\frac{1}{\lvert y\rvert}\log\pi_\theta(y\mid x)\right)$
SimPO	$-\log\sigma\left(\frac{\beta_{\text{reg}}}{\lvert y_w\rvert}\log\pi_\theta(y_w\mid x)-\frac{\beta_{\text{reg}}}{\lvert y_l\rvert}\log\pi_\theta(y_l\mid x)-\gamma\right)$

Table 9: Various preference optimization DPO objectives. The table is inspired from Meng et al. (2024).

where $Z^*(x)$ cancels out and motivates the DPO objective:

$$\mathcal{L}_{\text{DPO}}\left(\pi_\theta;\pi_{\text{ref}}\right):=\\ -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta_{\text{reg}}\log\frac{\pi_\theta\left(y_w\mid x\right)}{\pi_{\text{ref}}\left(y_w\mid x\right)}-\beta_{\text{reg}}\log\frac{\pi_\theta\left(y_l\mid x\right)}{\pi_{\text{ref}}\left(y_l\mid x\right)}\right)\right],\quad(38)$$

which is a supervised learning problem, requiring much less computation than the RLHF. To understand the loss objective of DPO, we can further examine its gradient as following:

$$\nabla_\theta\mathcal{L}_{\text{DPO}}(\pi_\theta;\pi_{ref})=\\ -\beta_{\text{reg}}\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Bigg[\underbrace{\sigma(\hat{r}_\theta(x,y_l)-\hat{r}_\theta(x,y_w))}_{\text{higher weight when estimate is wrong}}\Bigg[\underbrace{\nabla_\theta\log\pi(y_w\mid x)}_{\text{increase likelihood of }y_w}\\ -\underbrace{\nabla_\theta\log\pi(y_l\mid x)}_{\text{decrease likelihood of }y_l}\Bigg]\Bigg],\quad(39)$$

in which

$$\hat{r}_\theta(x,y)=\beta_{\text{reg}}\log\frac{\pi_\theta(y\mid x)}{\pi_{\text{ref}}\left(y\mid x\right)},\quad(40)$$

is called the implicit reward model for the policy π_θ in DPO.

5.1.1 Identity Preference Optimization (IPO)

For DPO variants, we first visit IPO, proposed in Azar et al. (2024), motivated to bypass the assumption of Bredley-Terry model in the derivation of DPO (which comes from the reward modeling stage of RLHF). Azar et al. (2024) first propose a generic form of regularized optimization objective as:

$$\max_{\pi} \underset{\substack{x\sim\rho \\ y\sim\pi(.|x) \\ y'\sim\mu(.|x)}}{\mathbb{E}} \left[\Psi\left(p^*\left(y \succ y' \mid x\right)\right)\right] - \beta_{\text{reg}} D_{\text{KL}}\left(\pi \| \pi_{\text{ref}}\right) \tag{41}$$

in which the new introduced function Ψ is non-decreasing. They show that Eq. (41) shares the same optimality as DPO when taking $\Psi(q) = \log(q/(1-q))$ (notably this equivalence still needs the Bradley-Terry model assumption). Furthermore, Azar et al. (2024) show that when $\Psi(x) = x$, i.e., when Ψ is the identity mapping, Eq. (41) is equivalent to:

$$\mathcal{L}_{\text{IPO}}\left(\pi_\theta; \pi_{\text{ref}}\right) := \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \left(\beta_{\text{reg}} \log \frac{\pi_\theta\left(y_w \mid x\right)}{\pi_{\text{ref}}\left(y_w \mid x\right)} - \beta_{\text{reg}} \log \frac{\pi_\theta\left(y_l \mid x\right)}{\pi_{\text{ref}}\left(y_l \mid x\right)} - \frac{1}{2}\right)^2, \tag{42}$$

if the offline dataset $\mathcal{D}$ is created by $x \sim \rho$ and $y, y' \sim \mu$. Notice that the derivation of the objective in Eq. (42) does not acquire Bredley-Terry model, thus IPO is *preference model free*. In Azar et al. (2024), it is also demonstrated through a synthetic bandit experiment that DPO can be prone to overfitting, while IPO could avoid this problem. In addition, also shows that online version of IPO (Calandriello et al., 2024) (see details of online DPO in Section 4.2) is indeed equivalent to Nash-MD proposed in Nash Learning from Human Feedback (Munos et al., 2023).

5.1.2 Rejection Sampling Optimization (RSO)

RSO revisits the derivation of DPO and interpret the objective as a maximum likelihood estimator (MLE) of the optimal policy based on Eq. (37) (Liu et al., 2023b). However, such a density estimation problem theoretically requires the datasets to be generated from the optimal policy instead of the SFT model in DPO. Thus, RSO algorithm is proposed to generate the datasets from the approximated optimal policy with an aid of a trained reward model r_{ϕ^*} and statistical rejection sampling, see in Algorithm 3. Notice that $\pi_{r_{\phi^*}}$ is computed by Eq. (35) with the learned reward model r_{ϕ^*}. Liu et al. (2023b) show that this *distribution correction* could help improve the performance of DPO by utilizing the resampled preference dataset.

In addition, RSO also unifies DPO and (normalized) SLiC-HF from the perspective of *loss function*; similar unified perspective also appeared in GPO (Tang et al., 2024b) (see e.g., in Table 1 of it):

$$\mathcal{L}_{\text{GPO}}\left(\pi_\theta; \pi_{\text{ref}}\right) := \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \left[f\left(\beta_{\text{reg}} \log \frac{\pi_\theta\left(y_w \mid x\right)}{\pi_{\text{ref}}\left(y_w \mid x\right)} - \beta_{\text{reg}} \log \frac{\pi_\theta\left(y_l \mid x\right)}{\pi_{\text{ref}}\left(y_l \mid x\right)}\right)\right], \tag{43}$$

for any convex function f, like in Table 11. GPO further provides an analysis of this formulation from an policy improvement and policy regularization tradeoff. Applying Taylor

Algorithm 3 RSO algorithm.

```
Start with empty $\mathcal{Y} \leftarrow \{\}$.
while not enough samples in $\mathcal{Y}$ do
    Generate $y \sim \pi_{\text{sft}}(y \mid x)$ that is not in $\mathcal{Y}$.
    Generate $u \sim U[0,1]$ and let $M = \min\left\{m \mid m \geq \frac{\pi_{r_{\phi^*}}(y|x)}{\pi_{\text{sft}}(y|x)} \text{ for all } y \notin \mathcal{Y}\right\}$.
    if $u < \frac{\pi_{r_{\phi^*}}(y|x)}{M\pi_{\text{sft}}(y|x)}$ then
        Accept $y$ and add it to $\mathcal{Y}$.
    else
        Reject $y$.
    end if
end while
```

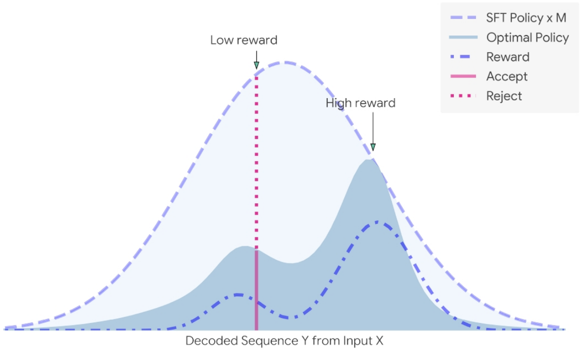

Table 10: RSO illustration in Liu et al. (2023b).

Method	Loss Function	$f(x)$
DPO	log logistic	$-\log\sigma(x)$
IPO	square	$(x-1)^2$
SLiC-HF	hinge loss	$\max(0, 1-x)$

Table 11: Unified perspective through loss function in Liu et al. (2023b); Tang et al. (2024b).

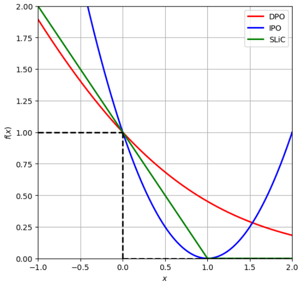

Table 12: Loss function comparison in Tang et al. (2024b).

Expansion of the form above yields:

$$\mathcal{L}_{\text{GPO}}\left(\pi_\theta; \pi_{\text{ref}}\right) = f(0) + \beta_{\text{reg}} \underbrace{f'(0)}_{<0} \underbrace{\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\rho_\theta\right]}_{\text{optimization}} + \frac{1}{2}\beta_{\text{reg}}^2 \underbrace{f''(0)}_{>0} \underbrace{\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\rho_\theta^2\right]}_{\text{regularization}}, \tag{44}$$

in which $\rho_\theta = \log\frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}\,(y_w|x)} - \log\frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}\,(y_l|x)}$ denotes 'implicit reward difference'.

5.1.3 f-DPO

DPO is derived from the RLHF objective which utilized the (reverse) KL divergence to prevent the deviation of new models from old models. f-DPO in Wang et al. (2023a) consider extending this statistical distance to general f-divergence. Concretely, for two probability distribution P and Q with probability density function p and q respectively, f-divergence is defined as:

$$D_f(P||Q) = \mathbb{E}_{q(x)}\left[f\left(\frac{p(x)}{q(x)}\right)\right], \tag{45}$$

and reverse KL divergence is a special case when taking $f(x) = x\log(x)$. Wang et al. (2023a) first show that through a first order condition of optimality / KKT and the similar change of variable technique in DPO, the RLHF objective with a f-divergence

$$\mathcal{L}_{\mathrm{RLHF}-f}(\phi) = \mathbb{E}_{x\sim\mathcal{D},y\sim\pi(\cdot|x)}\left[r_{\phi^*}(x,y) - \beta_{\mathrm{reg}} D_f(\pi(\cdot \mid x) \mid \pi_{\mathrm{ref}}(\cdot \mid x))\right], \tag{46}$$

could yield the f-DPO objective:

$$\begin{aligned}\mathcal{L}_{f-\mathrm{DPO}}\left(\pi_\theta;\pi_{\mathrm{ref}}\right) = \\ \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[-\log\sigma\left(\beta_{\mathrm{reg}} f'\left(\frac{\pi_\theta\left(y_w \mid x\right)}{\pi_{\mathrm{ref}}\left(y_w \mid x\right)}\right) - \beta_{\mathrm{reg}} f'\left(\frac{\pi_\theta\left(y_l \mid x\right)}{\pi_{\mathrm{ref}}\left(y_l \mid x\right)}\right)\right)\right].\end{aligned} \tag{47}$$

Special cases of Eq. (47) are when taking f divergence as α-divergence and JS-divergence, and Wang et al. (2023a) further argue that JS-divergence could possibly yield a better diversity and accuracy tradeoff than reverse KL, through small-scale experiments on e.g., IMDB controllable generation and fine-tuning Pythia 2.8B on Anthropic HH dataset.

5.1.4 Kahneman-Tversky Optimization (KTO)

KTO (Ethayarajh et al., 2024) is motivated to address the need of pairwise preferences datasets in DPO, which can be scarce and expensive. Instead of maximizing the log-likelihood of preferences in DPO and inspired by Kahneman & Tversky's prospect theory, KTO proposes to minimize a human-aware loss function (HALO) that represents the utility of generations and also takes into account the human nature of loss aversion. The resulting KTO objective decouples the pair-preferences into two separate terms that are further linearly combined:

$$\begin{aligned}\mathcal{L}_{\mathrm{KTO}}\left(\pi_\theta;\pi_{\mathrm{ref}}\right) :=& \\ -\,\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}&\left[\lambda_w\sigma(\beta_{\mathrm{reg}}\log\frac{\pi_\theta(y_w|x)}{\pi_{\mathrm{ref}}(y_w|x)} - z_{\mathrm{ref}}) + \lambda_l\sigma(z_{\mathrm{ref}} - \beta_{\mathrm{reg}}\log\frac{\pi_\theta(y_l|x)}{\pi_{\mathrm{ref}}(y_l|x)})\right],\end{aligned} \tag{48}$$

where $z_{\mathrm{ref}} = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\beta_{\mathrm{reg}}\mathrm{KL}\left(\pi_\theta(y|x)||\pi_{\mathrm{ref}}(y|x)\right)\right]$ acts like a subjective value and λ_w, λ_l are additonal hyper-parameters to be tuned. If there is only desired/undesired answer, the KTO objective will thus have only one term, which makes it *pairwise preference data free.*

5.1.5 Offset DPO (ODPO)

DPO objective cannot reflect the *significance* of the preference pairs i.e., the extent y_w is preferred to y_l, and ODPO (Amini et al., 2024) propose to add a margin to capture this significance; they model this margin, or they call offset Δ_r as a monotonically increasing function $f(\cdot)$ of the difference between the scores associated with the responses:

$$\Delta_r(x,y_w,y_l) = \alpha f\left(\mathrm{score}\left(x,y_w\right) - \mathrm{score}\left(x,y_l\right)\right), \tag{49}$$

where α is a hyper-parameter that controls the extent to which an offset should be enforced. The resulting objective becomes:

$$\begin{aligned}\mathcal{L}_{\mathrm{ODPO}}\left(\pi_\theta;\pi_{\mathrm{ref}}\right) :=& \\ -\,\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}&\left[\log\sigma\left(\beta_{\mathrm{reg}}\log\frac{\pi_\theta(y_w|x)}{\pi_{\mathrm{ref}}(y_w|x)} - \beta_{\mathrm{reg}}\log\frac{\pi_\theta(y_l|x)}{\pi_{\mathrm{ref}}(y_l|x)} - \Delta_r(x,y_w,y_l)\right)\right].\end{aligned} \tag{50}$$

5.1.6 Mallows-DPO

Mallows-DPO (Chen et al., 2024b) is motivated by DPO's lack of capability to characterize the diversity of human preferences. Inspired by Mallows Ranking Model (opposed to Bredley-Terry in RLHF and DPO) which has a natural carrier of a dispersion index, Mallows-DPO pays attention to the *dispersion* of the preferences: when human tends to agree about the answer to a certain question, e.g., '$1+1=?$', the preference dispersion will be small; however, the dispersion will be large for answer to a general open question. Chen et al. (2024b) propose a contextual scaled objective derived from MLE under Mallows: compared to DPO that puts equal weights on each prompt and preference pairs, the resulting Mallows-DPO adds a contextual scaling factor $\phi(x)$ that represents this dispersion of the preferences of answers to each prompt x:

$$\begin{aligned}&\mathcal{L}_{\mathrm{Mallows-DPO}}\left(\pi_\theta;\pi_{\mathrm{ref}}\right):=\\&\quad-\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\phi(x)\left[\beta_{\mathrm{reg}}\log\frac{\pi_\theta(y_w|x)}{\pi_{\mathrm{ref}}(y_w|x)}-\beta_{\mathrm{reg}}\log\frac{\pi_\theta(y_l|x)}{\pi_{\mathrm{ref}}(y_l|x)}\right]\right)\right].\end{aligned}\tag{51}$$

To compute this dispersion, Mallows-DPO provided a direct approach by using a normalized predictive entropy of preference pairs $\{y_i^w, y_i^l\}_{i=1,\ldots,N}$ with $N=\max(|y^w|,|y^l|)$:

$$\phi(x)=-\log\left(\frac{\frac{1}{2}\sum_{i=1}^{N-1}\left[H_{\pi_{\mathrm{ref}}}(Y_{i+1}\mid Y_i=y_i^w)+H_{\pi_{\mathrm{ref}}}(Y_{i+1}\mid Y_i=y_i^l)\right]}{\log(n)}\right).\tag{52}$$

To illustrate the effect of this additional term, when dispersion is high: $\phi(x)$ in Eq. (52) will be close to 0, and Mallows-DPO will put less weights on the corresponding preference pairs in the optimization objective to prevent from overfitting; In contrast, when dispersion is low, Mallows-DPO put more weights in the preference optimization objective, for which $\phi(x)$ is large and will lead to stronger effect of alignment.

5.1.7 LR-DPO

LR-DPO (Park et al., 2024), DPO with length regularization is motivated to address the problem of verbosity in the DPO setting. LR-DPO proposed a simple regularization strategy that prevents length exploitation by penalizing the rewards with length of the generation in standard RLHF objective:

$$\mathcal{L}_{\text{LR-RLHF}}(\phi)=\mathbb{E}_{x\sim\mathcal{D},y\sim\pi(\cdot|x)}\left[r_{\phi^*}(x,y)-\alpha|y|-\beta_{\mathrm{reg}}\,\mathrm{KL}(\pi(\cdot\mid x)\mid\pi_{\mathrm{ref}}(\cdot\mid x))\right],\tag{53}$$

in which α is a hyper-parameter that controls the extent of length regularization. Eq. (53) thus similarly yields a supervised learning objective referred as DPO with length regularization:

$$\begin{aligned}&\mathcal{L}_{\mathrm{LR-DPO}}\left(\pi_\theta;\pi_{\mathrm{ref}}\right)\\&\quad=-\mathop{\mathbb{E}}_{(x,y_w,y_l)\sim\mathcal{D}}\log\sigma\left(\beta_{\mathrm{reg}}\log\frac{\pi_\theta(y_w|x)}{\pi_{\mathrm{ref}}(y_w|x)}-\beta_{\mathrm{reg}}\log\frac{\pi_\theta(y_l|x)}{\pi_{\mathrm{ref}}(y_l|x)}-(\alpha|y_w|-\alpha|y_l|)\right).\end{aligned}\tag{54}$$

Park et al. (2024) further show that this can effectively improve model quality by addressing the verbosity issue.

5.1.8 Contrastive Preference Optimization (CPO)

CPO (Xu et al., 2024a) is motivated to improve the memory and speed efficiency of DPO by neglecting the reference policy, further accompanied by a SFT loss term:

$$\mathcal{L}_{\text{CPO}}\left(\pi_\theta\right) := -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log p_\theta(y_w|x) + \log\sigma\left(\beta_{\text{reg}}\log\frac{\pi_\theta(y_w|x)}{\pi_\theta(y_l|x)}\right)\right]. \tag{55}$$

5.1.9 Odds Ratio Preference Optimization (ORPO)

Opposed to maximizing the likelihood ratios of winning and losing answers in the preference pair in DPO, ORPO (Hong et al., 2024) propose that odds ratio can be a more sensible choice.

$$\begin{aligned}\mathcal{L}_{\text{ORPO}}\left(\pi_\theta\right) :=& \\ &-\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log p_\theta(y_w|x) + \lambda\log\sigma\left(\log\frac{p_\theta(y_w|x)}{1-p_\theta(y_w|x)} - \log\frac{p_\theta(y_l|x)}{1-p_\theta(y_l|x)}\right)\right]. \end{aligned}\tag{56}$$

where $p_\theta(y|x) = \exp\left(\frac{1}{|y|}\log\pi_\theta(y|x)\right)$. ORPO is similar to CPO in the sense that it is also reference model free and combined with a SFT loss; in addition, notably that ORPO also adopts a form of length regularization by normalizing the likelihoods with respect to the length, as in the definition of $p_\theta(y|x)$; finally, they compute odds ratio instead of the original likelihood ratio.

5.1.10 SimPO

SimPO (Meng et al., 2024) propose a simple yet effective objective that is claimed to match or even outperform the performance of DPO:

$$\mathcal{L}_{\text{SimPO}}\left(\pi_\theta\right) := -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\frac{\beta_{\text{reg}}}{|y_w|}\log\pi_\theta(y_w|x) - \frac{\beta_{\text{reg}}}{|y_l|}\log\pi_\theta(y_l|x) - \gamma\right)\right], \tag{57}$$

where γ is introduced as a target reward margin to help separating the winning and losing responses. SimPO is similar to CPO in the sense of being reference model free; it also adopted the length normalization for the likelihoods as in ORPO; finally, it additionally introduced a constant margin to be tuned that could help to further improve the performance by encouraging a larger difference between the normalized likelihoods.

5.2 Multi-Modal Models

5.2.1 Diffusion-DPO

Diffusion-DPO (Wallace et al., 2024) is adapting DPO to diffusion models where it uses a fixed dataset where each example contains a prompt and a pairs of images generated from a reference model with human label. Similar to RL for diffusion, the goal is still to align the base diffusion models to human preferences. The derivation is similar to RL framework for diffusion in DDPO and DPOK, and also DPO for Language Models:

$$\mathcal{L}(\theta) = -\mathbb{E}_{\left(x_0^w,x_0^l\right)\sim\mathcal{D}}\log\sigma(\beta_{\text{reg}}\mathbb{E}_{\substack{x_{1:T}^w\sim p_\theta\left(x_{1:T}^w|x_0^w\right)\\ x_{1:T}\sim p_\theta\left(x_{1:T}^l|x_0^l\right)}}\left[\log\frac{p_\theta\left(x_{0:T}^w\right)}{p_{\text{ref}}\left(x_{0:T}^w\right)} - \log\frac{p_\theta\left(x_{0:T}^l\right)}{p_{\text{ref}}\left(x_{0:T}^l\right)}\right]). \tag{58}$$

However, the main concern left is that the likelihood term of the generations $p_\theta\left(x_{0:T}\right)$ is not tractable if only given generation x_0. Wallace et al. (2024) further propose to use the forward process $q\left(x_{1:T} \mid x_0\right)$ of diffusion to match the distribution of backward process $p_\theta\left(x_{1:T} \mid x_0\right)$, and yield the final DPO-Diffusion objective:

$$
\begin{aligned}
L_{\text{DPO-diffusion}}(\theta) = -\mathbb{E}_{\left(x_0^w, x_0^l\right)\sim\mathcal{D}, t\sim\mathcal{U}[0,T], x_t^w\sim q\left(x_t^w \mid x_0^w\right), x_t^l\sim q\left(x_t^l \mid x_0^l\right)} \log\sigma(-\beta_{\text{reg}} T \\
\Big[\mathrm{KL}\left(q\left(x_{t-1}^w \mid x_{0,t}^w\right) \| p_\theta\left(x_{t-1}^w \mid x_t^w\right)\right) - \mathrm{KL}\left(q\left(x_{t-1}^w \mid x_{0,t}^w\right) \| p_{\text{ref}}\left(x_{t-1}^w \mid x_t^w\right)\right) \\
-\mathrm{KL}\left(q(x_{t-1}^l \mid x_{0,t}^l) \| p_\theta(x_{t-1}^l \mid x_t^l)\right) + \mathrm{KL}\left(q(x_{t-1}^l \mid x_{0,t}^l) \| p_{\text{ref}}(x_{t-1}^l \mid x_t^l)\right)\Big]),
\end{aligned} \tag{59}
$$

with each term can be readily computed.

5.2.2 POVID

POVID (Zhou et al., 2024b) proposes a method for performing preference optimization in visual language models (VLLM) with synthetically generated preferences. This is mainly aimed at attenuating the hallucination problems in VLLMs that arises due to lack of alignment between the language and vision modalities. Specifically, the authors use the ground-truth instructions as the preferred response and employ a two-stage approach to generate dis-preferred responses: first, use GPT-4V to inject hallucinatory texts into the preferred responses, and second, add diffusion noise to the image to trigger the inherent hallucination behavior of the VLLM by making the image difficult for the VLLM to understand. Both the strategies are merged together in an reformulation of the DPO loss as:

$$
\begin{aligned}
\mathcal{L}_{\text{POVID}}(\pi_\theta; \pi_{ref}) = -\mathbb{E}_{(x, y_w, y_l)\sim\mathcal{D}} \Bigg[\log\sigma\Bigg(\alpha \log \frac{\pi_\theta(y_w \mid x)}{\pi_{ref}(y_w \mid x)} \\
- \left(\beta_{\text{reg}_1} \log \frac{\pi_\theta(y_l^t \mid x)}{\pi_{ref}(y_l^t \mid x)} + \beta_{\text{reg}_2} \log \frac{\pi_\theta(y_l^n \mid x^n)}{\pi_{ref}(y_l^n \mid x^n)}\right)\Bigg)\Bigg],
\end{aligned} \tag{60}
$$

where $\alpha, \beta_{\text{reg}_1}, \beta_{\text{reg}_2}$ are coefficients for balancing preferred responses (y_w) and dispreferred responses (y_l^t, y_l^n). y_l^t indicates the dispreferred response generated using GPT-4V, and y_l^n denotes the dispreferred response generated using the noisy image x^n.

5.3 Sequence Likelihood Calibration (SLiC-HF)

SLiC-HF (Zhao et al., 2023) uses a sequence level contrastive learning training method to align the model's sequence likelihood over the decoded sequences by measuring their similarity with given reference sequences. The main reason to use a contrastive objective is to put more loss on negative sequence compared to positive sequences such that model puts more probability mass on generating positive sequences. Further, this specific formulation allows the use of human preference for ranking directly by using offline policy preference data $\mathcal{D}$ or by training a separate predictive ranking model on offline data. SLiC-HF obtains a supervised fine-tuned model $\pi_{\theta_{\text{ref}}}(y \mid x)$, which we denote as the reference model for consistency with RLHF pipelines on a reference dataset $(x, y_{\text{target}}) \sim \mathcal{D}$. The preference datasets $\{y_w, y_l\}_m$ is formulated by uniformly drawing answer pairs from $\pi_{\theta_{\text{ref}}}(\cdot \mid x)$ and ranking them by their similarity (from a score computed by a pre-trained model denoted as

$s(y, y_{\text{ref}}; x)$) to the target answer y_{ref}. The step after is to align the SFT model's sequence likelihood using the SLiC loss (Zhao et al., 2022):

$$\mathcal{L}_{\text{SLiC}}(\pi_\theta; \pi_{\text{ref}}) = \sum L^{\text{cal}}\left(\theta, x, y_{\text{target}}, \{y_w, y_l\}_m\right) + \lambda L^{\text{reg}}\left(\theta, \theta_{\text{ref}}; x, y_{\text{target}}\right), \tag{61}$$

in which L^{cal} is the calibration loss from SLiC and L^{reg} is the regularization loss to prevent the aligned model stray away from the SFT model. Taking a special case of L^{cal} and L^{reg} to be a rank calibration loss and cross entropy loss respectively, Eq. (61) becomes:

$$\mathcal{L}_{\text{SLiC}}(\pi_\theta; \pi_{\text{ref}}) = \underbrace{\max\left(0, \delta - \log \pi_\theta(y_w|x) + \log \pi_\theta(y_l|x)\right)}_{\text{rank calibration loss}} \underbrace{-\lambda \log \pi_\theta(y_{\text{ref}}|x)}_{\textit{regularization}}, \tag{62}$$

where, in the first term of calibration loss, we are maximizing the likelihood corresponding to the positive sequence y_w and minimizing negative sequence y_l and the margin δ is a hyperparameter represents which can be a constant or prompt dependent score/rank difference; the second term is just standard SFT loss. As a remark, one can use a secondary reward model, opposed to the similarity function in SLiC, trained on human preference data to classify positive or negative pairs (y_w, y_l).

6 Combined Policies and Sampling-Agnostic Alignment

In this section, we explore a different approach to human preference tuning by integrating off-policy and on-policy methods. This combined approach has proven to be highly effective and complementary in enhancing the alignment quality of the model. Additionally, we introduce sampling-agnostic alignment methods that can be applied to both off-policy and on-policy approaches.

6.1 ExPO

ExPO (Zheng et al., 2024a) provides a simple and training-free method for enhancing the alignment of large language models (LLMs) with human preferences. The core insight behind ExPO is that a model trained with DPO/RLHF can be viewed as an interpolation between two models with differing strengths. By leveraging this concept, one can potentially extrapolate a stronger model if the other two models are available. Specifically, if we denote the model π_{ExPO} as the interpolation of two other models, π_a and π_b, which may be trained using different alignment methods and datasets, ExPO assumes that combining these models will yield improved alignment. The stronger, better-aligned model π_{ExPO} is then obtained by extrapolating from the weights of these two relatively weaker models, as formulated below:

$$\pi_{\text{ExPO}} = (1+\alpha)\pi_a - \alpha\pi_b = \pi_a + \alpha\left(\pi_a - \pi_b\right) = \pi_a + \alpha\Delta\pi. \tag{63}$$

This method is shown to work when π_a and π_b are a stronger model from a combination of SFT model and a model further preference trained on top of it respectively. However in naive cases of choosing arbitrary π_a and π_b, it has shown to cause model collapse or degradation. Nevertheless broader applicability of this approach requires further research.

6.2 Policy-on Policy-off Policy Optimization (P3O)

P3O (Fakoor et al., 2020) is a simple and effective algorithm that uses the effective sample size to automatically manage the combination of on-policy and off-policy optimization. It performs gradient ascent using the gradient. Fakoor et al. (2020) describe how P3O integrates the following on-policy update with the off-policy update:

$$\nabla_\theta^{\text{on}} J\left(\pi_\theta\right) = \underset{s \sim d^\pi \theta, a \sim \pi_\theta}{\mathbb{E}}\left[g\left(\pi_\theta\right)\right], \tag{64}$$

$$\nabla_\theta^{\text{off}} J\left(\pi_\theta\right) = \underset{s \sim d_{\text{reg}}^\beta, a \sim \beta_{\text{reg}}}{\mathbb{E}}\left[\bar{\rho}_c g\left(\pi_\theta\right)\right], \tag{65}$$

where π_θ denotes a policy that is parameterized by parameters $\theta \in \mathbb{R}^n$, and q^{π_θ} and v^{π_θ} denote a parameterization of the state-action and state-only value functions, respectively. It is also denoted the baselined policy gradient integrand in short by following:

$$g\left(\pi_\theta\right) = \hat{A}^{\pi_\theta}(s, a) \nabla_\theta \log \pi_\theta(a \mid s), \tag{66}$$

$$\hat{A}^{\pi_\theta}(s, a) = \hat{q}^{\pi_\theta}(s, a) - \hat{v}^{\pi_\theta}(s). \tag{67}$$

It forms a unified policy optimization as following:

$$\underset{s \sim d^{\pi_\theta}, a \sim \pi_\theta}{\mathbb{E}}\left[g\left(\pi_\theta\right)\right] + \underset{s \sim d_{\text{reg}}^\beta, a \sim \beta_{\text{reg}}}{\mathbb{E}}\left[\bar{\rho}_c g\left(\pi_\theta\right)\right] - \lambda \nabla_\theta \underset{s \sim d_{\text{reg}}^\beta, a \sim \beta_{\text{reg}}}{\mathbb{E}} \mathrm{KL}\left(\beta_{\text{reg}}(\cdot \mid s) \| \pi_\theta(\cdot \mid s)\right). \tag{68}$$

The first term above is the standard on-policy gradient. The second term is the off-policy policy gradient with truncation of the IS ratio using a constant c while the third term allows explicit control of the deviation of the target policy π_θ from β_{reg}. Further, the KL-divergence term can be rewritten as $\underset{s \sim d^\pi \theta, a \sim \pi_\theta}{\mathbb{E}}[\log \rho]$ and therefore minimizes the importance ratio ρ over the entire replay buffer β_{reg}. There are two hyper-parameters in the P3O gradient: the IS ratio threshold c and the KL regularization co-efficient λ.

6.3 Reinforced Token Optimization (RTO)

RTO (Zhong et al., 2024) derives a token-wise reward function from preference data and conducts policy optimization using this learned reward signal. Broadly, RTO involves two primary steps: (1) learning a token-wise reward from preference data, and (2) optimizing this reward through RL training methods like PPO. Theoretically, the algorithm is framed as an MDP problem.

Theoretical Version It focuses on the offline setting assuming it has an offline dataset $\mathcal{D} = \{(\tau^w, \tau^l)\}$ that contains several trajectory pairs, where $\tau^w = \{(s_h^w, a_h^w)\}_{h=1}^H$ is preferred over $\tau^l = \{(s_h^l, a_h^l)\}_{h=1}^H$. Each pair of trajectories shares the same initial state (i.e., $s_1^w = s_1^l$), but differs in the subsequent tokens. They assume that the reward function is linear, and the results are extended to general function approximation. RTO computes the θ_{mle} based on $\mathcal{D}$ by maximizing the log likelihood and calculates the pessimistic reward $\hat{r}$ via token-wise reward learning. The output of the algorithm is policy $\hat{\pi}$.

Practical Version The key challenge in implementing RTO lies in learning the token-wise reward to be optimized from the offline data. In the most popular frameworks outlined in InstructGPT (Ouyang et al., 2022), Claude (Bai et al., 2022a), and LLaMA2 (Touvron et al., 2023a) projects replace the last layer of the LLM with a linear layer for a scalar output and maximize the log-likelihood. However, this approach gives only a sentence-level reward. To bridge the gap in the literature, they present the practical version of RTO, which features a novel calculation of token-wise reward. Their key observation is that, given a trajectory $\tau = \{(s_h, a_h)\}_{h=1}^H$, RTO is defined the following:

$$\begin{aligned}\sum_{h=1}^{H} \beta_{\mathrm{reg}} \log \frac{\pi^*_{\beta_{\mathrm{reg}}}(a_h|s_h)}{\pi_{\mathrm{ref}}(a_h|s_h)} &= \sum_{h=1}^{H} \big(Q^*_{\beta_{\mathrm{reg}}}(s_h, a_h) - V^*_{\beta_{\mathrm{reg}}}(s_h) - \log \pi_{\mathrm{ref}}(a_h|s_h)\big) \\ &= \sum_{h=1}^{H} r(s_h, a_h) - V^*_{\beta_{\mathrm{reg}}}(s_1) \qquad (69) \\ &+ \underbrace{\sum_{h=1}^{H-1} \big(\mathbb{E}_{s'\sim\mathcal{P}(\cdot|s_h,a_h)}[V^*_{\beta_{\mathrm{reg}}}(s')] - V^*_{\beta_{\mathrm{reg}}}(s_{h+1})\big)}_{(\star)}, \qquad (70)\end{aligned}$$

where the first equality uses the closed-form of optimal policy $\pi^*_{\beta_{\mathrm{reg}}}(a|s) = \exp\{(Q^*_{\beta_{\mathrm{reg}}}(s,a) - V^*_{\beta_{\mathrm{reg}}}(s))/\beta_{\mathrm{reg}}\}$, and the second equality follows from the fact that $Q^{\pi}_{\beta_{\mathrm{reg}}}(s,a) = r_{\beta_{\mathrm{reg}}}(s,a) + \mathbb{E}_{s'\sim\mathcal{P}(\cdot|s,a)}[V_{\beta_{\mathrm{reg}}\pi}(s')]$ with $r_{\beta_{\mathrm{reg}}}(s,a) = r(s,a) + \beta_{\mathrm{reg}} \log \pi_{\mathrm{ref}}(a|s)$. RTO focuses on the typical LLM generation scenario where the transition kernel is deterministic. Then, $(\star) = 0$, yielding that

$$\sum_{h=1}^{H} r(s_h, a_h) = \sum_{h=1}^{H} \beta_{\mathrm{reg}} \log \frac{\pi^*_{\beta_{\mathrm{reg}}}(a_h|s_h)}{\pi_{\mathrm{ref}}(a_h|s_h)} + V^*_{\beta_{\mathrm{reg}}}(s_1). \qquad (71)$$

Building upon this result and combining it with the definition of the BT model, for any trajectory pair $\{\tau^j = \{(s_h^j, a_h^j)\}_{h=1}^H\}_{j=1}^2$ satisfying $s_1^1 = s_1^2$, we have:

$$\begin{aligned}\mathbb{P}(\tau^1 \succ \tau^2) &= \sigma\left(\sum_{h=1}^{H} r(s_h^1, a_h^1) - \sum_{h=1}^{H} r(s_h^2, a_h^2)\right) \\ &= \sigma\left(\sum_{h=1}^{H} \beta_{\mathrm{reg}} \log \frac{\pi^*_{\beta_{\mathrm{reg}}}(a_h^1|s_h^1)}{\pi_{\mathrm{ref}}(a_h^1|s_h^1)} - \sum_{h=1}^{H} \beta_{\mathrm{reg}} \log \frac{\pi^*_{\beta_{\mathrm{reg}}}(a_h^2|s_h^2)}{\pi_{\mathrm{ref}}(a_h^2|s_h^2)}\right).\end{aligned} \qquad (72)$$

Similar to the bandit setting where the learning objective is equivalent to a BT model with sentence-wise reward $r^*(x,y) = \beta_{\mathrm{reg}} \log \frac{\pi^*_{\beta_{\mathrm{reg}}}(y|x)}{\pi_{\mathrm{ref}}(y|x)}$ (Rafailov et al., 2024), it shows that the learning objective in token-wise MDP equivalents to a BT model with a token-wise reward function

$$r^*(s_h = (x, y_{1:h-1}), a_h = y_h) = \beta_{\mathrm{reg}} \log \frac{\pi^*_{\beta_{\mathrm{reg}}}(a_h|s_h)}{\pi_{\mathrm{ref}}(a_h|s_h)} = \beta_{\mathrm{reg}} \log \frac{\pi^*_{\beta_{\mathrm{reg}}}(y_h|x, y_{1:h-1})}{\pi_{\mathrm{ref}}(y_h|x, y_{1:h-1})}, \qquad (73)$$

where x is the prompt, $y_{1:h-1}$ is the tokens generated so far, and y_h is the token chosen at the current step. RTO assigns the defined token-wise reward function to each step. Formally, for any h, it is defined as following:

$$\begin{aligned} &{\beta_{\text{reg}}}^1 \log \frac{\pi^*_{\beta_{\text{reg}}}(y_h|x, y_{1:h-1})}{\pi_{\text{ref}}(y_h|x, y_{1:h-1})} - \beta^2_{\text{reg}} \log \frac{\pi(y_h|x, y_{1:h-1})}{\pi_{\text{ref}}(y_h|x, y_{1:h-1})} \\ &\qquad \approx \beta^1_{\text{reg}} \log \frac{\pi_{\text{dpo}}(y_h|x, y_{1:h-1})}{\pi_{\text{ref}}(y_h|x, y_{1:h-1})} - \beta^2_{\text{reg}} \log \frac{\pi(y_h|x, y_{1:h-1})}{\pi_{\text{ref}}(y_h|x, y_{1:h-1})} := r_{\text{rto}}((x, y_{1:h-1}), y_h), \end{aligned} \tag{74}$$

as the token-wise reward used by RTO, where β^1_{reg} and β^2_{reg} are tuning parameters, and π is the current policy to be updated. In the last step, RTO uses π_{dpo}, the policy learned by DPO, as a proxy for the unknown $\pi^*_{\beta_{\text{reg}}}$. Finally, RTO employs PPO to optimize the token-wise reward r_{rto}.

7 Evaluation

Evaluation metrics and pipelines are essential for measuring the core capabilities of LLMs in performing tasks and assessing their alignment with human preferences in open-ended scenarios. Numerous evaluation metrics have been proposed in the literature. In this section, we will describe these metrics and evaluation methods across different modalities.

7.1 LLM As A Judge

Human evaluation is both costly and time-consuming. Developing an automatic evaluation method that closely aligns with human assessments can significantly reduce evaluation time and accelerate research progress. In this context, we outline the benchmarks employed for automatic evaluation using LLMs.

7.1.1 AlpacaEval

AlpacaEval (Dubois et al., 2024) win rate (against GPT4) is an LLM-based automatic evaluation that has high-level agreement to human. To further improve the fairness of the evaluation and address the verbosity of issue of GPT4 as a judge, Dubois et al. (2024) introduce a length-controlled version of AlpacaEval that aims to conduct measurement with outputs with similar lengths. The metric is used in AlpacaEval calculates win-rates for models across a variety of NLP tasks to measure of model capabilities compared to a baseline by using an LLM judge.

AlpacaEval 2.0 The judge uses GPT4-Turbo to replace GPT-3 based model "text-davinci-003" in the 1.0 version, which makes it more challenging and have a metric that better reflects the current SOTA model.

7.1.2 ChatbotArena

ChatbotArena (Chiang et al., 2024) is a benchmarking platform for Large Language Models (LLMs) that conducts anonymous, randomized battles in a crowdsourced environment. On this platform, users can pose questions and receive responses from two anonymous LLMs. After reviewing the answers, users vote for the response they prefer, with the identities of the models revealed only after voting. This crowdsourced approach effectively gathers a diverse array of user prompts, accurately reflecting real-world LLM applications. Utilizing this data, they apply a range of advanced statistical techniques, from the Bradley-Terry model (Bradley and Terry, 1952) to the E-values framework (Vovk and Wang, 2021), to estimate model rankings as reliably and efficiently as possible.

7.1.3 MT-Bench

MT-bench (Zheng et al., 2024b) is a series of open-ended questions designed to evaluate a chatbot's multi-turn conversational and instruction-following abilities. It is used in the platform that assesses these capabilities in a crowdsourced battle format. This platform is particularly useful for evaluating the quality of LLM-generated responses, utilizing judges like GPT-4. Consequently, employing LLM as a judge provides a scalable and explainable method to approximate human preferences, which would otherwise be very costly to obtain.

7.1.4 HELM

HELM (Liang et al., 2022) is a large-scale reproducible and transparent framework for evaluating LLM models to enhance the transparency of language models. The framework has seven metrics, such as accuracy, calibration, robustness, fairness, bias, toxicity, and efficiency.

7.2 Vision Language Model Evaluation

7.2.1 VHELM

VHELM[1] is an extension of the HELM framework (Liang et al., 2022) with the adaptation methods to assess the performance of VLMs by scoring the winning rates against the GPT-4V model.

7.2.2 MMStar

MMStar (Chen et al., 2024e) is a multi-modal benchmark consisting of 1,500 samples meticulously curated by human experts. It evaluates six core capabilities and 18 specific criteria to assess the multi-modal capacities of LVLMs. The samples are selected from existing benchmarks using an automated process, followed by human review to ensure each sample demonstrates visual dependency, minimal data leakage, and requires advanced multi-modal skills.

7.3 Speech Language Model Evaluation

7.3.1 SpeechLMScore

SpeechLMScore (Maiti et al., 2023) calculates the average log-probability of a speech signal by converting it into discrete tokens and assessing the average probability of generating the

[1] https://crfm.stanford.edu/helm/vhelm/latest/

token sequence. Formally, SpeechLMScore($\mathbf{x}|\theta$) is defined as:

$$\text{SpeechLMScore}(\mathbf{d}|\theta) = \frac{1}{T}\sum_{i=1}^{T} \log p(d_i|d_{<i}, \theta), \tag{75}$$

where θ is an LM used to generate the score. Specifically, to compute SpeechLMScore, the process involves: i) encoding the speech into discrete tokens $\mathbf{d} = d_1 \cdots d_T$, and ii) iteratively calculating the log probability of each token d_i given all preceding tokens $d_1 \cdots d_{i-1}$ using θ, i.e., $\log p(d_i|d_{<i}, \theta)$. SpeechLMScore thus measures the average log-probability of a sequence of speech tokens. This metric is closely related to the perplexity of a speech sample, essentially indicating how perplexed a speech language model is when presented with a set of discrete tokens from speech $\mathbf{x}$.

7.3.2 SpeechBERTScore

SpeechBERTScore (Saeki et al., 2024) evaluates the BERTScore for self-supervised dense speech features derived from both generated and reference speech, even when these sequences differ in length. This method utilizes BERTScore as a metric to assess the quality of speech generation. By computing the BERTScore for SSL feature sequences from both the generated and reference speech, SpeechBERTScore effectively captures their semantic alignment.

7.4 Reward Model Evaluation

One way to assess the quality of our model is by evaluating the performance of the reward model using a benchmark. Zhu et al. (2024); Jiang et al. (2023b) propose using validation sets from previous RLHF training processes, such as Anthropic's Helpful and Harmless data (Bai et al., 2022a) or OpenAI's Learning to Summarize (Stiennon et al., 2020). Additionally, newly released preference data, aimed at expanding the diversity of preference training datasets, such as UltraFeedback (Cui et al., 2023), UltraInteract (Yuan et al., 2024a), and Nectar (Zhu et al., 2024), lack test sets, necessitating a new style of evaluation for reward models. RewardBench is a benchmark dataset and codebase designed for this purpose (Lambert et al., 2024). The dataset comprises a collection of prompt-chosen-rejected triplets that span various domains, including chat, reasoning, and safety. This allows for a comprehensive evaluation of how reward models perform on challenging, structured, and out-of-distribution queries.

8 Discussion and Research Directions

In this section, we describe topics related to human preferences that are either underexplored or still in their early stages. We also discuss potential future research areas that could be highly beneficial for advancing the field.

8.1 Discussion

8.1.1 Effectiveness of Optimization Components

In the literature on preference tuning, the comparative performance of different methods remains unclear, particularly when comparisons are not conducted under fair conditions. This is largely because RL is highly sensitive to changes in hyper-parameters, and running multiple hyper-parameter configurations is very costly. For instance, when a new method is proposed, the baseline may not be fully optimized, resulting in weaker baselines. Another issue in automatic evaluation using LLMs as judges is the bias introduced by the pre-training data. A model might prefer predictions generated by a similar type of model. For example, a GPT-4 model may favor outputs from its own model family over those from other models, such as Llama. Additionally, judge models may have a preference for longer sequences or text in certain positions (Zheng et al., 2024b). Therefore, finding a less biased model is crucial during evaluation. Consequently, the effectiveness of each method, along with their optimized components and the models used in automatic evaluation, needs further investigation and careful consideration.

8.1.2 Offline vs. Online Algorithms

Through theoretical and experimental analysis, Xu et al. (2024c) explore the limitations of DPO and find that DPO is sensitive to distribution shifts between the base model outputs and preference data. They suggest that iterative DPO, which involves continuous updating, is more effective than training on static data. However, they also find that DPO fails to improve performance on challenging tasks such as code generation. From a different perspective, Tang et al. (2024a) clarify the confusion surrounding the limitations of offline algorithms' performance, often attributed to the bounded performance of offline algorithms. The paper discusses that the dichotomy between online and offline algorithms is frequently inaccurate in practice. An offline algorithm that continuously updates its data stream effectively functions as an online algorithm. Consequently, the shortcomings identified in offline learning can be mitigated by adopting a more careful approach to the data generation process.

8.2 Research Directions

Here, we explore potential research directions that offer significant opportunities for further investigation and development. These avenues hold promise for both academic researchers and industry practitioners, providing ground for innovative studies and practical applications. We summarize key ideas and topics that could drive future advancements in the field, highlighting areas where there is ample room for exploration and growth.

8.2.1 Multilinguality

While significant resources have been allocated to enhance the safety of LLMs for deployment, the safety of multilingual LLMs remains underexplored. Ahmadian et al. (2024b) is one of the pioneering works pushing the boundaries of aligning language models by optimizing for both general and safety performance simultaneously in a multilingual setting using Distributional DPO. Similarly, Li et al. (2024) propose exploring DPO training to reduce toxicity in multilingual open-ended generations. Another line of research focuses on using multilingual alignment based on human preferences to improve reasoning abilities, aiming to align reasoning processes in other languages with those in the dominant language (She et al., 2024). There is still ample room for exploration in the multilingual space, particularly in examining the cultural aspects of multilingualism (Adilazuarda et al., 2024; AlKhamissi et al., 2024) and improving the alignment of LLM for generation (Winata et al., 2021b). It is crucial to cover more diverse languages, including regional languages, different dialects (Aji et al., 2022), and code-switching (Winata et al., 2021a), which are common phenomena in bilingual and multilingual communities (Winata et al., 2024). Additionally, the exploration of multilingual topics in vision-language and speech tasks remains open for further investigation.

8.2.2 Multi-modality

While alignment in LLMs has been extensively studied, alignment for multi-modal models has not yet been investigated to the same extent. Sun et al. (2023) and Zhou et al. (2024b) align LLaVA (Liu et al., 2024a) using PPO and DPO, respectively. Similarly, Li et al. (2023c) and Yu et al. (2023) employ DPO and its variations to align the Qwen-VL (Bai et al., 2023) and Muffin (Yu et al., 2023) models. Notably, in addition to different alignment strategies and base models, all these works introduce novel preference datasets for alignment, varying in size, collection methods, and generation schemes. Consequently, while each of these studies offers valuable insights into alignment for multi-modal LLMs, it can sometimes be challenging to attribute reported improvements to specific proposed choices. Furthermore, Amirloo et al. (2024) examine each component of multi-modal alignment independently, involving sampling from the model during policy optimization.

8.2.3 Speech Applications

The application of preference tuning in speech technology is in its early stages, offering many opportunities for future exploration. As research advances, preference tuning is expected to enhance various speech-related technologies, including TTS and speech recognition systems, by incorporating human preferences to improve performance and user satisfaction. In TTS, it can help select the most natural and pleasing synthetic voices (Zhang et al., 2024a), while in speech recognition, it can ensure more accurate and contextually appropriate transcriptions. Additionally, preference tuning can benefit voice assistants, automated customer service systems, and language learning tools by creating more intuitive and effective interfaces. Ongoing research and experimentation will be essential to fully realize the potential of preference tuning in speech technology, aiming to develop systems that are both technically proficient and closely aligned with human communication and preferences.

8.2.4 Unlearning

Yao et al. (2023b); Zhang et al. (2024b) propose an alignment technique for unlearning by utilizing negative examples, which are easier and cheaper to collect than the positive examples needed for preference tuning. This method is considered computationally efficient, with costs comparable to light supervised finetuning. They demonstrate that unlearning is particularly appealing when resources are limited and the priority is to stop generating undesirable outputs. Despite using only negative samples, unlearning can achieve better alignment performance than RLHF. The unlearning method can be very useful in removing harmful responses, erasing copyright-protected content, and reducing hallucinations. This approach is promising and has potential for further exploration in future work.

8.2.5 Benchmarking Preference Tuning Methods

Developing a comprehensive benchmark for various preference tuning methods is essential for gaining a clearer understanding of their individual effectiveness. Currently, the effectiveness of each method is somewhat unclear, making it difficult to fully appreciate their value. By creating a benchmark, we can systematically assess and compare these methods, thereby clarifying their strengths and weaknesses. This effort to elucidate the usefulness of each approach is vital for advancing our knowledge and improving the application of preference tuning techniques. Such a benchmark would not only enable more informed decisions when selecting the most suitable method for specific tasks but also stimulate innovation by identifying areas that require further refinement and development. Ultimately, this initiative aims to enhance the overall effectiveness and reliability of preference tuning methods across various applications.

8.2.6 Mechanistic Understanding of Preference Tuning Methods

Despite the popularity of preference tuning methods for LLM alignment, explanations for their underlying mechanisms in which models become “aligned” still lack, thus making it difficult to explain phenomena like jailbreaks (Chao et al., 2023). Taking toxicity reduction as the task and applying DPO on GPT2-medium, Lee et al. (2024) suggest that capabilities may be rather bypassed instead of removed. Thus, current preference tuning methods may still be vulnerable to reverse-engineering and the models tuned are easy to be unaligned again. More interpretation of preference tuning methods could possibly address these concerns by ensuring that models not only meet alignment objectives more reliably but also provide clearer insights into how these objectives are achieved; it could also possibly help lead to better preference tuning methods that can mitigate issues such as jailbreaking and other forms of misalignment, where models exhibit undesirable behaviors despite appearing aligned during training.

Acknowledgement: Wenpin Tang and Hanyang Zhao are supported by NSF grant DMS-2206038, a start-up grant at Columbia University, and the Columbia Innovation Hub grant. The works of Hanyang Zhao and David D. Yao are part of a Columbia-CityU/HK collaborative project that is supported by InnotHK Initiative, The Government of the HKSAR and the AIFT Lab.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, and Harkirat Behl. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, and Shyamal Anadkat. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Muhammad Farid Adilazuarda, Sagnik Mukherjee, Pradhyumna Lavania, Siddhant Singh, Ashutosh Dwivedi, Alham Fikri Aji, Jacki O'Neill, Ashutosh Modi, and Monojit Choudhury. Towards measuring and modeling" culture" in llms: A survey. *arXiv preprint arXiv:2403.15412*, 2024.

Bo Adler, Niket Agarwal, Ashwath Aithal, Dong H Anh, Pallab Bhattacharya, Annika Brundyn, Jared Casper, Bryan Catanzaro, Sharon Clay, and Jonathan Cohen. Nemotron-4 340b technical report. *arXiv preprint arXiv:2406.11704*, 2024.

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms. *arXiv preprint arXiv:2402.14740*, 2024a.

Arash Ahmadian, Beyza Ermis, Seraphina Goldfarb-Tarrant, Julia Kreutzer, Marzieh Fadaee, and Sara Hooker. The multilingual alignment prism: Aligning global and local preferences to reduce harm. *arXiv preprint arXiv:2406.18682*, 2024b.

Alham Fikri Aji, Genta Indra Winata, Fajri Koto, Samuel Cahyawijaya, Ade Romadhony, Rahmad Mahendra, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, and Timothy Baldwin. One country, 700+ languages: Nlp challenges for underrepresented languages and dialects in indonesia. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7226–7249, 2022.

Badr AlKhamissi, Muhammad ElNokrashy, Mai AlKhamissi, and Mona Diab. Investigating cultural alignment of large language models. *arXiv preprint arXiv:2402.13231*, 2024.

Loubna Ben Allal, Raymond Li, Denis Kocetkov, Chenghao Mou, Christopher Akiki, Carlos Munoz Ferrandis, Niklas Muennighoff, Mayank Mishra, Alex Gu, and Manan Dey. Santacoder: don't reach for the stars! *arXiv preprint arXiv:2301.03988*, 2023.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Hesslow, Julien Launay, and Quentin Malartic. The falcon series of open language models. *arXiv preprint arXiv:2311.16867*, 2023.

Afra Amini, Tim Vieira, and Ryan Cotterell. Direct preference optimization with an offset. *arXiv preprint arXiv:2402.10571*, 2024.

Elmira Amirloo, Jean-Philippe Fauconnier, Christoph Roesmann, Christian Kerl, Rinu Boney, Yusu Qian, Zirui Wang, Afshin Dehghan, Yinfei Yang, and Zhe Gan. Understanding alignment in multimodal llms: A comprehensive study. *arXiv preprint arXiv:2407.02477*, 2024.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, and Zhifeng Chen. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

AI Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1, 2024.

Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Kelly Marchisio, and Sebastian Ruder. Aya 23: Open weight releases to further multilingual progress. *arXiv preprint arXiv:2405.15032*, 2024.

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

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, and Fei Huang. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

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

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, and Cameron McKinnon. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022b.

Marco Bellagente, Jonathan Tow, Dakota Mahan, Duy Phung, Maksym Zhuravinskyi, Reshinth Adithyan, James Baicoianu, Ben Brooks, Nathan Cooper, and Ashish Datta. Stable lm 2 1.6 b technical report. *arXiv preprint arXiv:2402.17834*, 2024.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, and Edward Raff. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR, 2023.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023.

Hannah Brown, Katherine Lee, Fatemehsadat Mireshghallah, Reza Shokri, and Florian Tramèr. What does it mean for a language model to preserve privacy? In *Proceedings of the 2022 ACM conference on fairness, accountability, and transparency*, pages 2280–2292, 2022.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, and Amanda Askell. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Samuel Cahyawijaya, Holy Lovenia, Alham Fikri Aji, Genta Winata, Bryan Wilie, Fajri Koto, Rahmad Mahendra, Christian Wibisono, Ade Romadhony, and Karissa Vincentio. Nusacrowd: Open source initiative for indonesian nlp resources. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13745–13818, 2023.

Samuel Cahyawijaya, Holy Lovenia, Fajri Koto, Rifki Afina Putri, Emmanuel Dave, Jhonson Lee, Nuur Shadieq, Wawan Cenggoro, Salsabil Maulana Akbar, and Muhammad Ihza Mahendra. Cendol: Open instruction-tuned generative large language models for indonesian languages. *arXiv preprint arXiv:2404.06138*, 2024.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, and Pei Chu. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*, 2024.

Daniele Calandriello, Daniel Guo, Remi Munos, Mark Rowland, Yunhao Tang, Bernardo Avila Pires, Pierre Harvey Richemond, Charline Le Lan, Michal Valko, and Tianqi Liu. Human alignment of large language models through online preference optimisation. *arXiv preprint arXiv:2403.08635*, 2024.

Hanqun Cao, Cheng Tan, Zhangyang Gao, Yilun Xu, Guangyong Chen, Pheng-Ann Heng, and Stan Z Li. A survey on generative diffusion models. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

Souradip Chakraborty, Jiahao Qiu, Hui Yuan, Alec Koppel, Furong Huang, Dinesh Manocha, Amrit Singh Bedi, and Mengdi Wang. Maxmin-rlhf: Towards equitable alignment of large language models with diverse human preferences. *arXiv preprint arXiv:2402.08925*, 2024.

Stanley H Chan. Tutorial on diffusion models for imaging and vision. *arXiv preprint arXiv:2403.18103*, 2024.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries. *arXiv preprint arXiv:2310.08419*, 2023.

Canyu Chen and Kai Shu. Can llm-generated misinformation be detected? *arXiv preprint arXiv:2309.13788*, 2023.

Chen Chen, Yuchen Hu, Wen Wu, Helin Wang, Eng Siong Chng, and Chao Zhang. Enhancing zero-shot text-to-speech synthesis with human feedback. *arXiv preprint arXiv:2406.00654*, 2024a.

Haoxian Chen, Hanyang Zhao, Henry Lam, David Yao, and Wenpin Tang. Mallows-dpo: Fine-tune your llm with preference dispersions. *arXiv preprint arXiv:2405.14953*, 2024b.

Jingyi Chen, Ju-Seung Byun, Micha Elsner, and Andrew Perrault. Reinforcement learning for fine-tuning text-to-speech diffusion models. *arXiv preprint arXiv:2405.14632*, 2024c.

Lichang Chen, Jiuhai Chen, Chenxi Liu, John Kirchenbauer, Davit Soselia, Chen Zhu, Tom Goldstein, Tianyi Zhou, and Heng Huang. Optune: Efficient online preference tuning. *arXiv preprint arXiv:2406.07657*, 2024d.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, and Dahua Lin. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*, 2024e.

Minshuo Chen, Song Mei, Jianqing Fan, and Mengdi Wang. An overview of diffusion models: Applications, guided generation, statistical rates and optimization. *arXiv preprint arXiv:2404.07771*, 2024f.

Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. Training deep nets with sublinear memory cost. *arXiv preprint arXiv:1604.06174*, 2016.

Zhipeng Chen, Kun Zhou, Wayne Xin Zhao, Jingyuan Wang, and Ji-Rong Wen. Low-redundant optimization for large language model alignment. *arXiv preprint arXiv:2406.12606*, 2024g.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, and Joseph E Gonzalez. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6, 2023.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, and Joseph E Gonzalez. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv preprint arXiv:2403.04132*, 2024.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, and Sebastian Gehrmann. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

Sayak Ray Chowdhury, Anush Kini, and Nagarajan Natarajan. Provably robust dpo: Aligning language models with noisy feedback. *arXiv preprint arXiv:2403.00409*, 2024.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards. *arXiv preprint arXiv:2309.17400*, 2023.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. Free dolly: Introducing the world's first truly open instruction-tuned llm. *Company Blog of Databricks*, 2023.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, and Jean Maillard. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*, 2022.

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

Wenliang Dai, Junnan Li, D Li, AMH Tiong, J Zhao, W Wang, B Li, P Fung, and S Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. arxiv 2023. *arXiv preprint arXiv:2305.06500*, 2, 2023.

John Dang, Arash Ahmadian, Kelly Marchisio, Julia Kreutzer, Ahmet Üstün, and Sara Hooker. Rlhf can speak many languages: Unlocking multilingual preference optimization for llms. *arXiv preprint arXiv:2407.02552*, 2024.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, 2023.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*, 2023.

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

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

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

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

Rasool Fakoor, Pratik Chaudhari, and Alexander J Smola. P3o: Policy-on policy-off policy optimization. In *Uncertainty in artificial intelligence*, pages 1017–1027. PMLR, 2020.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, and Vishrav Chaudhary. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48, 2021.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Reinforcement learning for fine-tuning text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. *arXiv preprint arXiv:2212.05032*, 2022.

Tao Gong, Chengqi Lyu, Shilong Zhang, Yudong Wang, Miao Zheng, Qian Zhao, Kuikun Liu, Wenwei Zhang, Ping Luo, and Kai Chen. Multimodal-gpt: A vision and language model for dialogue with humans. *arXiv preprint arXiv:2305.04790*, 2023.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, and Yizhong Wang. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*, 2024.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, and Chenjie Gu. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998*, 2023.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, and Olli Saarikivi. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*, 2023.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, and Bilal Piot. Direct language model alignment from online ai feedback. *arXiv preprint arXiv:2402.04792*, 2024a.

Yiju Guo, Ganqu Cui, Lifan Yuan, Ning Ding, Jiexin Wang, Huimin Chen, Bowen Sun, Ruobing Xie, Jie Zhou, and Yankai Lin. Controllable preference optimization: Toward controllable multi-objective alignment. *arXiv preprint arXiv:2402.19085*, 2024b.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3326, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Neurips*, volume 33, pages 6840–6851, 2020.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, and Aidan Clark. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pages 30016–30030, 2022.

Jiwoo Hong, Noah Lee, and James Thorne. Orpo: Monolithic preference optimization without reference model. *arXiv preprint arXiv:2403.07691*, 2(4):5, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36, 2024.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023a.

Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. Towards mitigating llm hallucination via self reflection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1827–1843, 2023b.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, and Lucile Saulnier. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023a.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, and Florian Bressand. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. Llm-blender: Ensembling large language models with pairwise comparison and generative fusion. In *Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics (ACL 2023)*, 2023b.

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

Hyunwoo Kim, Youngjae Yu, Liwei Jiang, Ximing Lu, Daniel Khashabi, Gunhee Kim, Yejin Choi, and Maarten Sap. Prosocialdialog: A prosocial backbone for conversational agents. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4005–4029, 2022.

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

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:36652–36663, 2023.

Wouter Kool, Herke van Hoof, and Max Welling. Buy 4 reinforce samples, get a baseline for free! *Deep RL Meets Structured Prediction*, 2019.

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

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, and Kenton Lee. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 318–327, 2023.

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

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, and Huu Nguyen. The bigscience roots corpus: A 1.6 tb composite multilingual dataset. *Advances in Neural Information Processing Systems*, 35:31809–31826, 2022.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, and Matthias Gallé.

Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2023.

Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K Kummerfeld, and Rada Mihalcea. A mechanistic understanding of alignment algorithms: A case study on dpo and toxicity. *arXiv preprint arXiv:2401.01967*, 2024.

Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback. *arXiv preprint arXiv:2302.12192*, 2023.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, 2020.

Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Fanyi Pu, Jingkang Yang, Chunyuan Li, and Ziwei Liu. Mimic-it: Multi-modal in-context instruction tuning. *arXiv preprint arXiv:2306.05425*, 2023a.

Haonan Li, Fajri Koto, Minghao Wu, Alham Fikri Aji, and Timothy Baldwin. Bactrian-x: Multilingual replicable instruction-following models with low-rank adaptation. *arXiv preprint arXiv:2305.15011*, 2023b.

Lei Li, Zhihui Xie, Mukai Li, Shunian Chen, Peiyi Wang, Liang Chen, Yazheng Yang, Benyou Wang, and Lingpeng Kong. Silkie: Preference distillation for large visual language models. *arXiv preprint arXiv:2312.10665*, 2023c.

Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, and Jenny Chim. Starcoder: may the source be with you! *arXiv preprint arXiv:2305.06161*, 2023d.

Xiaochen Li, Zheng-Xin Yong, and Stephen H Bach. Preference tuning for toxicity mitigation generalizes across languages. *arXiv preprint arXiv:2406.16235*, 2024.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023e.

Ziniu Li, Tian Xu, Yushun Zhang, Zhihang Lin, Yang Yu, Ruoyu Sun, and Zhi-Quan Luo. Remax: A simple, effective, and efficient reinforcement learning method for aligning large language models. In *Forty-first International Conference on Machine Learning*, 2023f.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, and Ananya Kumar. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.

Xingxing Liang, Yang Ma, Yanghe Feng, and Zhong Liu. Ptr-ppo: Proximal policy optimization with prioritized trajectory replay. *arXiv preprint arXiv:2112.03798*, 2021.

Youwei Liang, Junfeng He, Gang Li, Peizhao Li, Arseniy Klimovskiy, Nicholas Carolan, Jiao Sun, Jordi Pont-Tuset, Sarah Young, and Feng Yang. Rich human feedback for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19401–19411, 2024.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024b.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pages 423–439. Springer, 2022.

Rosanne Liu, Dan Garrette, Chitwan Saharia, William Chan, Adam Roberts, Sharan Narang, Irina Blok, Rj Mical, Mohammad Norouzi, and Noah Constant. Character-aware models improve visual text rendering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16270–16297, 2023a.

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

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742, 2020.

Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, and Jianfeng Gao. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024c.

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

Holy Lovenia, Rahmad Mahendra, Salsabil Maulana Akbar, Lester James V Miranda, Jennifer Santoso, Elyanah Aco, Akhdan Fadhilah, Jonibek Mansurov, Joseph Marvin Imperial, and Onno P Kampman. Seacrowd: A multilingual multimodal data hub and benchmark suite for southeast asian languages. *arXiv preprint arXiv:2406.10118*, 2024.

Yinquan Lu, Wenhao Zhu, Lei Li, Yu Qiao, and Fei Yuan. Llamax: Scaling linguistic horizons of llm by enhancing translation capabilities beyond 100 languages. *arXiv preprint arXiv:2407.05975*, 2024.

Calvin Luo. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*, 2022.

Ziyang Luo, Can Xu, Pu Zhao, Qingfeng Sun, Xiubo Geng, Wenxiang Hu, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. Wizardcoder: Empowering code large language models with evol-instruct. *arXiv preprint arXiv:2306.08568*, 2023.

Soumi Maiti, Yifan Peng, Takaaki Saeki, and Shinji Watanabe. Speechlmscore: Evaluating speech generation using speech language model. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

James Manyika and Sissie Hsiao. An overview of bard: an early experiment with generative ai. *AI. Google Static Documents*, 2, 2023.

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

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, and Georg Ostrovski. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PMLR, 2016.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, and Hailey Schoelkopf. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, 2023.

Gabriel Mukobi, Peter Chatain, Su Fong, Robert Windesheim, Gitta Kutyniok, Kush Bhatia, and Silas Alberti. Superhf: Supervised iterative learning from human feedback. *arXiv preprint arXiv:2310.16763*, 2023.

Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, and Andrea Michi. Nash learning from human feedback. *arXiv preprint arXiv:2312.00886*, 2023.

Xuan-Phi Nguyen, Wenxuan Zhang, Xin Li, Mahani Aljunied, Qingyu Tan, Liying Cheng, Guanzheng Chen, Yue Deng, Sen Yang, and Chaoqun Liu. Seallms–large language models for southeast asia. *arXiv preprint arXiv:2312.00738*, 2023.

Jinjie Ni, Fuzhao Xue, Yuntian Deng, Jason Phang, Kabir Jain, Mahir Hitesh Shah, Zangwei Zheng, and Yang You. Instruction in the wild: A user-based instruction dataset. *GitHub repository*, 2023.

Aitor Ormazabal, Che Zheng, Cyprien de Masson d'Autume, Dani Yogatama, Deyu Fu, Donovan Ong, Eric Chen, Eugenie Lamprecht, Hai Pham, and Isaac Ong. Reka core, flash, and edge: A series of powerful multimodal language models. *arXiv preprint arXiv:2404.12387*, 2024.

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

Artemis Panagopoulou, Le Xue, Ning Yu, Junnan Li, Dongxu Li, Shafiq Joty, Ran Xu, Silvio Savarese, Caiming Xiong, and Juan Carlos Niebles. X-instructblip: A framework for aligning x-modal instruction-aware representations to llms and emergent cross-modal reasoning. *arXiv preprint arXiv:2311.18799*, 2023.

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

Jupinder Parmar, Shrimai Prabhumoye, Joseph Jennings, Mostofa Patwary, Sandeep Subramanian, Dan Su, Chen Zhu, Deepak Narayanan, Aastha Jhunjhunwala, and Ayush Dattagupta. Nemotron-4 15b technical report. *arXiv preprint arXiv:2402.16819*, 2024.

Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation. *arXiv preprint arXiv:2310.03739*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, and Jack Clark. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, and Susannah Young. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*, 2021.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140): 1–67, 2020.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022.

Javier Rando and Florian Tramèr. Universal jailbreak backdoors from poisoned human feedback. *arXiv preprint arXiv:2311.14455*, 2023.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, and Julian Schrittwieser. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

Takaaki Saeki, Soumi Maiti, Shinnosuke Takamichi, Shinji Watanabe, and Hiroshi Saruwatari. Speechbertscore: Reference-aware automatic evaluation of speech generation leveraging nlp evaluation metrics. *arXiv preprint arXiv:2401.16812*, 2024.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, and Tim Salimans. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, and Arun Raja. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2021.

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

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Shuaijie She, Shujian Huang, Wei Zou, Wenhao Zhu, Xiang Liu, Xiang Geng, and Jiajun Chen. Mapo: Advancing multilingual reasoning through multilingual alignment-as-preference optimization. *arXiv preprint arXiv:2401.06838*, 2024.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

Shivalika Singh, Freddie Vargus, Daniel Dsouza, Börje F Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, and Laura OMahony. Aya dataset: An open-access collection for multilingual instruction tuning. *arXiv preprint arXiv:2402.06619*, 2024.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

Saleh Soltan, Shankar Ananthakrishnan, Jack FitzGerald, Rahul Gupta, Wael Hamza, Haidar Khan, Charith Peris, Stephen Rawls, Andy Rosenbaum, and Anna Rumshisky. Alexatm 20b: Few-shot learning using a large-scale multilingual seq2seq model. *arXiv preprint arXiv:2208.01448*, 2022.

Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. Preference ranking optimization for human alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18990–18998, 2024.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

Yanshen Sun, Jianfeng He, Limeng Cui, Shuo Lei, and Chang-Tien Lu. Exploring the deceptive power of llm-generated fake news: A study of real-world detection challenges. *arXiv preprint arXiv:2403.18249*, 2024.

Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, and Yiming Yang. Aligning large multimodal models with factually augmented rlhf. *arXiv preprint arXiv:2309.14525*, 2023.

Wenpin Tang. Fine-tuning of diffusion models via stochastic control: entropy regularization and beyond. *arXiv preprint arXiv:2403.06279*, 2024.

Wenpin Tang and Hanyang Zhao. Score-based diffusion models via stochastic differential equations–a technical tutorial. *arXiv preprint arXiv:2402.07487*, 2024.

Yunhao Tang, Daniel Zhaohan Guo, Zeyu Zheng, Daniele Calandriello, Yuan Cao, Eugene Tarassov, Rémi Munos, Bernardo Ávila Pires, Michal Valko, and Yong Cheng. Understanding the performance gap between online and offline alignment algorithms. *arXiv preprint arXiv:2405.08448*, 2024a.

Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Rémi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Ávila Pires, and Bilal Piot. Generalized preference optimization: A unified approach to offline alignment. *arXiv preprint arXiv:2402.05749*, 2024b.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: an instruction-following llama model (2023). *URL https://github. com/tatsu-lab/stanford_alpaca*, 1(9), 2023.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, and Anja Hauth. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, and Juliette Love. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

SM Tonmoy, SM Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*, 2024.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, and Faisal Azhar. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, and Shruti Bhosale. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Masatoshi Uehara, Yulai Zhao, Tommaso Biancalani, and Sergey Levine. Understanding reinforcement learning-based fine-tuning of diffusion models: A tutorial and review. *arXiv preprint arXiv:2407.13734*, 2024a.

Masatoshi Uehara, Yulai Zhao, Kevin Black, Ehsan Hajiramezanali, Gabriele Scalia, Nathaniel Lee Diamant, Alex M Tseng, Tommaso Biancalani, and Sergey Levine. Fine-tuning of continuous-time diffusion models as entropy-regularized control. *arXiv preprint arXiv:2402.15194*, 2024b.

Ahmet Üstün, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, and Amr Kayid. Aya model: An instruction finetuned open-access multilingual language model. *arXiv preprint arXiv:2402.07827*, 2024.

Vladimir Vovk and Ruodu Wang. E-values: Calibration, combination and applications. *The Annals of Statistics*, 49(3):1736–1754, 2021.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8228–8238, 2024.

Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse kl: Generalizing direct preference optimization with diverse divergence constraints. *arXiv preprint arXiv:2309.16240*, 2023a.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. *arXiv preprint arXiv:2406.12845*, 2024a.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, and David Stap. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, 2022.

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

Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. Helpsteer2: Open-source dataset for training top-performing reward models. *arXiv preprint arXiv:2406.08673*, 2024b.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does LLM safety training fail? *Advances in Neural Information Processing Systems*, 36, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Martin Weyssow, Aton Kamanda, and Houari Sahraoui. Codeultrafeedback: An llm-as-a-judge dataset for aligning large language models to coding preferences. *arXiv preprint arXiv:2403.09032*, 2024.

Ronald J Williams. *Reinforcement-learning connectionist systems.* College of Computer Science, Northeastern University, 1987.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

Genta Indra Winata, Samuel Cahyawijaya, Zihan Liu, Zhaojiang Lin, Andrea Madotto, and Pascale Fung. Are multilingual models effective in code-switching? In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 142–153, 2021a.

Genta Indra Winata, Andrea Madotto, Zhaojiang Lin, Rosanne Liu, Jason Yosinski, and Pascale Fung. Language models are few-shot multilingual learners. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 1–15, 2021b.

Genta Indra Winata, Ruochen Zhang, and David Ifeoluwa Adelani. Miners: Multilingual language models as semantic retrievers. *arXiv preprint arXiv:2406.07424*, 2024.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*, 2023a.

Tianhao Wu, Banghua Zhu, Ruoyu Zhang, Zhaojin Wen, Kannan Ramchandran, and Jiantao Jiao. Pairwise proximal policy optimization: Harnessing relative feedback for llm alignment. *arXiv preprint arXiv:2310.00212*, 2023b.

Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score: Better aligning text-to-image models with human preference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2096–2105, 2023c.

Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-play preference optimization for language model alignment. *arXiv preprint arXiv:2405.00675*, 2024.

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

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

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *Forty-first International Conference on Machine Learning*, 2024a. URL https://openreview.net/forum?id=51iwkioZpn.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36, 2024b.

Jing Xu, Andrew Lee, Sainbayar Sukhbaatar, and Jason Weston. Some things are more cringe than others: Preference optimization with the pairwise cringe loss. *arXiv preprint arXiv:2312.16682*, 2023b.

Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. Is dpo superior to ppo for llm alignment? a comprehensive study. *arXiv preprint arXiv:2404.10719*, 2024c.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing. *arXiv preprint arXiv:2406.08464*, 2024d.

Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023.

Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. Regularizing hidden states enables learning generalizable reward model for llms. *arXiv preprint arXiv:2406.10216*, 2024.

Jia-Yu Yao, Kun-Peng Ning, Zhen-Hui Liu, Mu-Nan Ning, and Li Yuan. Llm lies: Hallucinations are not bugs, but features as adversarial examples. *arXiv preprint arXiv:2310.01469*, 2023a.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning. *arXiv preprint arXiv:2310.10683*, 2023b.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022a.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, and Burcu Karagol Ayan. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3): 5, 2022b.

Tianyu Yu, Jinyi Hu, Yuan Yao, Haoye Zhang, Yue Zhao, Chongyi Wang, Shan Wang, Yinxv Pan, Jiao Xue, and Dahai Li. Reformulating vision-language foundation models and datasets towards universal multimodal assistants. *arXiv preprint arXiv:2310.00653*, 2023.

Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, and Maosong Sun. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13807–13816, 2024.

Lifan Yuan, Ganqu Cui, Hanbin Wang, Ning Ding, Xingyao Wang, Jia Deng, Boji Shan, Huimin Chen, Ruobing Xie, and Yankai Lin. Advancing llm reasoning generalists with preference trees. *arXiv preprint arXiv:2404.02078*, 2024a.

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

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. Rrhf: Rank responses to align language models with human feedback without tears. *arXiv preprint arXiv:2304.05302*, 2023.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*, 2023a.

Dong Zhang, Zhaowei Li, Shimin Li, Xin Zhang, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. Speechalign: Aligning speech generation to human preferences. *arXiv preprint arXiv:2404.05600*, 2024a.

Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 543–553, 2023b.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*, 2024b.

Shu Zhang, Xinyi Yang, Yihao Feng, Can Qin, Chia-Chih Chen, Ning Yu, Zeyuan Chen, Huan Wang, Silvio Savarese, and Stefano Ermon. Hive: Harnessing human feedback for instructional visual editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9026–9036, 2024c.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, and Xi Victoria Lin. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. Wildchat: 1m chatgpt interaction logs in the wild. *arXiv preprint arXiv:2405.01470*, 2024.

Yao Zhao, Mikhail Khalman, Rishabh Joshi, Shashi Narayan, Mohammad Saleh, and Peter J Liu. Calibrating sequence likelihood improves conditional language generation. In *The eleventh international conference on learning representations*, 2022.

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

Chujie Zheng, Ziqi Wang, Heng Ji, Minlie Huang, and Nanyun Peng. Weak-to-strong extrapolation expedites alignment. *arXiv preprint arXiv:2404.16792*, 2024a.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, and Eric Xing. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36, 2024b.

Zhisheng Zheng, Puyuan Peng, Ziyang Ma, Xie Chen, Eunsol Choi, and David Harwath. Bat: Learning to reason about spatial sounds with large language models. *arXiv preprint arXiv:2402.01591*, 2024c.

Han Zhong, Guhao Feng, Wei Xiong, Li Zhao, Di He, Jiang Bian, and Liwei Wang. Dpo meets ppo: Reinforced token optimization for rlhf. *arXiv preprint arXiv:2404.18922*, 2024.

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

Yiyang Zhou, Chenhang Cui, Rafael Rafailov, Chelsea Finn, and Huaxiu Yao. Aligning modalities in vision large language models via preference fine-tuning. *arXiv preprint arXiv:2402.11411*, 2024b.

Banghua Zhu, Evan Frick, Tianhao Wu, Hanlin Zhu, Karthik Ganesan, Wei-Lin Chiang, Jian Zhang, and Jiantao Jiao. Starling-7b: Improving helpfulness and harmlessness with rlaif. In *First Conference on Language Modeling*, 2024.

Zifeng Zhuang, Kun Lei, Jinxin Liu, Donglin Wang, and Yilang Guo. Behavior proximal policy optimization. *arXiv preprint arXiv:2302.11312*, 2023.

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