Policy Optimization in RLHF: The Impact of Out-of-preference Data

Ziniu Li*[1,2], Tian Xu*[3,4,5], and Yang Yu†[3,4,5]

[1]The Chinese University of Hong Kong, Shenzhen
[2]Shenzhen Research Institute of Big Data
[3]National Key Laboratory for Novel Software Technology, Nanjing University
[4]School of Artificial Intelligence, Nanjing University
[5]Polixir.ai

December 19, 2023

Abstract

Aligning intelligent agents with human preferences and values is important. This paper examines two popular alignment methods: Direct Preference Optimization (DPO) and Reward-Model-Based Policy Optimization (RMB-PO). A variant of RMB-PO, referred to as RMB-PO+ is also considered. These methods, either explicitly or implicitly, learn a reward model from preference data and differ in the data used for policy optimization to unlock the generalization ability of the reward model. In particular, compared with DPO, RMB-PO additionally uses policy-generated data, and RMB-PO+ further leverages new, preference-free data. We examine the impact of such out-of-preference data. Our study, conducted through controlled and synthetic experiments, demonstrates that DPO performs poorly, whereas RMB-PO+ performs the best. In particular, even when providing the policy model with a good feature representation, we find that policy optimization with adequate out-of-preference data significantly improves performance by harnessing the reward model's generalization capabilities.[1]

1 Introduction

Developing intelligent agents requires alignment with human values [Russell and Norvig, 2010]. A standard practice involves providing a human preference dataset for the agent to learn from. According to utility theory [Fishburn et al., 1979], preference is connected with a certain reward function. Currently, there are two kinds of alignment methods: the reward-model-free approach, such as Direct Preference Optimization (DPO) [Rafailov et al., 2023] and Identity Policy Optimization

*Equal contribution. Author ordering is determined by coin flip. Emails: ziniuli@link.cuhk.edu.cn and xut@lamda.nju.edu.cn

†Corresponding author. Email: yuy@nju.edu.cn

[1]Code is available at https://github.com/liziniu/policy_optimization

(IPO) [Azar et al., 2023], and the reward-model-based approaches, exemplified by the so-called Reinforcement Learning from Human Feedback (RLHF) [Christiano et al., 2017, Stiennon et al., 2020, OpenAI, 2023]. Specifically, the first method directly learns a decision policy from a preference dataset, without the need to train a separate reward model. On the other hand, the second approach involves training a reward model from the preference data and subsequently employing policy optimization (e.g., reinforcement learning) on new, *preference-free* data. While both approaches have been shown to improve the decision policy by leveraging preference data, the superiority of one method over the other remains unclear, a determination crucial for future advancements.

We briefly explain why this question is hard to answer. The main difficulty lies in the intuition that the policy model, powered by a (pre-trained) neural network or learned from preference-data, possesses a certain generalization ability. Therefore, it is unclear whether policy optimization on additional data, with an imperfect yet generalizable reward model, is necessary.

In this paper, we explore the above question through controlled synthetic experiments. We study two contextual bandit tasks with linear function approximation and neural function approximation, respectively. Our studies take a stochastic optimization viewpoint. Concretely, we argue that the policy optimization in these methods corresponds to different versions of stochastic approximation of the (expected) reward maximization problem. In particular, with a reward model, stochastic approximation can be performed more accurately with out-of-preference data, which interestingly does not increase the sample complexity of preference data.

Our experiments validate the above ideas. One main experiment, where we manually ensure that the policy shares the *same* good feature representation with the reward model, shows that policy optimization with additional out-of-preference data still improves generalization performance. Other experiments also support this claim. Finally, we provide a discussion about this phenomenon with reference to other fields, such as imitation learning and reinforcement learning.

2 Problem Formulation

We consider the so-called contextual bandit [Langford and Zhang, 2007, Lu et al., 2010] formulation for the alignment problem. Let s and a be the state (i.e., context) and decision action, respectively. We aim to obtain a decision policy π that acts optimally in terms of reward maximization:

$$\pi^\star \leftarrow \operatorname*{argmax}_{\pi} \mathbb{E}_{s\sim\rho(\cdot)}\mathbb{E}_{a\sim\pi(\cdot|s)}\left[r(s,a)\right], \tag{1}$$

where the symbol ρ denotes the state distribution, and r is the ground truth reward function. In the context of alignment, the difficulty is that the reward function is unknown but only preferences over two actions are observed. Typically, the Bradley-Terry-Luce [Bradley and Terry, 1952] assumption is used:

$$\mathbb{P}\left(a > a'|s\right) = \frac{\exp(r(s,a))}{\exp(r(s,a)) + \exp(r(s,a'))},$$

where the symbol $a > a'$ means that a is more preferred compared with a'. Given a preference dataset $D_{\text{pref}} = \{(s_i, a_i, a_i')\}_{i=1}^n$, where $a_i > a_i'$ is assumed without loss of generality, the reward learning objective is

$$\widehat{r} \leftarrow \operatorname*{argmax}_{r} \sum_{i=1}^{n} \log\left(\sigma\left(r(s_i,a_i) - r(s_i,a_i')\right)\right), \quad (s_i,a_i,a_i') \sim D_{\text{pref}}. \tag{2}$$

where $\sigma(\cdot)$ is the sigmoid function. Let π_{ref} be a reference policy model and $\beta > 0$ be a hyper-parameter. The Reward-Model-Based Policy Optimization (RMB-PO) approach optimizes the

policy on the policy-generated data by sampling states from the preference dataset:

$$\widehat{\pi}_{\text{RMB-PO}} \leftarrow \operatorname*{argmax}_{\pi} \sum_{i=1}^{n} \mathbb{E}_{a\sim\pi(\cdot|s_i)} \left[\widehat{r}(s_i, a)\right] + \beta D_{\text{KL}}(\pi(\cdot|s_i), \pi_{\text{ref}}(\cdot|s_i)), \quad s_i \sim D_{\text{pref}}. \tag{3}$$

In [Ouyang et al., 2022, Touvron et al., 2023], a variant of RMB-PO, refereed to as RMB-PO+ in this paper, further leverages a new, *preference-free* dataset $D_{\text{new}} = \{s_j\}_{j=1}^{m}$ for policy optimization:

$$\widehat{\pi}_{\text{RMB-PO+}} \leftarrow \operatorname*{argmax}_{\pi} \sum_{j=1}^{m} \mathbb{E}_{a\sim\pi(\cdot|s_j)} \left[\widehat{r}(s_j, a)\right] + \beta D_{\text{KL}}(\pi(\cdot|s_j), \pi_{\text{ref}}(\cdot|s_j)), \ s_j \sim D_{\text{new}}. \tag{4}$$

Note that the dataset D_{new} is cheap to obtain and usually $m \geq n$ [Ouyang et al., 2022].

In addition to the above methods, the Direct Preference Optimization (DPO) is developed in [Rafailov et al., 2023], which combines Equation (2) and Equation (3) to a single supervised objective:

$$\widehat{\pi}_{\text{DPO}} \leftarrow \operatorname*{argmax}_{\pi} \sum_{i=1}^{n} \log \sigma \left(\beta \log \frac{\pi(a_i|s_i)}{\pi_{\text{ref}}(a_i|s_i)} - \beta \log \frac{\pi(a_i'|s_i)}{\pi_{\text{ref}}(a_i'|s_i)} \right), \quad (s_i, a_i, a_i') \sim D_{\text{pref}}. \tag{5}$$

Note that DPO uses solely the preference data, without policy-generated data or preference-free data.

2.1 A Stochastic Optimization Perspective

Based on the above formulations, we argue that all the mentioned methods are stochastic approximations to Equation (1), with errors arising from three sources:

1) the reward model error $|\widehat{r}(s, a) - r(s, a)|$;

2) the estimation error when using finite samples to calculate the expectation $\mathbb{E}_{a\sim\pi(\cdot|s)}[\cdot]$;

3) the estimation error when using finite samples to calculate the expectation $\mathbb{E}_{s\sim\rho(\cdot)}[\cdot]$.

The first error primarily results from having only finite preference data, which is expensive to obtain. Compared with DPO, RMB-PO aims to mitigate the second error, while RMB-PO+ further reduces the third error. We note that RMB-PO and RMB-PO+ do not increase the sample complexity of preference data but only incur additional computation steps.

In our study, we assume that the action space $\mathcal{A}$ is finite and the expectation $\mathbb{E}_{a\sim\pi(\cdot|s)}[\cdot]$ is easy to obtain. For applications with combinatorial action spaces (e.g., large language models), computing this expectation is generally intractable, and practical algorithms may employ the policy to generate stochastic trials to approximate the expectation; refer to, e.g., [Li et al., 2023]. We note that even in this case, the sample complexity of preference data does not increase.

Another viewpoint is that the RMB-PO and RMB-PO+ methods incorporate additional policy-generated data and preference-free data, essentially employing a form of data augmentation. We summarize the effect of such data augmentation in Table 1.

	DPO	RMB-PO	RMB-PO+
Improvement of Stochastic Approximation in State Space	✗	✗	✓
Improvement of Stochastic Approximation in Action Space	✗	✓	✓

Table 1: Illustration of the data augmentation effect in RMB-PO and RMB-PO+.

3 Experiments

In this section, we conduct numerical experiments to validate the improvement of RMB-PO and RMB-PO+ by better stochastic approximation. All of our experiments are run with 10 different random seeds (2021-2030), and the averaged results are reported. Note that we set π_{REF} to be a policy with a uniform action distribution in all experiments and $\beta = 0.01$ for all methods. Besides, we use a policy with a uniform action distribution to collect the preference data.

3.1 Linear Bandit

We study a linear bandit task, where we have $r(s,a) = \phi_r(s,a)^\top \theta_r^\star$, with $\phi_r(s,a) \in \mathbb{R}^d$ denoting the feature representation and $\theta_r^\star \in \mathbb{R}^d$ as the parameter. In this case, the reward learning optimization problem is convex, so we use CVXPY [Diamond and Boyd, 2016] to find the solution $\widehat{r}$. In particular, we use the feature map $\phi_r(s,a)$ and the parameter $\theta_r^\star$ as

$$\phi_r(s,a) = \left((a+1)\cdot\cos(s\cdot\pi), \frac{1}{a+1}\cdot\sin(s\cdot\pi)\right)^\top, \quad \theta_r^\star = (1,2)^\top,$$

where $s \in \mathcal{S} = [0,1]$ and $a \in \mathcal{A} = \{0,1,2,3\}$. A uniform distribution over $\mathcal{S}$ is studied. For the policy, we consider the parameterization

$$\pi(a|s) = \frac{\exp(\phi_\pi(s,a)^\top \theta_\pi)}{\sum_{a'} \exp(\phi_\pi(s,a')^\top \theta_\pi)},$$

with $\phi_\pi(s,a)$ and θ_π both in $\mathbb{R}^2$. In this case, the policy optimization problem is a non-convex problem, but the gradient domination condition holds [Agarwal et al., 2021]. We use the gradient ascent method with the AdaGrad optimizer [Duchi et al., 2011] (a step size of 0.1 is used).

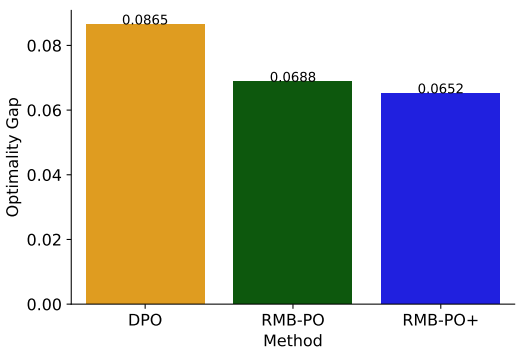


Figure 1: Optimality gap with $\phi_\pi = \phi_r$.

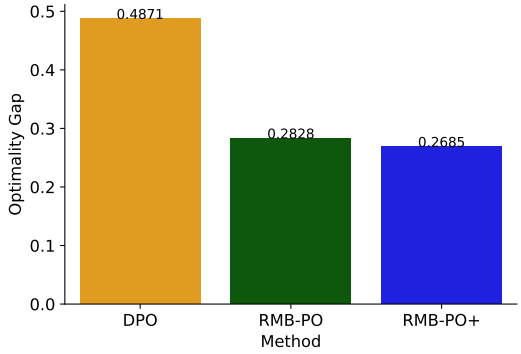


Figure 2: Optimality gap with $\phi_\pi \neq \phi_r$.

We examine two scenarios. In the first scenario, there is no feature mismatch between the reward and policy models, i.e., $\phi_\pi = \phi_r$. In the second, we use a different feature map for policy:

$$\phi_\pi(s,a) = \left((a+1)\cdot\sin(s\cdot\pi), \frac{1}{a+1}\cdot\cos(s\cdot\pi)\right)^\top.$$

In our experiments, we set the size of preference data to be $n = 20$ and the size of preference-free data to be $m = 10n$, resulting in training accuracy of the reward model ranging from 60% to 80%. We display the optimality gap $|r(\pi^\star) - r(\widehat{\pi})|$ (the smaller, the better) in Figure 1 and Figure 2, where $r(\pi)$ is the expected performance, i.e., $r(\pi) = \mathbb{E}_{s\sim\rho(\cdot)}\mathbb{E}_{a\sim\pi(\cdot|s)}[r(s,a)]$ (in our experiments, we use 5000 sampled states to approximate this expectation).

From Figure 1, we see that even though the policy model is provided with a good feature (e.g., in Figure 1), RMB-PO methods can benefit from out-of-preference data. In the case where $\phi_\pi \neq \phi_r$ in Figure 2, we find that RMB-PO+ is better than RMB-PO by leveraging additional preference-free data. Thus, we believe it is crucial to learn the optimal action (as inferred by the reward model) on out-of-preference data, even when the two models share the same good feature.

Following the same setup, we provide ablation studies regarding the size of preference-free data used in RMB-PO+. See the results in Figure 3 and Figure 4. We find that the previous conclusions still hold true.

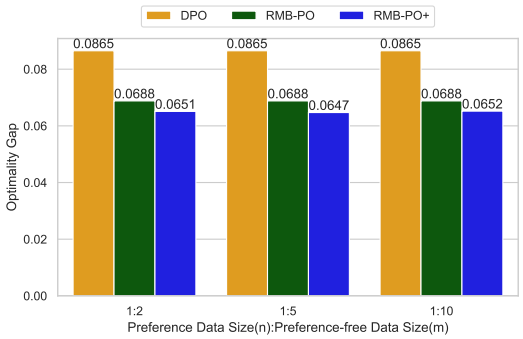


Figure 3: Optimality gap with $\phi_\pi = \phi_r$.

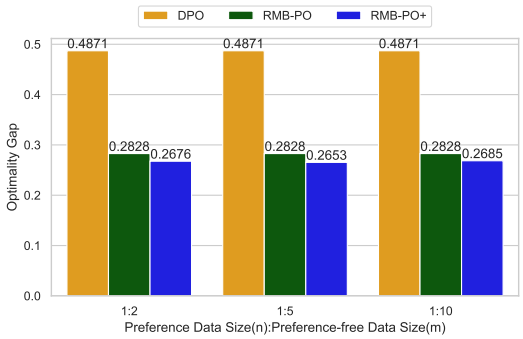
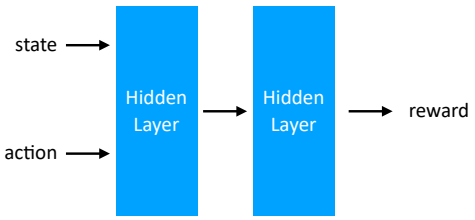


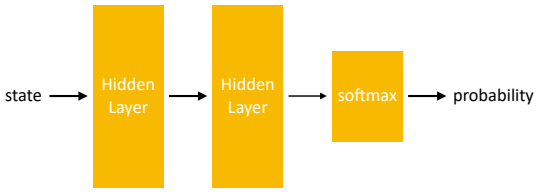
Figure 4: Optimality gap with $\phi_\pi \neq \phi_r$.

3.2 Neural Bandit

In this section, we study a neural bandit problem. Specifically, we study the case where $r(s, a) = f_{\theta_r^\star}(s, a)$, with $f_{\theta_r^\star}$ being a fixed 1-hidden-layer multi-layer perceptron (MLP) neural network, having a hidden size of 64. For reward learning, we use a 2-hidden-layer MLP with a hidden size of 64, and the policy network is also a 2-hidden-layer MLP with a hidden size of 64. We consider a continuous state space $\mathcal{S} = [-1, 1]^{50}$ and a discrete action space $\mathcal{A} = \{0, 1, 2, \ldots, 9\}$. The state distribution ρ is uniform and one-hot feature representation for actions is used.



(a) Reward neural network.



(b) Policy neural network.

Figure 5: Architectures of the reward and policy models.

We note that, unlike in the linear bandit case where we could fix the feature representations of the reward and policy models to be the same, in this case, the feature representations of the reward and policy models are purely learned from the given data. The architectures of the reward and policy models are shown in Figure 5. All neural networks are optimized using the Adam optimizer [Kingma and Ba, 2015] with a step size of 10^{-3}.

We run experiments with varying sizes of preference-free data m while fixing the preference data size at $n = 50$. We report the results in Figure 6. First, we observe that RMB-PO and RMB-PO+

significantly outperform DPO. Furthermore, simply using a preference-free data size that is twice as large already improves performance over RMB-PO, and further scaling does not help too much.

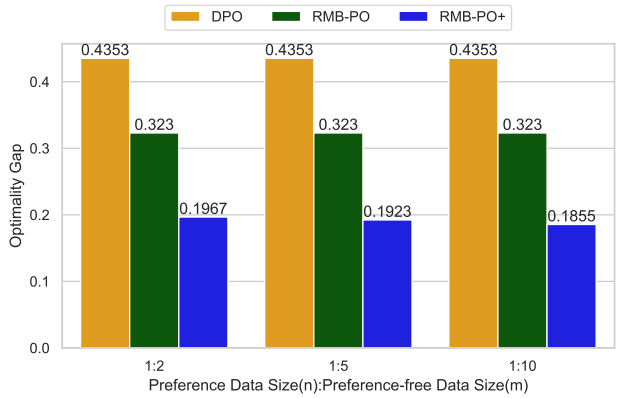


Figure 6: Optimality gap of learned policies in the neural bandit task.

4 Discussion

Our results are related to imitation learning [Osa et al., 2018], where the goal is to learn a policy from expert demonstrations. A popular approach to achieve this goal is through behavioral cloning (BC) [Pomerleau, 1991], which trains a policy model by maximizing the likelihood of expert data. Note that the working mechanism of BC is quite similar to DPO, as in Equation (5), where the likelihood of positively preferred actions is increased and that of negatively preferred actions is decreased:

$$\pi_{\mathrm{BC}} \leftarrow \operatorname*{argmax}_{\pi} \sum_{i=1}^{n} \log \pi(a_i|s_i), \quad (s_i, a_i) \sim D_{\mathrm{E}},$$

where D_{E} is the expert dataset. In contrast, Ghasemipour et al. [2019] showed that adversarial imitation learning (AIL) methods, such as GAIL [Ho and Ermon, 2016], leverage a recovered reward function to perform policy optimization on "out-of-expert-data" through online interaction, significantly improving performance. Following the formulation in [Xu et al., 2022], the training objective of reward-model-based AIL can be re-formulated as

$$\pi_{\mathrm{AIL}} \leftarrow \operatorname*{argmin}_{\pi} \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \left| d_{\pi}(s,a) - \widehat{d}_{\mathrm{E}}(s,a) \right|,$$

where $\widehat{d}_{\mathrm{E}}$ is the empirical state-action distribution estimated from D_{E}, and $d_{\pi}(s,a)$ is obtained from online interaction. For the above optimization problem, states beyond those in the expert dataset are used. We notice that Xu et al. [2022] theoretically proved that AIL, by leveraging additional online data for imitation, can outperform BC.

Additionally, our research is related to transition-model-based reinforcement learning (RL) methods, where the goal is to find an optimal policy through interactions with environments. Many empirical successes suggest that transition-model-based approaches are superior in terms of sample complexity [Luo et al., 2019, Janner et al., 2019]. We do not aim to present a detailed discussion since RL involves lots of concepts and notations. Instead, we would like to highlight that our findings align with the understanding that additional policy optimization on transition-model-generated data is helpful. We would like to refer readers to [Hafner et al., 2020, Schrittwieser et al., 2020, Yu et al.,

2020, Luo et al., 2023] for the effect of data augmentation in transition-model-based RL methods.

Finally, we note that compared with reward-model-free methods such as DPO [Rafailov et al., 2023], reward-model-based policy optimization (RMB-PO) methods do not require extra preference annotation. For applications such as large language models (LLMs), training and storing a reward model has been shown to be highly efficient, as demonstrated in [Yao et al., 2023]. The primary challenge in RMB-PO lies in the huge action space during policy optimization. However, this issue can be effectively addressed by computationally efficient methods like those proposed by [Dong et al., 2023, Li et al., 2023].

5 Conclusion

Our results underscore the importance of policy optimization on out-of-preference data to unlock the reward's generalization capacity. Otherwise, the policy may conflict with the learned reward model on out-of-preference data and suffer a poor performance.

References

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *The Journal of Machine Learning Research*, 22(1):4431–4506, 2021.

Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. A general theoretical paradigm to understand learning from human preferences. *arXiv preprint arXiv:2310.12036*, 2023.

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.

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*, pages 4299–4307, 2017.

Steven Diamond and Stephen Boyd. Cvxpy: A python-embedded modeling language for convex optimization. *The Journal of Machine Learning Research*, 17(1):2909–2913, 2016.

Hanze Dong, Wei Xiong, Deepanshu Goyal, 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.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Peter C Fishburn, Peter C Fishburn, et al. *Utility theory for decision making*. Krieger NY, 1979.

Seyed Kamyar Seyed Ghasemipour, Richard S. Zemel, and Shixiang Gu. A divergence minimization perspective on imitation learning methods. In *Proceedings of the 3rd Conference on Robot Learning*, pages 1259–1277, 2019.

Danijar Hafner, Timothy P. Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *Proceedings of the 8th International Conference on Learning Representations*, 2020.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems 29*, pages 4565–4573, 2016.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. In *Advances in neural information processing systems 32*, pages 12498–12509, 2019.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

John Langford and Tong Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. *Advances in Neural Information Processing Systems 20*, 2007.

Ziniu Li, Tian Xu, Yushun Zhang, Yang Yu, Ruoyu Sun, and Zhi-Quan Luo. Remax: A simple, effective, and efficient method for aligning large language models. *arXiv preprint arXiv:2310.10505*, 2023.

Tyler Lu, Dávid Pál, and Martin Pál. Contextual multi-armed bandits. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, pages 485–492, 2010.

Fan-Ming Luo, Tian Xu, Xingchen Cao, and Yang Yu. Reward-consistent dynamics models are strongly generalizable for offline reinforcement learning. *arXiv preprint arXiv:2310.05422*, 2023.

Yuping Luo, Huazhe Xu, Yuanzhi Li, Yuandong Tian, Trevor Darrell, and Tengyu Ma. Algorithmic framework for model-based deep reinforcement learning with theoretical guarantees. In *Proceedings of the 7th International Conference on Learning Representations*, 2019.

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

Takayuki Osa, Joni Pajarinen, Gerhard Neumann, J. Andrew Bagnell, Pieter Abbeel, and Jan Peters. An algorithmic perspective on imitation learning. *Foundations and Trends in Robotic*, 7 (1-2):1–179, 2018.

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

Dean Pomerleau. Efficient training of artificial neural networks for autonomous navigation. *Neural Computation*, 3(1):88–97, 1991.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

Stuart J Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. London, 2010.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

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.

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

Tian Xu, Ziniu Li, Yang Yu, and Zhi-Quan Luo. Understanding adversarial imitation learning in small sample regime: A stage-coupled analysis. *arXiv preprint arXiv:2208.01899*, 2022.

Zhewei Yao, Reza Yazdani Aminabadi, Olatunji Ruwase, Samyam Rajbhandari, Xiaoxia Wu, Ammar Ahmad Awan, Jeff Rasley, Minjia Zhang, Conglong Li, Connor Holmes, et al. Deepspeed-chat: Easy, fast and affordable rlhf training of chatgpt-like models at all scales. *arXiv preprint arXiv:2308.01320*, 2023.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *Advances in Neural Information Processing Systems 33*, pages 14129–14142, 2020.