The Human Factor: Addressing Diversity in Reinforcement Learning from Human Feedback

How can RLHF deal with possibly conflicting feedback?

Javier Páez Franco[1]
Supervisors: Dr. Luciano Cavalcante Siebert[1]**, Antonio Mone**[1]
[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 23, 2024

Name of the student: Javier Páez Franco
Final project course: CSE3000 Research Project
Thesis committee: Dr. Luciano Cavalcante Siebert, Antonio Mone, Dr. Wendelin Böhmer

Abstract

Reinforcement Learning from Human Feedback (RLHF) is a promising approach to training agents to perform complex tasks by incorporating human feedback. However, the quality and diversity of this feedback can significantly impact the learning process. Humans are highly diverse in their preferences, expertise, and capabilities. This paper investigates the effects of conflicting feedback on the agent's performance. We analyse the impact of environmental complexity and examine various query selection strategies. Our results show that RLHF performance rapidly degrades with even minimal conflicting feedback in simple environments, and current query selection strategies are ineffective in handling feedback diversity. We thus conclude that addressing diversity is crucial for RLHF, suggesting alternative reward modelling approaches are needed. Full code is available on GitHub.

Acronyms

A2C Advantage Actor Critic

LLM Large Language Model

PPO Proximal Policy Optimization

RL Reinforcement Learning

RLHF Reinforcement Learning from Human Feedback

TRPO Trust Region Policy Optimization

1 Introduction

Reinforcement Learning (RL) is an area of Machine Learning where we attempt to optimize the actions or decisions that an agent takes to navigate through an environment via trial and error [1]. RL has been very successful in highly complex environments when provided with an optimal reward function [2, 3]. However, this reward function is typically sophisticated and challenging to design manually [4]. Reinforcement Learning from Human Feedback (RLHF) aims to model a reward function from human feedback, addressing not only the limitations of traditional reward function engineering but also benefitting agent alignment. Once we learn the reward model, we can use RL to train the agent. RLHF has proven successful in various applications, ranging from robotics control [5] and gaming [6] to chatbots [7, 8]. Notably, RLHF has emerged as a crucial strategy for fine-tuning Large Language Models (LLMs) such as ChatGPT [9, 10].

However, despite its successes, RLHF is not without its challenges. Fine-tuned LLMs are known to produce biased, inaccurate, and harmful responses [9, 11, 12, 13]. The difficulties in obtaining high-quality human feedback, managing inconsistent, diverse, and noisy feedback, and ensuring scalability are well-documented [14]. Moreover, the simplicity of the reward model function in RLHF fails to capture and balance the preferences of individuals, thereby overlooking the rich diversity of human preferences [15, 16]. This oversight leads to one of the major challenges in RLHF - dealing with diverse preferences. Evaluators often disagree, creating conflicting feedback; for instance, during Anthropic's LLM training, the agreement rate between researchers and crowd workers was as low as 63% [17], or OpenAI, that found

that the inter-annotator agreement rates among training labellers were at $72.6\% \pm 1.6\%$ [8].

Despite the criticality, most of the latest RLHF approaches ignore the consideration of the diversity in human preference, treating these differences as noise [7, 18, 19]. As a result, when preferences differ, the majority wins, greatly restricting and potentially subduing the opinions of the minorities, leading to social biases. To mitigate this issue, several research approaches have been recently made. *Safe RLHF* [20] decouples human preferences regarding helpfulness and harmlessness, significantly reducing harmful responses. *MaxMin-RLHF* [16] learns a mixture of preference distributions via an expectation-maximization algorithm. *Nash-MD* [21] attempts to fully represent the richness of human preferences by achieving Nash equilibrium in the preference model. In addition, other approaches attempt to learn multiple reward functions, such as [22, 23]. Another line of research focuses on the consensus-based algorithm for aggregating human representations [17, 24].

While these approaches have shown promise in alternative reward modelling, there has been no comprehensive evaluation of the actual effect and relevance of conflicting data, especially concerning single utility RLHF, the current state-of-the-art. We lack a thorough understanding of how diversity influences the overall objective. Consequently, in this paper, we aim to answer the research question:

How can RLHF deal with possibly conflicting feedback coming from multiple individuals?

This paper's contributions are:

- To compare the performance of RLHF when using different degrees of conflicting feedback.
- To evaluate how the complexity of the environment impacts RLHF's ability to handle conflicting feedback.
- To study the effectiveness of RLHF's query selection strategies in handling diversity.

This study focuses solely on the original RLHF algorithm. Although alternative reward modelling techniques have been proposed, as previously mentioned, their novelty, complexity, and data gathering requirements have limited their application to research settings only. In addition, we will use Proximal Policy Optimization (PPO), a state-of-the-art RL algorithm, to train the agent after learning the reward model.

The remainder of this paper is structured as follows. In Section 2, we provide a formal description of RLHF and PPO. The evaluation framework is introduced in Section 3. Section 4 details the experimental setup along with the results. Section 5 discusses how diversity influences RLHF based on the prior experiments. Then, Section 6 details responsible research practices applied during this study. Finally, Section 7 concludes the paper by summarizing the findings and discussing some potential directions for further research.

2 Preliminaries

To understand the context and mechanisms underlying the research presented in this paper, it is crucial to explore the techniques that our work builds upon. This section provides a concise overview of the methods that underpin our work: PPO and RLHF.

2.1 PPO

Proximal Policy Optimization (PPO) is an on-policy reinforcement learning policy optimization method [25] that combines concepts from Advantage Actor Critic (A2C) [26], by having multiple workers, and Trust Region Policy Optimization (TRPO) [27], by using a trust region to improve the actor. The main idea is that PPO restricts the degree to which a policy can change during each update, thereby reducing the risk of harmful updates that can cause performance to deteriorate severely. Mathematically, Schulman et al. [25] defines the main objective as:

$$L^{CLIP}(\theta) = \hat{\mathbb{E}}_t \left[\min(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t) \right] \tag{1}$$

where epsilon is a hyperparameter, $\hat{A}_t$ is an estimator of the advantage function, and $r_t(\theta)$ is the ratio of probabilities to take action a_t between the new and the old policy.

2.2 RLHF

The current RLHF approaches [7, 8, 28, 29] fit a single reward function to the human preferences while simultaneously training a policy to optimize the current predicted reward function, as depicted in Figure 1.

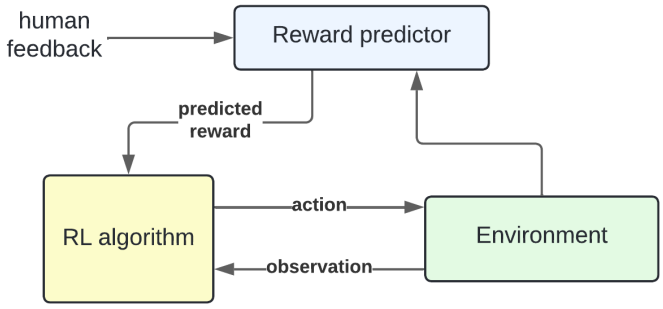


Figure 1: Schematic illustration of RLHF [4]. The **reward predictor** is trained from human feedback with respect to comparisons of trajectory segments from the **environment**, and the **RL algorithm** maximizes the predicted reward.

We consider an agent interacts with an environment over a sequence of steps; at each time t, the agent receives an observation $o_t \in \mathcal{O}$ from the environment and then sends an action $a_t \in \mathcal{A}$ to the environment. The resulting sequence $y = ((o_0, a_0), (o_1, a_1), \ldots, (o_{k-1}, a_{k-1})) \in (\mathcal{O} \times \mathcal{A})^k$ is called *trajectory segment*. The goal of the agent is to produce trajectories which are preferred by a human overseer.

The human overseer is given two trajectory segments to indicate which segment they prefer. The human judgements are recorded in a database $\mathcal{D}$ of triplets (y^1, y^2, μ), where y^1, y^2 are trajectory segments and μ is a distribution indicating which segment the human prefers.

The human's probability of preferring a segment y^1 over y^2 can be expressed as a preference predictor, adhering to the Bradley-Terry model [30]:

$$\hat{P}(y^1 \succ y^2) = \frac{\exp(\sum \hat{r}(o_t^1, a_t^1))}{\exp(\sum \hat{r}(o_t^1, a_t^1)) + \exp(\sum \hat{r}(o_t^2, a_t^2))} = \sigma(\sum \hat{r}(o_t^1, a_t^2) - \sum \hat{r}(o_t^2, a_t^2)) \quad (2)$$

where $\hat{r}$ is the latent reward model and $\sigma(x) = 1/(1 + \exp(-x))$ is the logistic function.

We choose $\hat{r}$ to minimize the cross-entropy loss between the predictions and human labels:

$$\text{loss}(\hat{r}) = -\sum_{(y^1, y^2, \mu) \in \mathcal{D}} \mu(1) \log \hat{P}[y^1 \succ y^2] + \mu(2) \log \hat{P}[y^2 \succ y^1] \quad (3)$$

The parameters of $\hat{r}$ can be optimized via supervised learning to fit the comparisons collected from humans. After using $\hat{r}$ to compute rewards, we are left with a traditional reinforcement learning problem. Typically, the PPO algorithm is used to train the policy of the agent [4].

RLHF's pseudocode is described in Algorithm 1. It is important to note that the standard RLHF algorithm is divided into two stages: (1) reward learning and (2) RL training. The reward learning phase is further divided into two steps: (i) query generation and (ii) the training of the reward function. This latter step is based on the oracle's feedback, as per Equations 2 and 3. The RL training phase is more conventional and involves running an RL algorithm, often PPO, with the currently trained reward function.

Algorithm 1 Generic RLHF Algorithm [4]

Initialize parameters θ (policy), ϕ (critic), and ψ (reward)
Initialize replay buffer $\mathcal{B}$ with randomly-generated trajectories
Let $\mathcal{D}$ be the database with the human judgements
for $i = 1 : N$ **do**
    // Reward learning
    Generate queries from $\mathcal{B}$
    Update $\mathcal{D}$ with answers to queries from the oracle(s)
    Update ψ using $\mathcal{D}$
    // RL training
    Update $\mathcal{B}$ with new trajectories generated with π_θ
    Update ϕ (critic) and θ (actor) using $\mathcal{R}_\psi$ and $\mathcal{B}$
end for

2.2.1 Query Selection Strategies

A crucial aspect of RLHF is how queries (i.e. trajectory segments) are chosen. There are two primary strategies:

- **Random selection**: We select queries uniformly at random, without any specific criteria or bias.
- **Active selection**: We prioritize queries with the highest variance of rewards from the learned model. The goal is to minimize the uncertainty of the predictions quickly.

Christiano et al. [28] and Gleave et al. [31] show that neither of these methods is universally superior. The effectiveness of each can vary depending on the context and the nature of the data. Therefore, it is essential to understand how both strategies handle conflicting data.

3 Methods

This section outlines the methodology proposed in this study. We discuss how to model diverse preferences and the technical details around implementing the RLHF algorithm. The methodology is summarized in Figure 2.

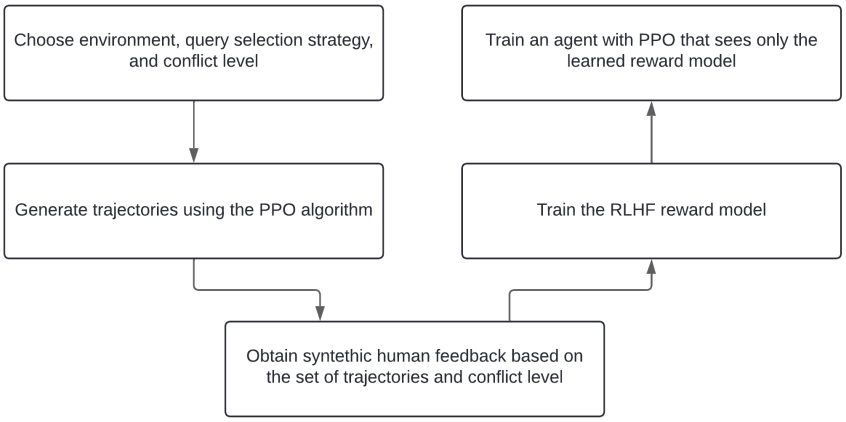


Figure 2: Flowchart detailing the methodology.

3.1 Conflicting Human Feedback

The first step of the RLHF algorithm is the generation of a set of trajectories. These were generated using PPO. While other RL algorithms could have been considered, PPO has already demonstrated a high level of performance, significantly outperforming a random policy.

Next, RLHF gathers human feedback on the set of trajectories. For the purpose of this research, the data used in the experiments was synthetic, serving as a proxy for human behaviour. Collecting real human feedback was not feasible within the given timeframe, as it would be more challenging to acquire and less controllable [32]. Moreover, our evaluation focuses on analysing performance differences between varying levels of diversity, where optimal performance is not strictly required. Previous research has shown that synthetic feedback can provide similar performance to real feedback [28], validating the use of synthetic data in our experiments.

After generating the trajectories, we obtained feedback using synthetic preferences based on the ground-truth rewards. For each fragment pair (y^1, y^2), we started by calculating the discounted sum of rewards for each fragment segment. We then sampled preferences

from a softmax distribution, obtaining the probability μ that fragment y^1 is preferred over fragment y^2.

Finally, to introduce diversity into the preference predictor, we calculated the conflicting preference μ_c as follows:

$$\mu_c = \begin{cases} \mu & \text{if } U \geq p \\ 1 - \mu & \text{otherwise} \end{cases} \tag{4}$$

Where U is a random number drawn from the uniform distribution $\mathcal{U}(0, 1)$, and p is the conflicting probability. When $p = 0$, the model reduces to the original Bradley-Terry model. As p increases, the likelihood of reversing our preference increases. At $p = 1$, the preferences are entirely reversed.

3.2 Evaluation

This research aims to analyse the behaviour of RLHF under varying degrees of conflicting feedback. Six different levels of conflicting data were tested: 0%, 25%, 40%, 50%, 75%, and 100%. The 0% level serves as a baseline with no conflict, adhering strictly to the original reward of the environment. The 25%, 40%, and 50% levels introduce moderate conflict, where feedback inconsistently aligns with the true reward. The 75% and 100% levels represent challenging scenarios, greatly deviating from the true reward. This diversity was implemented based on Equation 4, where the level determines the probability of inverting the original value.

This broad spectrum provides a comprehensive understanding of how different degrees of conflict impact performance. For instance, if RLHF can maintain performance within the first few levels in the lower-complexity environments, the higher conflict levels would allow us to analyse the extent to which RLHF can handle conflicts. Additionally, the last conflict levels serve as a benchmark to determine if there is a threshold beyond which performance bottoms out.

4 Experiments

In this section, we present a comprehensive empirical evaluation of diversity. First, we introduce the environments used in our experiments. Then, we detail our experimental framework, including how we evaluate the agents' performance. Finally, we analyse the performance in the different environments and briefly discuss the results.

4.1 Environment design

We describe our experimental setup across three environments of different complexity. The complexity of an environment can be determined based on:

- **State and action space**. The larger the state and action spaces, the more complex the models and exploration strategies need to be.
- **Dynamics complexity**. Environments with more complex dynamics (e.g., robotics arms, humanoid robots) require more sophisticated control and exploration strategies.

- **Reward structure**. Delayed or sparse rewards need advanced exploration strategies.

Based on this, we chose the following environments from the Gymnasium framework [33], in increasing complexity:

- **Pendulum**. A very simple 2D environment where the agent needs to swing a pendulum in an upright position. The observation space is small (3 dimensions), and the action space is continuous (1 action).
- **Lunar Lander**. A 2D environment where an agent needs to land a spacecraft on the moon. The observation space is relatively small (8 dimensions), and the action space is discrete (4 actions).
- **Bipedal Walker**. A more complex 2D environment where the agent needs to move a walker robot in a straight line. The observation space is bigger (24 dimensions), and the action space is continuous (4 actions).

The reward structure is dense for all chosen environments. The environment models are shown in Figure 3.

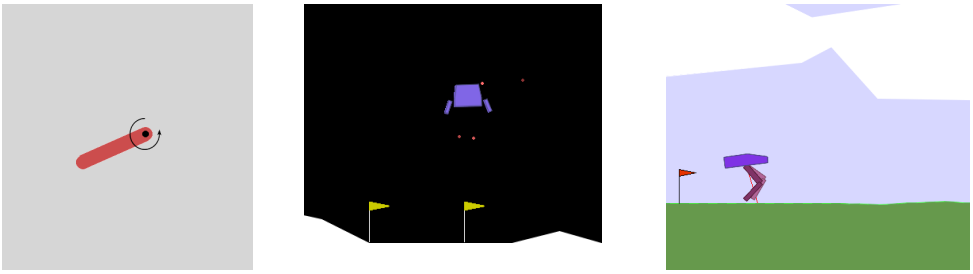


Figure 3: The environments used in the experiments. From left to right: Pendulum, Lunar Lander, and the Bipedal Walker environments.

Note that some of these environments have a variable horizon, which has confounded prior evaluation [34]. Since episode termination conditions are often correlated with reward, variable-length episodes provide a side channel of reward information that algorithms can exploit. To avoid this, we ensured the episodes had a constant length, ending after 3500 time steps.

4.2 Experimental design and configuration

The experiments were implemented using PyTorch [35] and Imitation [36]. These libraries ensure the accurate and bug-free implementation of the RLHF implementation, following the procedure described in Section 2.

Across all environments, we utilised PPO, tuning the hyperparameters for each specific environment by using the Optuna framework [37]. Some hyperparameters, such as the number of synthetic queries, had to be limited due to hardware and time constraints. Nevertheless, it resulted in satisfactory performance, slightly inferior to training with the ground-truth

rewards. The complete list of the hyperparameters can be found in Appendix A.

In our experiments, we first train the RLHF's reward models. We use an ensemble of 3 predictors, which is necessary to estimate uncertainty and apply active selection. Subsequently, we use the learned reward function to train the PPO agent.

After 1000 environment steps, the agent policies are evaluated across 10 test episodes in an independent evaluation environment. The average reward per episode is recorded. Analysing the development of the average reward allows us to assess not only the performance of the policy but also the reward model, as the PPO agent relies solely on the reward model learned by RLHF. The results are averaged across three experiments, each with different random seeds, to reduce uncertainty and the influence of any random events.

Once all agents have completed their training, we compare the learned policies using permutation tests on their mean rewards of the final 20 episodes, with a significance level of $p = 0.005$. This threshold is recommended in scientific research to discriminate significant from non-significant results [38].

A permutation test is a statistical method that allows for a confident comparison of agent performance [39]. We set the null hypothesis that both agents perform similarly, meaning no significant difference exists between their rewards. The permutation test determines if the difference is significant or could have occurred by chance, assuming the null hypothesis is true. If the actual difference is unlikely to have occurred by chance, indicated by a p-value below the significance level, the null hypothesis is rejected, and it is concluded that there is a significant difference between the agents' performances.

4.3 Results

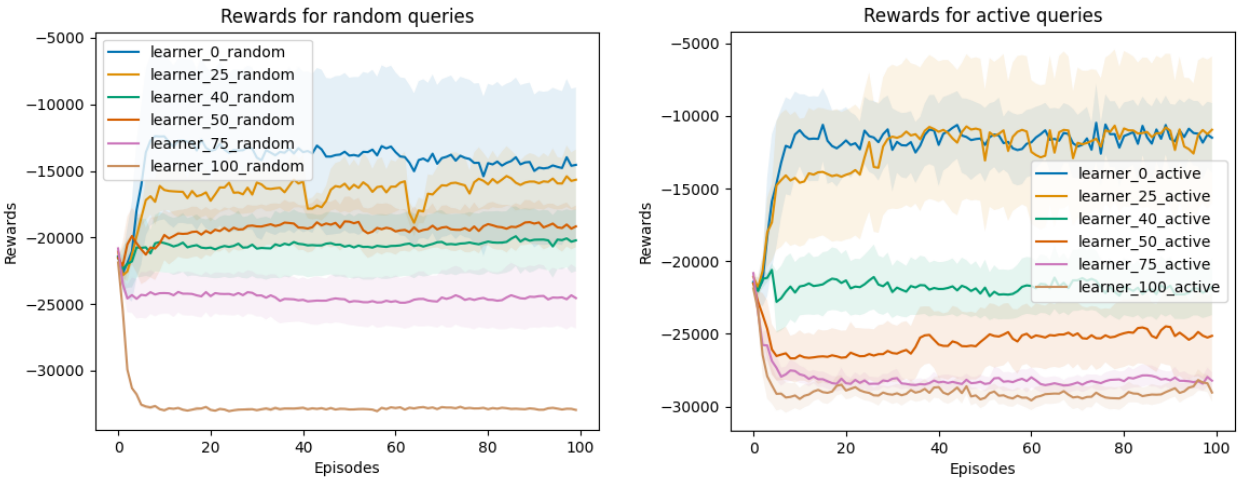
In this section, we present our empirical findings and briefly outline their significance towards answering the research question. The learners are denoted as `learner_x`, where `x` indicates the percentage of conflicting feedback, e.g., `learner_40` represents 40% conflicting feedback.

4.3.1 Pendulum

We first present the results of training RLHF in the Pendulum environment with varying levels of conflicting feedback. This environment is characterized by its low complexity. Figure 4 and Table 1 illustrate the results for both random and active query selection.

Active selection shows better results than random selection when there is no conflicting feedback, achieving a final mean reward of approximately -11500 and -14550, respectively. The standard error with random selection (the shaded blue area) is much greater than with active selection. Interestingly, `learner_25` (in orange) shows a similar performance to the no-conflict scenario (in blue) when using active selection, as confirmed by the permutation tests in Table 1b. In other words, low diversity does not significantly impact performance.

However, despite the choice of query selection strategies influencing the agents' ability to handle diversity, the results indicate that this alone is insufficient. For both strategies, performance degrades rapidly at higher levels of conflicting feedback, e.g., 40% or 50%, which are common disagreement ratios.



(a) Random query selection

(b) Active query selection

Figure 4: Results on the Pendulum environment. We compare different levels of conflicting feedback based on the mean evaluating reward over 100 episodes. 68% confidence intervals (one standard error from the mean) are shown in the shaded regions.

4.3.2 Lunar Lander

We analyse the behaviour in the Lunar Lander environment, a more complex scenario. Figure 5 presents the results for the different query selection strategies.

Similar to the Pendulum environment, active selection outperforms random selection when there is no conflicting feedback, achieving a final reward of -989 and -2210, respectively. However, as the percentage of conflicting feedback rises, the performance of both methods deteriorates. Interestingly, random querying appears to be more resilient to diversity, with a decline to -18000, compared to active's drop to -30000. Moreover, the standard deviation for active selection rapidly increases when dealing with conflicting feedback levels around 50%.

Permutation tests show that the agent with no conflicting feedback (`learner_0`) significantly outperforms the others, always yielding a p-value of 0.0001. This highlights how the environment's complexity greatly influences the decline in performance. In contrast to the Pendulum environment, a 25% probability of conflict is already sufficient to significantly impact the agent's performance, regardless of the query selection strategy.

4.3.3 Bipedal Walker

Finally, we analyse the Bipedal Walker environment, a more complex scenario. The results are shown in Figure 6. As with the Lunar Lander environment, permutation tests indicate that `learner_0` is significantly better than the other learners, consistently resulting in a p-value of 0.0001.

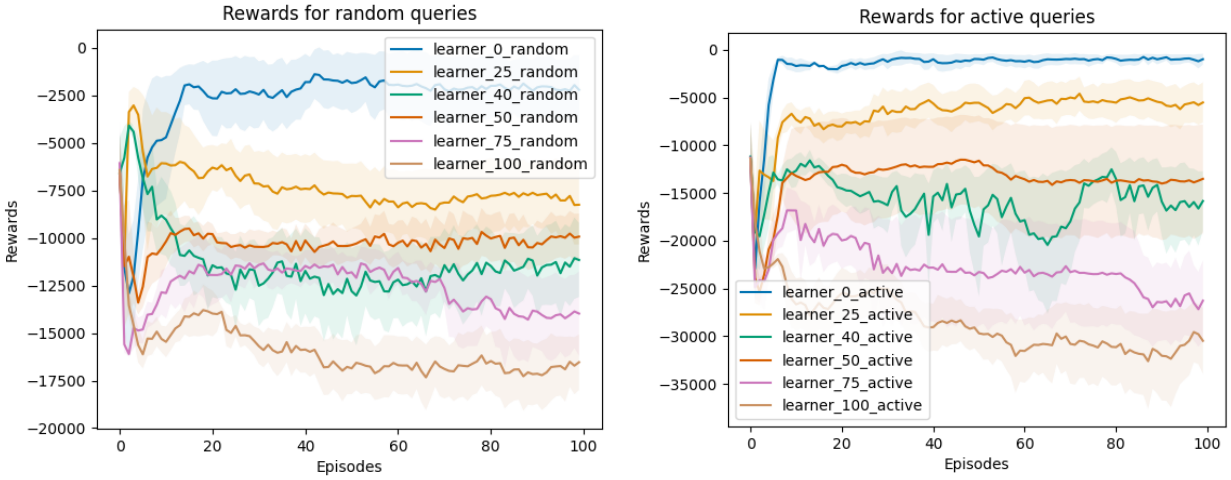
Agent 1	Agent 2	P-value
learner_0	learner_25	0.0001
learner_0	learner_40	0.0001
learner_0	learner_50	0.0001
learner_0	learner_75	0.0001
learner_0	learner_100	0.0001

(a) Random query selection

Agent 1	Agent 2	P-value
learner_0	learner_25	0.476
learner_0	learner_40	0.0001
learner_0	learner_50	0.0001
learner_0	learner_75	0.0001
learner_0	learner_100	0.0001

(b) Active query selection

Table 1: Permutation tests on the Pendulum environment. These tests are based on the last 20 reward episodes, using a significance level of 0.005. If the p-value is below this threshold, the difference in performance between the agents is significantly large. Otherwise, they have similar performance.



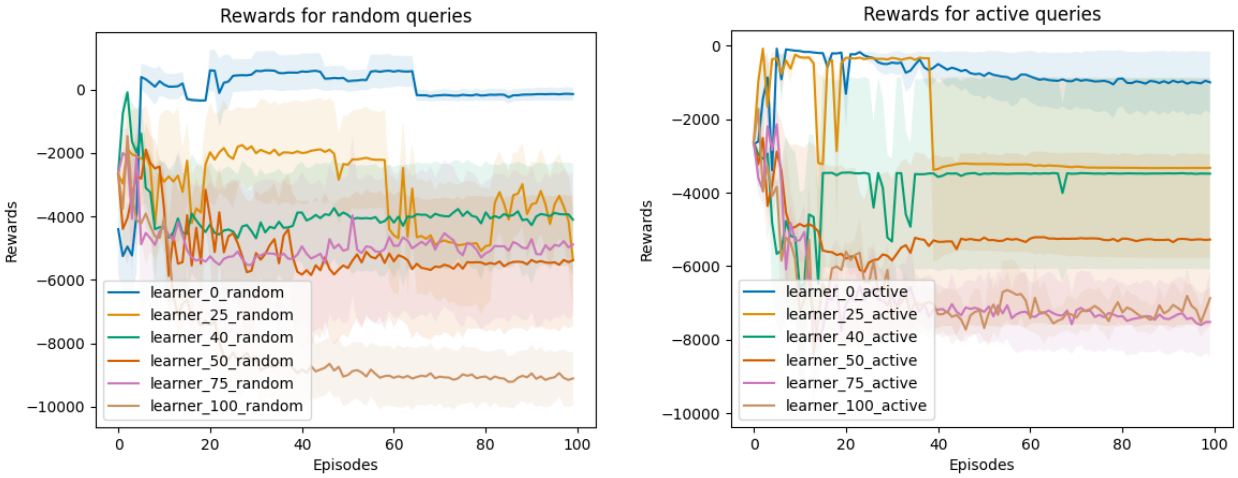
(a) Random query selection

(b) Active query selection

Figure 5: Results on the Lunar Lander environment. We compare different levels of conflicting feedback based on the mean evaluating reward over 100 episodes. 68% confidence intervals (one standard error from the mean) are shown in the shaded regions.

This environment supports our previous findings: the current RLHF algorithm is likely unable to efficiently address diversity, and regular levels of conflicts, such as 25%, can significantly impact the final outcomes. Similar to the Lunar Lander environment, random selection seems to handle diversity better than active selection, as shown by the performance of `learner_75`. However, this advantage is still insufficient to prevent a significant decline in performance. Furthermore, as in previous scenarios, the standard error for active selection increases rapidly at around 50% conflicting probability.

Note that the permutation tests' tables have been omitted for the Lunar Lander and Bipedal Walker environments due to their similarity to the Pendulum's table 1a for random query selection, as the results are consistently 0.0001. Namely, except for active selection in the Pendulum environment, `learner_0` is always significantly superior to the other agents.



(a) Random query selection

(b) Active query selection

Figure 6: Results on the Bipedal Walker environment. We compare different levels of conflicting feedback based on the mean evaluating reward over 100 episodes. 68% confidence intervals (one standard error from the mean) are shown in the shaded regions.

5 Discussion

In this study, we critically examined the limitations of conventional single-reward RLHF concerning the diversity of human feedback. Our findings suggest significant limitations, demonstrating that even a minimal amount of conflicting data can quickly degrade performance, particularly in complex tasks. The environment can also greatly influence the final result. For example, in a simple environment like Pendulum, RLHF can handle low levels of conflict without any problems. However, when we move to more complex environments such as Lunar Lander or Bipedal Walker, its performance drops significantly in the presence of diversity.

Besides, we analysed the effect of query selection techniques and confirmed their substantial impact when dealing with conflicting data. Consistent with findings from [28, 31], we observed that none of them constantly provides better results. However, we also found that active selection appears to be more effective in managing diversity when the environment's complexity is low, whereas random selection performs better in handling conflicting feedback in more complex tasks.

The confidence intervals also provided interesting insights regarding query selection strategies. When training without conflict, the standard error is significantly greater with random selection, which is expected since active selection specifically aims to reduce variance. For active selection, the standard error rapidly peaks near 50%, remaining relatively small at 0% or 100%. This pattern is logical, as diversity could alter the uncertainty of the predictions in unexpected ways. In contrast, the standard error remains mostly stable with random selection, since it does not aim to reduce uncertainty but simply selects the next query randomly.

These findings highlight the need for further research into alternative reward representations and query selection strategies. More effective management of feedback diversity is essential for improving performance.

We recognise that we did not exhaustively explore certain hyperparameters, such as the configuration of the evaluation environment and the exclusive use of PPO, and limited other parameters like training time steps, principally due to hardware and time constraints. Consequently, we cannot completely rule out the possibility of different outcomes with alternative parameter configurations. However, we took steps to ensure the consistency of our results by averaging several runs with different seeds and analysing different tasks.

Furthermore, we acknowledge the absence of high-complexity environments, such as MuJoCo [40] and Arcade Learning Environments [41], which are commonly used in RL literature [4, 6, 28]. These environments provide a more realistic setting for investigating the effects of diversity on RLHF, with a broader variety of tasks and conditions. However, they require numerous training steps to finish the task successfully. As a result, we could not incorporate them into the current study due to hardware resources and time constraints.

6 Responsible Research

In this section, we discuss the ethical aspects of our research.

Data source. A key ethical aspect of RL experiments is the provenance of data. Despite the title of this study mentioning human data, no data from humans was used in this research. We exclusively used synthetic data generated using RL algorithms in virtual environments. This approach allows for easier data control while mitigating any ethical concerns, such as those related to consent, reidentification, data manipulation, and more.

Ethical considerations. When conducting this study, we have committed to maintaining the essential principles of ethical research. RLHF algorithms, as detailed in Section 1, possess a great potential for misuse or unintended consequences, particularly in applications like LLMs, where these algorithms can be susceptible to social biases and discrimination. For example, Abid et al. [12] demonstrate how LLMs can inaccurately associate Muslims with violence. Our research is deeply connected to these issues, as they all revolve around the conflicting opinions of human experts. We aspire for the potential applications of our research to be oriented towards beneficial and ethical goals.

Plagiarism and biases. To ensure the highest standards of scientific integrity, avoiding plagiarism and conflicts of interest is crucial. We have meticulously documented all the sources used throughout this study. The research was conducted solely by the author under the guidance of supervisors. None of these parties has had any conflicting opinions or been influenced by any third party.

Reproducibility. Responsible research dictates that a study must be reproducible, valid, and must have not been manipulated. To adhere to these principles, we have provided a transparent and detailed explanation of the methodologies, including the development of the RLHF algorithms and our experimental process. Furthermore, we are committed to the

FAIR (Findable, Accessible, Interoperable, and Reusable) principles by making the code used within this study publicly accessible under an MIT licence[1]. Finally, Appendix A contains all the hyperparameter settings for replicating the experiments.

7 Conclusions and Future Work

Reinforcement Learning from Human Feedback (RLHF) is a powerful Reinforcement Learning algorithm that models the reward function from human feedback. Despite its many successes, RLHF faces significant challenges, especially when handling conflicting feedback. This study has investigated the importance and impact of diversity on the RLHF algorithm by answering the research question: *How can RLHF deal with possibly conflicting feedback coming from multiple individuals?*

Our results show that the performance of RLHF is significantly affected by even modest amounts of conflicting feedback, with degradation observed at levels as low as 25%. Even LLM labellers, trained to provide reliable feedback, diverge up to 40% of the time [8, 17]. It is only in extremely simple environments such as Pendulum where RLHF agents can barely maintain their performance. In addition, we discovered that randomly selecting queries yields better results than active selection in complex environments with high levels of feedback diversity. However, this improvement was insufficient to prevent performance degradation.

In summary, our research provides evidence of the issue that diversity supposes for the Reinforcement Learning from Human Feedback algorithm and similar methods. We hope our work stimulates further investigation into alternative reward models and query selection strategies.

Future work could investigate the performance of alternative reward modelling approaches, such as *Safe RLHF* [20], or the combination of multiple reward functions. Incorporating real human feedback, rather than relying solely on synthetic feedback, is another crucial area for exploration. Additionally, more complex scenarios, such as Large Language Models like LlaMA [42] or the MuJoCo environments [40], could be studied to better understand the effects of diversity across various applications.

Another promising direction would be to quantify the complexity of the environments and analyse the relationship between this complexity and the degradation in performance when diversity increases. Finally, experimenting with other imitation learning methods that use human feedback, such as the Direct Preference Optimization (DPO) algorithm [43], would be highly valuable.

Acknowledgements

First and foremost, I am grateful to my supervisor, Antonio Mone, and responsible professor, Dr. Luciano Cavalcante Siebert, for their guidance and support throughout the development of this thesis. Their dedication and expertise have not only made this project enjoyable but

[1] GitHub repository: https://github.com/umenzi/diversity-rlhf

have also greatly enriched the final research.

I would also like to express my sincere appreciation to my parents, Silvia Franco González and Jose Manual Páez Fernández, and brother, Daniel Páez Franco. They have been pillars of strength and support throughout my university years, and their belief in my abilities has been a constant source of inspiration.

References

[1] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *CoRR*, vol. cs.AI/9605103, 1996. [Online]. Available: https://arxiv.org/abs/cs/9605103

[2] V. Mnih, K. Kavukcuoglu, D. Silver, A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529–533, Feb. 2015. [Online]. Available: https://doi.org/10.1038/nature14236

[3] D. Silver, A. Huang, C. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. P. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, p. 484–489, Jan. 2016. [Online]. Available: https://doi.org/10.1038/nature16961

[4] T. Kaufmann, P. Weng, V. Bengs, and E. Hüllermeier, "A survey of reinforcement learning from human feedback," *arXiv (Cornell University)*, Jan. 2023. [Online]. Available: https://arxiv.org/abs/2312.14925

[5] I. Hejna, Donald Joseph and D. Sadigh, "Few-shot preference learning for human-in-the-loop rl," Mar. 2023. [Online]. Available: https://proceedings.mlr.press/v205/iii23a.html

[6] B. Ibarz, J. Leike, T. Pohlen, G. Irving, S. Legg, and D. Amodei, "Reward learning from human preferences and demonstrations in atari," *arXiv (Cornell University)*, vol. 31, p. 8011–8023, Jan. 2018. [Online]. Available: https://arxiv.org/pdf/1811.06521.pdf

[7] N. Stiennon, L. Ouyang, J. Wu, D. M. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, and P. F. Christiano, "Learning to summarize from human feedback," *CoRR*, vol. abs/2009.01325, 2020. [Online]. Available: https://arxiv.org/abs/2009.01325

[8] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," in *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., vol. 35. Curran Associates, Inc., 2022, pp. 27 730–27 744. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf

[9] OpenAI, "GPT-4 technical report," *CoRR*, vol. abs/2303.08774, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2303.08774

[10] R. Anil, S. Borgeaud, Y. Wu, J. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, D. Silver, S. Petrov, M. Johnson, I. Antonoglou, J. Schrittwieser, A. Glaese, J. Chen, E. Pitler, T. P. Lillicrap, A. Lazaridou, O. Firat, J. Molloy, M. Isard, P. R. Barham, T. Hennigan, B. Lee, F. Viola, M. Reynolds, Y. Xu, R. Doherty, E. Collins, C. Meyer, E. Rutherford, E. Moreira, K. Ayoub, M. Goel, G. Tucker, E. Piqueras, M. Krikun, I. Barr, N. Savinov, I. Danihelka, B. Roelofs, A. White, A. Andreassen, T. von Glehn, L. Yagati, M. Kazemi, L. Gonzalez, M. Khalman, J. Sygnowski, and et al., "Gemini: A family of highly capable multimodal models," *CoRR*, vol. abs/2312.11805, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2312.11805

[11] E. Perez, S. Ringer, K. Lukošiūtė, K. Nguyen, E. Chen, S. Heiner, C. Pettit, C. Olsson, S. Kundu, S. Kadavath, A. Jones, A. Chen, B. Mann, B. Israel, B. Seethor, C. McKinnon, C. Olah, D. Yan, D. Amodei, D. Amodei, D. Drain, D. Li, E. Tran-Johnson, G. Khundadze, J. Kernion, J. Landis, J. Kerr, J. Mueller, J. Hyun, J. Landau, K. Ndousse, L. Goldberg, L. Lovitt, M. Lucas, M. Sellitto, M. Zhang, N. Kingsland, N. Elhage, N. Joseph, N. Mercado, N. DasSarma, O. Rausch, R. Larson, S. McCandlish, S. Johnston, S. Kravec, S. E. Showk, T. Lanham, T. Telleen-Lawton, T. Brown, T. Henighan, T. Hume, Y. Bai, Z. Hatfield-Dodds, J. Clark, S. R. Bowman, A. Askell, R. Grosse, D. Hernandez, D. Ganguli, E. Hubinger, N. Schiefer, and J. Kaplan, "Discovering language model behaviors with model-written evaluations," Dec. 2022. [Online]. Available: https://arxiv.org/abs/2212.09251

[12] A. Abid, M. Farooqi, and J. Zou, "Large language models associate muslims with violence," *Nature Machine Intelligence*, vol. 3, no. 6, p. 461–463, Jun. 2021. [Online]. Available: https://doi.org/10.1038/s42256-021-00359-2

[13] R. Bommasani, D. A. Hudson, E. Adeli, R. B. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, E. Brynjolfsson, S. Buch, D. Card, R. Castellon, N. S. Chatterji, A. S. Chen, K. Creel, J. Q. Davis, D. Demszky, C. Donahue, M. Doumbouya, E. Durmus, S. Ermon, J. Etchemendy, K. Ethayarajh, L. Fei-Fei, C. Finn, T. Gale, L. Gillespie, K. Goel, N. D. Goodman, S. Grossman, N. Guha, T. Hashimoto, P. Henderson, J. Hewitt, D. E. Ho, J. Hong, K. Hsu, J. Huang, T. Icard, S. Jain, D. Jurafsky, P. Kalluri, S. Karamcheti, G. Keeling, F. Khani, O. Khattab, P. W. Koh, M. S. Krass, R. Krishna, R. Kuditipudi, and et al., "On the opportunities and risks of foundation models," *CoRR*, vol. abs/2108.07258, 2021. [Online]. Available: https://arxiv.org/abs/2108.07258

[14] H. Lee, S. Phatale, H. Mansoor, T. Mesnard, J. Ferret, K. Lu, C. Bishop, E. Hall, V. Carbune, A. Rastogi, and S. Prakash, "Rlaif: Scaling reinforcement learning from human feedback with ai feedback," Sep. 2023. [Online]. Available: http://arxiv.org/abs/2309.00267

[15] S. Casper, X. Davies, C. Shi, T. K. Gilbert, J. Scheurer, J. Rando, R. Freedman, T. Korbak, D. Lindner, P. Freire, T. Wang, S. Marks, C. Ségerie, M. Carroll, A. Peng, P. J. K. Christoffersen, M. Damani, S. Slocum, U. Anwar, A. Siththaranjan, M. Nadeau, E. J. Michaud, J. Pfau, D. Krasheninnikov, X. Chen, L. Langosco, P. Hase, E. Biyik,

A. D. Dragan, D. Krueger, D. Sadigh, and D. Hadfield-Menell, "Open problems and fundamental limitations of reinforcement learning from human feedback," *CoRR*, vol. abs/2307.15217, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.15217

[16] S. Chakraborty, J. Qiu, H. Yuan, A. Koppel, F. Huang, D. Manocha, A. S. Bedi, and M. Wang, "Maxmin-rlhf: Towards equitable alignment of large language models with diverse human preferences," *CoRR*, vol. abs/2402.08925, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.08925

[17] Y. Bai, A. Jones, K. Ndousse, A. Askell, A. Chen, N. DasSarma, D. Drain, S. Fort, D. Ganguli, T. Henighan, N. Joseph, S. Kadavath, J. Kernion, T. Conerly, S. El-Showk, N. Elhage, Z. Hatfield-Dodds, D. Hernandez, T. Hume, S. Johnston, S. Kravec, L. Lovitt, N. Nanda, C. Olsson, D. Amodei, T. Brown, J. Clark, S. McCandlish, C. Olah, B. Mann, and J. Kaplan, "Training a helpful and harmless assistant with reinforcement learning from human feedback," Apr. 2022. [Online]. Available: https://arxiv.org/abs/2204.05862

[18] Y. Wang, W. Zhong, L. Li, F. Mi, X. Zeng, W. Huang, L. Shang, X. Jiang, and Q. Liu, "Aligning large language models with human: A survey," *CoRR*, vol. abs/2307.12966, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.12966

[19] C. Baumler, A. Sotnikova, and H. Daumé III, "Which examples should be multiply annotated? active learning when annotators may disagree," in *Findings of the Association for Computational Linguistics: ACL 2023*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 10 352–10 371. [Online]. Available: https://aclanthology.org/2023.findings-acl.658

[20] J. Dai, X. Pan, R. Sun, J. Ji, X. Xu, M. Liu, Y. Wang, and Y. Yang, "Safe rlhf: Safe reinforcement learning from human feedback," Oct. 2023. [Online]. Available: https://arxiv.org/abs/2310.12773

[21] R. Munos, M. Valko, D. Calandriello, M. G. Azar, M. Rowland, Z. D. Guo, Y. Tang, M. Geist, T. Mesnard, A. Michi, M. Selvi, S. Girgin, N. Momchev, O. Bachem, D. J. Mankowitz, D. Precup, and B. Piot, "Nash learning from human feedback," Dec. 2023. [Online]. Available: https://arxiv.org/abs/2312.00886

[22] M. A. Bakker, M. J. Chadwick, H. Sheahan, M. H. Tessler, L. Campbell-Gillingham, J. Balaguer, N. McAleese, A. Glaese, J. Aslanides, M. M. Botvinick, and C. Summerfield, "Fine-tuning language models to find agreement among humans with diverse preferences," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/f978c8f3b5f399cae464e85f72e28503-Abstract-Conference.html

[23] J. Jang, S. Kim, B. Y. Lin, Y. Wang, J. Hessel, L. Zettlemoyer, H. Hajishirzi, Y. Choi, and P. Ammanabrolu, "Personalized soups: Personalized large language model alignment via post-hoc parameter merging," *CoRR*, vol. abs/2310.11564, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.11564

[24] G. Kovac, M. Sawayama, R. Portelas, C. Colas, P. F. Dominey, and P. Oudeyer, "Large language models as superpositions of cultural perspectives," *CoRR*, vol. abs/2307.07870, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.07870

[25] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *CoRR*, vol. abs/1707.06347, 2017. [Online]. Available: http://arxiv.org/abs/1707.06347

[26] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," *CoRR*, vol. abs/1602.01783, 2016. [Online]. Available: http://arxiv.org/abs/1602.01783

[27] J. Schulman, S. Levine, P. Moritz, M. I. Jordan, and P. Abbeel, "Trust region policy optimization," *CoRR*, vol. abs/1502.05477, 2015. [Online]. Available: http://arxiv.org/abs/1502.05477

[28] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/d5e2c0adad503c91f91df240d0cd4e49-Paper.pdf

[29] D. M. Ziegler, N. Stiennon, J. Wu, T. B. Brown, A. Radford, D. Amodei, P. F. Christiano, and G. Irving, "Fine-tuning language models from human preferences," *CoRR*, vol. abs/1909.08593, 2019. [Online]. Available: http://arxiv.org/abs/1909.08593

[30] R. A. Bradley and M. E. Terry, "Rank analysis of incomplete block designs: i. the method of paired comparisons," *Biometrika*, vol. 39, no. 3/4, p. 324, Dec. 1952. [Online]. Available: https://doi.org/10.2307/2334029

[31] A. Gleave and G. Irving, "Uncertainty estimation for language reward models," *CoRR*, vol. abs/2203.07472, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2203.07472

[32] M. Zakour, A. Mellouli, and R. G. Chaudhari, "Hoisim: Synthesizing realistic 3d human-object interaction data for human activity recognition," in *30th IEEE International Conference on Robot & Human Interactive Communication, RO-MAN 2021, Vancouver, BC, Canada, August 8-12, 2021*. IEEE, 2021, pp. 1124–1131. [Online]. Available: https://doi.org/10.1109/RO-MAN50785.2021.9515349

[33] M. Towers, J. K. Terry, A. Kwiatkowski, J. U. Balis, G. d. Cola, T. Deleu, M. Goulão, A. Kallinteris, A. KG, M. Krimmel, R. Perez-Vicente, A. Pierré, S. Schulhoff, J. J. Tai, A. T. J. Shen, and O. G. Younis, "Gymnasium," Mar. 2023. [Online]. Available: https://zenodo.org/record/8127025

[34] I. Kostrikov, K. K. Agrawal, S. Levine, and J. Tompson, "Addressing sample inefficiency and reward bias in inverse reinforcement learning," *CoRR*, vol. abs/1809.02925, 2018. [Online]. Available: http://arxiv.org/abs/1809.02925

[35] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito,

M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 8024–8035. [Online]. Available: https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

[36] A. Gleave, M. Taufeeque, J. Rocamonde, E. Jenner, S. H. Wang, S. Toyer, M. Ernestus, N. Belrose, S. Emmons, and S. Russell, "imitation: Clean imitation learning implementations," arXiv:2211.11972v1 [cs.LG], 2022. [Online]. Available: https://arxiv.org/abs/2211.11972

[37] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," *CoRR*, vol. abs/1907.10902, 2019. [Online]. Available: http://arxiv.org/abs/1907.10902

[38] D. J. Benjamin, J. O. Berger, M. Johannesson, B. A. Nosek, E.-j. Wagenmakers, R. Berk, K. A. Bollen, B. Brembs, L. Brown, C. Camerer, D. Cesarini, C. D. Chambers, M. Clyde, T. D. Cook, P. De Boeck, Z. Dienes, A. Dreber, K. Easwaran, C. Efferson, E. Fehr, F. Fidler, A. P. Field, M. Forster, E. I. George, R. Gonzalez, S. Goodman, E. Green, D. P. Green, A. G. Greenwald, J. D. Hadfield, L. V. Hedges, L. Held, T. H. Ho, H. Hoijtink, D. J. Hruschka, K. Imai, G. Imbens, J. P. A. Ioannidis, M. Jeon, J. H. Jones, M. Kirchler, D. Laibson, J. List, R. Little, A. Lupia, E. Machery, S. E. Maxwell, M. McCarthy, D. A. Moore, S. L. Morgan, M. Munafó, S. Nakagawa, B. Nyhan, T. H. Parker, L. Pericchi, M. Perugini, J. Rouder, J. Rousseau, V. Savalei, F. D. Schönbrodt, T. Sellke, B. Sinclair, D. Tingley, T. Van Zandt, S. Vazire, D. J. Watts, C. Winship, R. L. Wolpert, Y. Xie, C. Young, J. Zinman, and V. E. Johnson, "Redefine statistical significance," *Nature Human Behaviour*, vol. 2, no. 1, p. 6–10, Sep. 2017. [Online]. Available: https://doi.org/10.1038/s41562-017-0189-z

[39] A. Gleave, M. Taufeeque, J. Rocamonde, E. Jenner, S. H. Wang, S. Toyer, M. Ernestus, N. Belrose, S. Emmons, and S. Russell, "imitation: Reliably compare algorithm performance." [Online]. Available: https://imitation.readthedocs.io/en/latest/tutorials/9_compare_baselines.html

[40] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*. IEEE, 2012, pp. 5026–5033. [Online]. Available: https://doi.org/10.1109/IROS.2012.6386109

[41] M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling, "The arcade learning environment: An evaluation platform for general agents," *CoRR*, vol. abs/1207.4708, 2012. [Online]. Available: http://arxiv.org/abs/1207.4708

[42] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models," *CoRR*, vol. abs/2302.13971, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2302.13971

[43] R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, and C. Finn, "Direct preference optimization: Your language model is secretly a reward model," *CoRR*, vol. abs/2305.18290, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.18290

A Hyperparameters

We used the Optuna framework [37] to tune the hyperparameters of the algorithms. The specific hyperparameters employed for PPO can be found in Table 2, while the hyperparameters for RLHF are listed in Table 3.

It is important to note that the initial selection of some PPO hyperparameters was influenced by the documentation provided by Stable-Baselines3 and the Center for Human-Compatible AI, the creators of the Imitation library, on Hugging Face. These hyperparameters were subsequently improved using the Optuna framework.

Table 2: Hyperparameters of PPO for the environments

Hyperparameter	**Pendulum**	**Lunar Lander**	**Bipedal Walker**
batch_size	64	64	64
clip_range	0.2	0.1	0.18
ent_coef	0.01	0.01	0.0
learning_rate	1×10^{-3}	3×10^{-4}	3×10^{-4}
gae_lambda	0.95	0.9	0.95
gamma	0.91	0.999	0.999
n_envs	8	8	8
n_epochs	10	4	10
n_steps	1024	2048	2048
n_timesteps	100_000	1_000_000	2_000_000
policy	FeedForward32	MLP	MLP

Table 3: Hyperparameters of RLHF for the environments

Hyperparameter	**Pendulum**	**Lunar Lander**	**Bipedal Walker**
total_timesteps	500_000	500_000	2_000_000
total_comparisons	500	500	700
num_iterations	60	60	60
reward_trainer_epochs	3	1	4
fragment_length	100	97	100
transition_oversampling	1	1.7	1.7
initial_comparison_frac	0.1	0.32	0.32
exploration_frac	0.24	0.24	0.25
temperature	0.22	1.7	1.8
discount_factor	1	0.95	0.96

B Hardware Specifications

For all experiments, we use a machine equipped with an NVIDIA GeForce RTX 3070, an AMD Ryzen 7 5800h, and 32 GB of RAM. We could not use a supercomputer such as DelftBlue due to technical issues. One train run (training the RLHF reward model, training

the RL agent, and evaluating it) typically takes around 4 to 5 hours. We believe the biggest bottleneck is the large amount of training steps required to accurately estimate the reward model. In addition, the biggest constraint during PPO training is the evaluation, as we test our agent for 10 episodes after 10000 training steps.