Evaluation of Reinforcement Learning from Human Feedback for Mobile Manipulation

Kenneth Akers
College of Computing
Georgia Institute of Technology
Atlanta, USA
kakers@gatech.edu

Daniel Basman
College of Computing
Georgia Institute of Technology
Atlanta, GA
dbasman2@gatech.edu

Jeremy Collins
Institute for Robotics and Intelligent Machines
Georgia Institute of Technology
Atlanta, GA
jer@gatech.edu

Naveen Balaji
Institute for Robotics and Intelligent Machines
Georgia Institute of Technology
Atlanta, GA
nnagarathinam6@gatech.edu

***Abstract*—This paper explores the use of reinforcement learning from human feedback (RLHF) to train robots for a household manipulation task. RLHF enables researchers to incorporate human feedback via reward shaping, allowing for the customization and optimization of robots' behavior based on human preferences. This study utilized the Stretch RE1 robot within OpenAI Gym in a simulated cup placement task, where policies were trained with and without human feedback. Two hypotheses were tested for quantitative and qualitative evaluation, respectively, showing that policies trained with human feedback led to a lower task success rate but were rated as more similar to human movement than those trained without human feedback. Various factors of human preferences were also examined to determine what features other than accuracy could be studied to learn what the model optimizes for. The study demonstrates the potential for RLHF to better align robot behavior with user expectations, allowing non-technical users to adjust the robot's behavior as well as have robots themselves learn and understand what to prioritize within a task.**

Keywords—RLHF, human feedback, reinforcement learning, OpenAI Gym, human-robot interaction

I. Introduction

As robots become more integrated into society and collaborate with humans, developing methods that allow non-technical users to teach and customize robots' behavior is becoming increasingly important. Reinforcement learning from human feedback (RLHF) is a recently popularized method for training models to be targeted towards human preferences. Ratings garnered from humans observing a robot repeatedly attempt a task are incorporated into the training process via reward shaping. This enables researchers to tune reward signals for tasks that would otherwise be difficult to define explicitly. In this project, we explore how using human feedback can better align robot behavior with user expectations in a simple manipulation task, thereby allowing users with little technical expertise to teach and adjust the robot's behavior to their necessary tasks.

II. Related Work

RLHF allows for models to be more personalized towards specific tasks and optimized based on constraints given by human evaluators. Prior studies have shown this method to play an effective role in training large language model outputs, allowing them to finetune their answers to be more "human-like." DeepMind, for example, demonstrated how reinforcement learning can be used to increase the performance of multimodal interactive agents such as a multi-joint agent attempting a backflip [1]. Later work from Anthropic showed that iterative, online training from human preferences collected from Amazon's Mechanical Turk (MTurk) yielded language that better aligned with natural language dialog semantics [2]. Furthermore, models trained with human preferences showed strong robustness to red-teaming efforts, in which researchers purposefully input malicious prompts to elicit undesirable model responses [3]. This is especially important for public facing dialog models, where response helpfulness is a priority and model harmlessness is paramount [4], [5]. A more recent deployment of RLHF that has gained significant public attention and use is OpenAI's ChatGPT transformer-based chatbot [6]. Using an Elo ranking system, the researchers were able to finetune the language model by training a reward model on human rankings of generated text.

Aside from aligning LLMs with natural dialog expectations, RLHF is a powerful tool for capturing desired behaviors when explicit reward modeling is infeasible or ineffective. Test-to-image models often struggle to adhere to human aesthetic expectations. Human hands, for example, are notoriously difficult for these deep generative models to depict in realistic poses. Since a sufficiently large and specific dataset of hand images was infeasible to ascertain, Wu et al. collected large quantities of human preferences from online sources to successfully generate realistic limbs [7]. Similar endeavors have provided evidence that human rankings can accelerate model convergence in such tasks [8], [9]. However, it is important to note that overoptimization of human preferences when training the reward model can impede ground truth performance [10].

Human preferences are a valuable dimension in training for embodied and simulated agents as well. Deployments in video games, such as Atari, result in superhuman performance while maintaining high humanness scores [11]. In the physical domain, experiments in course navigation with intermediate task objectives showed actor-critic algorithms converged on optimal policies faster with human teachers [12]. In both contexts, reward hacking appears to persist only when single-shot, offline preferences are used. Online, human-in-the-loop feedback effectively prevents reward hacking [12], [13].

These results not only show that RLHF can be an effective tool for eliciting nuanced behavior, but also that it is an appropriate choice when specialized data is scarce and human judgements are easily obtained. While these deployments show promising results, this work is dominated by language applications. As such, in this project, we aim to apply RLHF to a simple manipulation task to investigate its potential to align robotic policies with human preferences.

III. Methods

In order to model human preferences in a data-driven manner, we train a neural network to predict a scalar reward value given a trajectory produced by a robot policy. We first collect a dataset *D* of trajectories produced by a stochastic policy in our cup placement environment. Each trajectory is modeled as a sequence of observation-action pairs.

$$D = \{\tau_1, \tau_2, \ldots, , \tau_N\}$$

$$\tau_i = \big((o_1, a_1), (o_2, a_2), \ldots, (o_k, a_k)\big)$$

Where τ is a trajectory, N is the number of trajectories, o is an observation, a is an action, and k is the number of observation-action pairs in a single trajectory. Once all trajectories are collected, they are sent to Amazon Mechanical Turk, where we ask labelers to choose, given videos of randomly paired trajectories, which policy did a better job of placing the cup on a target position. These labels augment our dataset to include human preferences, and each trajectory now has a single Boolean value indicating whether it was preferred over its pair. The full dataset with labels can be represented as:

$$D = \{(\tau_1, \gamma_1), (\tau_2, \gamma_2), \ldots, , (\tau_N, \gamma_N)\}$$

Where $\gamma_i \in \{0,1\}$, indicating whether trajectory τ_i was chosen over its random pair in Mechanical Turk. We also included an option for indicating that the trajectories were indistinguishable, and found that excluding these pairs from training improves performance, as opposed to assigning them equal preference. Once the complete dataset of trajectories and their corresponding preferences was collected, we then trained a binary classifier to predict whether a given trajectory will be preferred by a human labeler. The architecture of the preference classifier is a multi-layer perceptron followed by a sigmoid activation, and outputs a continuous scalar value $p \in (0,1)$ representing the probability that the input trajectory will be preferred.

Once we have a model that can predict the probability of being preferred by a human rater, we can then formulate reward as a performance score in a game, much like Elo in chess. Following [14], we choose the Bradley-Terry model to produce a reward given the probability of preference. The Bradley-Terry model treats the difference in reward between a pair of trajectories as an independent Bernoulli random variable, where the probability p_{ij} that $\tau_i$is preferred over τ_j is modeled as:

$$p_{ij} = \frac{e^{r_i}}{e^{r_i} + e^{r_j}} \qquad (1)$$

Solving for the log-odds corresponding to p_{ij}:

$$\log\left(\frac{p_{ij}}{1 - p_{ij}}\right) = r_i - r_j \qquad (2)$$

Notice that this model is overparameterized – if we add a constant to both r_i and r_j, all outcomes remain unchanged. Thus, we can simplify the problem by letting $r_j = 0$ for each pair. Using this trick, we can model the reward independently for each trajectory and its preference percentage, yielding our final equation for the reward:

$$r = \log\left(\frac{p}{1 - p}\right) \quad (3)$$

Where p is the output of the preference classifier, indicating the predicted percentage that a given trajectory will be preferred by a human labeler, and r is the corresponding scalar reward. We trained two reward models, one on 50 pairs of trajectories, and another on 500 pairs of trajectories.

A. Simulation details:

For our experiment, one goal is to test whether the implementation of RLHF has a statistically significant improvement on the performance of the Stretch robot. Hello Robot's Stretch RE1 is a mobile manipulator designed to perform simple domestic tasks around the home and support individuals for their assistive tasks [15]. The robot's design features a compliant tendon-actuated gripper with a 3-DoF wrist, a telescoping arm on a vertical lift, and a non-holonomic mobile base.

Assistive Gym [16] is a framework integrated into OpenAI Gym that enables the development of control algorithms for robots assisting humans in daily living tasks. This framework allows researchers to benchmark control algorithms, develop their own tasks, and compare robot designs. For our experiment we utilized the Stretch mobile environment, developed using Assistive Gym, as this would allow us to iterate on our design quicker and employ any policies that we wanted to test. We evaluated it on a simple cup placement task, where it needed to move a cup, whether by lifting, sliding, or other methods, and get it into a designated area. While this is a simple task, it can provide a lot of insight into how humans perform it, and whether there is a best way to do so. Pairs of videos were taken and uploaded to Amazon MTurk, where they were evaluated to see which was considered more “humanlike”. These preferences were then used to train our RLHF policy and compare it to the regular A2C cup placement method. To evaluate this, we came up with two hypotheses, one which evaluates the task quantitatively in terms of success, and another to evaluate human characteristics qualitatively.

B. *Reinforcement learning algorithm (SAC):*

Soft Actor-Critic (SAC) is an off-policy, model-free deep reinforcement learning algorithm based on the maximum entropy framework, addressing challenges like high sample complexity and brittle convergence in deep RL. The algorithm combines off-policy updates and a stable stochastic actor-critic formulation to achieve state-of-the-art performance on continuous control benchmarks, outperforming both on-policy and off-policy methods. SAC is highly stable and less sensitive to hyperparameters than existing methods like DDPG. By maximizing both expected reward and entropy, the maximum entropy framework improves exploration and robustness. SAC demonstrates substantial performance improvements and sample efficiency over prior on-policy and off-policy methods.

C. *Distance based reward function:*

In the proposed project, we use the Soft Actor-Critic (SAC) algorithm for a simulated robot picking an object from the given location. The reward function is designed to encourage the robot to minimize the distance between the gripper and the object, reach the target location, and release the object at the target while penalizing unnecessary steps. If the object is not on target, the reward function adds a term proportional to the gripper status (i.e., whether it's holding the object or not). The function also includes a term that minimizes the distance between the object and the target, multiplied by a weight. Finally, a constant penalty term is added to the reward function to discourage unnecessary steps. This reward function design ensures that the robot learns to perform the task effectively and efficiently while promoting exploration and robustness, as encouraged by the maximum entropy framework.

D. *Hypotheses*

1. Policies trained with human feedback will yield a higher task success rate compared to those trained without human feedback for the cup manipulation task.
2. Human observer ratings of trajectories generated with human preferences will have a higher mean rating of similarity to human movement than trajectories without human preferences implemented.

For our quantitative evaluation we utilized task success rate to test our model, performing 20 trials for each reward model as well as the control. Success was defined as the base of the cup contacting the goal by the end of a policy rollout. We used both 100 datapoints to choose preferences for, as well as 1,000 to see if more data allowed for preferences to be picked up better.

For the qualitative evaluation, we asked 23 participants to subjectively rate the "humanness" of a policy rollout trained with human feedback and a policy rollout trained on the distance-based reward function. The policy rating was a 5-point Likert scale, which allowed people to decide how close the robot's performance was to a human's replication of the event. Overall, we were able to gain a better understanding of how human preferences can be inputted into our policy and how this in turn can affect the model's evaluation of task and what it prioritizes. We find that there is a statistically significant difference between the ratings, validating our second hypothesis.

IV. Results

The preference classifiers correctly predict human preferences with 61.1% accuracy given 100 preferences and 75.5% accuracy given 1000 preferences. Using Equation 3, we can convert our preference predictions to scalar reward values and optimize the predicted trajectories to maximize this reward with soft actor-critic, a popular reinforcement learning algorithm.

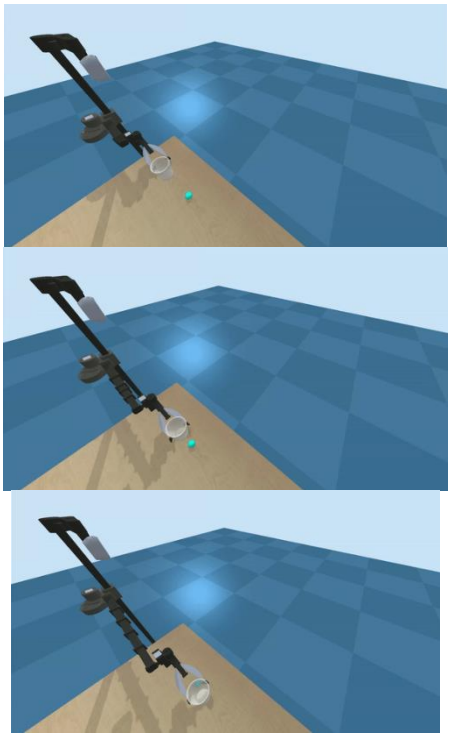

Fig. 1. Policy trained from human preferences.

Table I. Task Success Rates. p=0.00044

Method	Task Success Rate
Without human feedback	85%
With human feedback (100 examples)	30%
With human feedback (1000 examples)	25%

Through our experiments, we ultimately reject our first hypothesis. A z-test reveals that policies trained only with RLHF (see Fig. 1) achieved a significantly lower success rate than those trained by optimizing a hard-coded reward using the soft actor-critic algorithm (see Fig. 2). We speculate that this is due to the rigid definition of success that we defined before conducting our experiments. We defined success as the base of the cup coming into contact with the blue goal dot by the end of an episode. However, success in this task is much more nuanced. For example, assuming this cup is meant to

contain a liquid, we do not want to tip it over or accelerate it in such a way that would spill the liquid. We also would prefer to pick and place the cup as opposed to sliding it and manipulating the cup in such a way that it does not take a roundabout path or collide with obstacles.

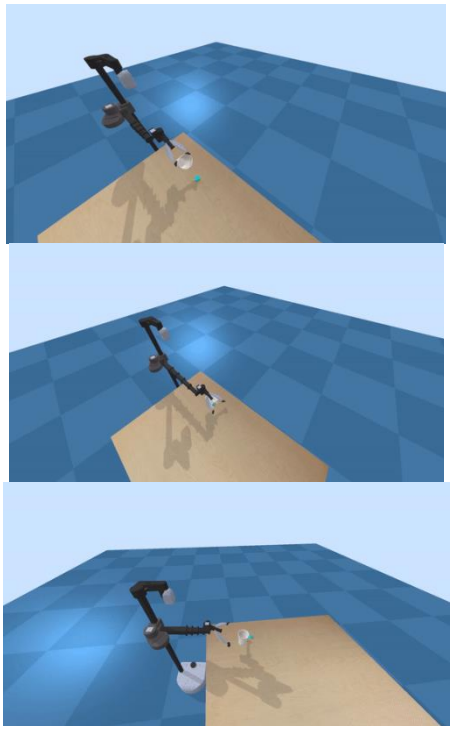

Fig. 2. Policy trained with a hand-coded reward function.

Many of these qualities that define success for an episode are difficult to define in a reward function, and as the complexity of the reward function increases, it often becomes harder to learn due to local maxima in the function that maps observations and actions to rewards. However, all these nuances in the task could plausibly be captured via human feedback, as this allows us to gain information from the complex, fine-grained understanding of the world that humans possess.

While the policy learned from human feedback achieves a lower success rate, it was qualitatively assessed to have more human-like behavior by human raters (see Table II). And indeed this was statistically significant at the 0.05 confidence level per a two-sample t test. This indicates that our success criteria were indeed imperfect, and that a more comprehensive definition of success may have produced different results. Additionally, better results may have been produced if reinforcement learning from human feedback had been used as a fine-tuning method, as is popularly done with language models, instead of using it to learn from scratch. Lastly, applying RLHF in an iterative manner, where preferences are recursively attained after training on previous preferences, may also be a viable method for improving performance.

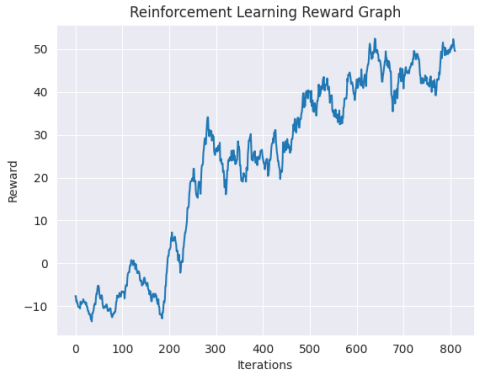

Fig. 3. Baseline reward curve.

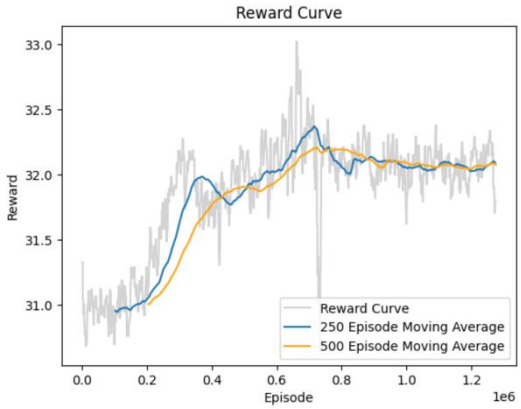

Fig. 4. 1000-sample reward curve. Note that this reward scale differs from the baseline.

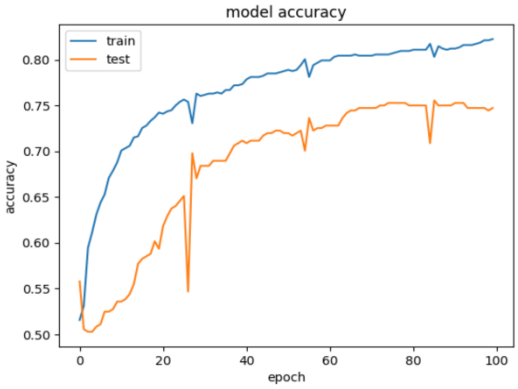

Fig. 5. Accuracy for the 1000-sample human preference classifier.

Table II. Policy humanness scores (1-5). n = 24, $p < 0.00001$

	Without Human Feedback	With Human Feedback
Sample mean	1.21	2.96
Std. dev.	0.41	0.86

References

[1] J. Abramson *et al.*, “Improving Multimodal Interactive Agents with Reinforcement Learning from Human Feedback.” arXiv, Nov. 21, 2022. doi: 10.48550/arXiv.2211.11602.

[2] Y. Bai *et al.*, “Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback.” arXiv, Apr. 12, 2022. doi: 10.48550/arXiv.2204.05862.

[3] D. Ganguli *et al.*, “Red Teaming Language Models to Reduce Harms: Methods, Scaling Behaviors, and Lessons Learned.” arXiv, Nov. 22, 2022. doi: 10.48550/arXiv.2209.07858.

[4] A. Glaese *et al.*, “Improving alignment of dialogue agents via targeted human judgements.” arXiv, Sep. 28, 2022. doi: 10.48550/arXiv.2209.14375.

[5] D. Hendrycks *et al.*, “Aligning AI With Shared Human Values.” arXiv, Jul. 24, 2021. http://arxiv.org/abs/2008.02275

[6] T. Brown *et al.*, “Language Models are Few-Shot Learners,” in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2020, pp. 1877–1901.

[7] X. Wu, K. Sun, F. Zhu, R. Zhao, and H. Li, “Better Aligning Text-to-Image Models with Human Preference.” arXiv, Mar. 25, 2023. doi: 10.48550/arXiv.2303.14420.

[8] R. Arakawa, S. Kobayashi, Y. Unno, Y. Tsuboi, and S. Maeda, “DQN-TAMER: Human-in-the-Loop Reinforcement Learning with Intractable Feedback.” arXiv, Oct. 27, 2018. doi: 10.48550/arXiv.1810.11748.

[9] B. Zhu, J. Jiao, and M. I. Jordan, “Principled Reinforcement Learning with Human Feedback from Pairwise or K-wise Comparisons.” arXiv, Mar. 19, 2023.http://arxiv.org/abs/2301.11270

[10] L. Gao, J. Schulman, and J. Hilton, “Scaling Laws for Reward Model Overoptimization.” arXiv, Oct. 19, 2022. doi: 10.48550/arXiv.2210.10760.

[11] B. Ibarz, J. Leike, T. Pohlen, G. Irving, S. Legg, and D. Amodei, “Reward learning from human preferences and demonstrations in Atari.” arXiv, Nov. 15, 2018. doi: 10.48550/arXiv.1811.06521.

[12] J. MacGlashan *et al.*, “Interactive Learning from Policy-Dependent Human Feedback,” in *Proceedings of the 34th International Conference on Machine Learning*, PMLR, Jul. 2017, pp. 2285–2294.

[13] Y. Yuan, Z. L. Yu, Z. Gu, X. Deng, and Y. Li, “A novel multi-step reinforcement learning method for solving reward hacking,” *Appl. Intell.*, vol. 49, no. 8, pp. 2874–2888, Aug. 2019, doi: 10.1007/s10489-019-01417-4.

[14] P. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei, “Deep reinforcement learning from human preferences.” arXiv, Feb. 17, 2023.http://arxiv.org/abs/1706.03741

[15] C. C. Kemp, A. Edsinger, H. M. Clever, and B. Matulevich, “The Design of Stretch: A Compact, Lightweight Mobile Manipulator for Indoor Human Environments.” arXiv, Oct. 20, 2022. doi: 10.48550/arXiv.2109.10892

[16] Erickson, Z., Gangaram, V., Kapusta, A., Liu, C. K., & Kemp, C. C. (2020, May). Assistive gym: A physics simulation framework for assistive robotics. In 2020 IEEE International Conference on Robotics and Automation (ICRA) (pp. 10169-10176). IEEE