A NEW MODEL
OF HUMAN PREFERENCES FOR LEARNING REWARD FUNCTIONS

HONORS THESIS

Stephane Hatgis-Kessell

ABSTRACT

Poor alignment between the objectives that intelligent agents optimize for and the desires of human stakeholders risks limiting the utility of decision-making systems and may pose dangers to end users. In reinforcement learning, this objective is encoded via a reward function. Manually specifying an aligned reward function is notoriously difficult. Reinforcement learning from human feedback (RLHF) is one approach to address this problem, where a reward function is instead learned from human preferences over trajectory segments. RLHF algorithms, however, require a precise model of human preferences in order to learn a reward function. What this preference model should be has remained largely unexplored, with most prior work assuming that human preferences arise solely from the sum of rewards along each segment. We find this assumption to be fundamentally flawed and instead assert that human preferences arise from a segment's deviation from optimal behavior. We propose the regret model of human preferences, showing that it is more aligned with human decision-making and leads to more performant learned reward functions. Using these findings we then reframe influential prior work in RLHF and provide a new theoretically principled perspective on past approaches. Finally, we outline an important direction for future research; developing preference elicitation interfaces to subtly guide human preferences towards a specific model and thereby improve reward learning.

Keywords Reinforcement learning from human feedback · Preference learning · Alignment

1 Introduction

Reinforcement learning (RL) is a framework where an agent learns how to behave by interacting with its environment. The agent receives a scalar reward which tells it how well it is doing, and its goal is to maximize its expected discounted sum of rewards. This learning paradigm has led to many notable achievements in robotics (Haarnoja et al. [2018], Mnih et al. [2013], Mahmood et al. [2018], Andrychowicz et al. [2020], Kalashnikov et al. [2018]). Like all optimization algorithms, however, RL agents are fundamentally limited by the objective they optimize. For RL agents this objective is encoded via the reward function, where an incorrectly specified reward function may lead to significant failures. Manually specifying the reward function is particularly error-prone, limiting the utility of RL systems. For example, a reward function that encodes the desire to "maximize dust collected off the floor" seems sensible but may result in a robot that dumps dust onto the floor in order to immediately pick it up again. When agents that optimize for unaligned objectives are placed in important decision-making roles, they may cause catastrophic consequences (Amodei et al. [2016]).

Rather than relying on humans to specify correct reward functions, we seek to learn reward functions from human data. Specifically, we focus on learning a reward function from human preferences over possible behaviors; the idea being that while it is difficult to precisely specify what good behavior is it is comparatively easier to recognize it. Once a reward function is learned from a dataset of human preferences, any RL algorithm can be used to find a policy using that reward function. This RLHF framework has shown recent promise, from training agents to master Atari video games (Christiano et al. [2017]) to fine-tuning large language models (LLMs) to sound more human-like (Ouyang et al. [2022]). We address several foundational open problems in RLHF when human feedback is given as preferences.

First, we question a ubiquitous assumption made by influential prior work; that human preferences arise probabilistically from only the sum of rewards over a segment, i.e., the segment's partial return ([Christiano et al., 2017, Sadigh et al., 2017, Ibarz et al., 2018, Bıyık et al., 2021, Lee et al., 2021a,b, Ziegler et al., 2019, Wang et al., 2022, Ouyang et al., 2022,

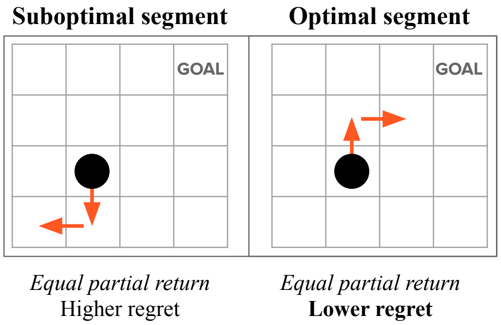


Figure 1: Here, we see two segments in a task with -1 reward each time step. The partial return preference model is indifferent between these segments because each has a partial return of -2. However, in the right segment, the agent has made better decisions; it moves toward rather than away from the goal. Our proposed regret preference model measures a segment's deviation from optimal decision-making. The right segment is therefore more likely to be preferred by a regret preference model. We suspect our human readers will also tend to prefer the right segment.

Bai et al., 2022, Glaese et al., 2022, OpenAI, 2022]). These prior works assume that human preferences are decided by the reward accrued during a segment, but ignore important start and end-state information that influences the segment's desirability. Instead, we posit that human preferences over segment pairs arise from a different statistic: each segment's deviation from optimal behavior. We measure this as the segment's regret under an optimal policy, and provide an intuitive comparison in Figure 1. We empirically show that this regret preference model is a better predictor of real human preferences than the partial return model. We then present a tractable algorithm for learning a reward function using the regret model and empirically show that this results in learning more performant reward functions from human data. We also show that when using both the regret and partial return models to learn a reward function from preferences it has generated itself, using our regret model induces more performant policies. These results indicate that the regret model of human preferences both normatively and descriptively outperforms the partial return model.

Next, we focus on the question "what happens when preferences are generated by the regret model, but are learned using the partial return model?" After all, recent work has achieved impressive results when learning with the partial return model despite our results indicating that human preferences are more aligned with the regret model. To answer this question, we show that these influential prior works may not actually be learning a reward function, but instead an approximation of the optimal advantage function. This insight reframes many RLHF algorithms and highlights a more principled approach to learning a policy from human preferences.

Finally, we focus on the prescriptive side of learning from preferences. Our empirical results indicate that human preferences are more aligned with the regret preference model than the partial return preference model. Our results also indicate that learning with preferences that perfectly follow the regret model induces better policies than when learning with preferences that perfectly follow the partial return model. Human preferences, however, do not perfectly follow the regret model. Motivated by this insight, we additionally seek to answer the following question: "Can we nudge human preferences closer to the regret model?"

This thesis seeks to provide a comprehensive overview of my work over the last three years that focuses on how to better learn aligned agent behavior from human preferences. In summary, our main contributions are listed below:

- We propose a new model for human preferences that is based on a segment's regret instead of its partial return.
- We empirically show that, when learning from a dataset of human preferences, the regret model both better predicts the human preferences and induces more performant policies.
- We empirically show that when the partial return and regret models are trained on datasets of preferences that precisely follow each model, the regret model induces more performant policies.
- Overall, we show that the choice of preference model impacts the quality of the learned reward function.
- We reframe prior work in light of our findings, showing that influential prior work may be learning an approximation of an optimal advantage function rather than a reward function.
- We highlight the implications of mistaking a learned approximate of an optimal advantage function for a learned reward function, resulting in a more principled approach to learning from human preferences.
- We introduce a concrete direction for designing preference elicitation techniques to nudge human preferences toward the regret model, yielding more performant learned policies.

My thesis is largely derived from two previous papers, Knox et al. [2022] and Knox et al. [2023a]. I reference these papers throughout this thesis, which each contains more in-depth results and analysis. My thesis aims to hit the main points, as well as present some new research currently in progress. I completed this work largely in collaboration with Dr. Brad Knox and under the supervision of Prof. Peter Stone and Prof. Scott Niekum. I am eternally grateful for their continued support and mentorship, which has shaped my academic ambitions.

2 Preliminaries: Preference models for learning reward functions

We consider the task environment to be modeled as a deterministic Markov Decision Process (MDP). This MDP is defined by the tuple (S,A,T,γ,D_0,r):

- S: the set of possible states.
- A: the set of possible actions.
- $T:S\times A\to S$: the transition function which represents the state transition given a state and action pair.
- γ: the discount factor. Unless specified otherwise, we assume undiscounted tasks (i.e., $\gamma=1$).
- D_0: the initial distribution of states.
- $r:S\times A\times S\to\mathbb{R}$: the reward function where the reward at time t depends on s_t, a_t, and s_{t+1}. Note that in this paper r always refers to the ground truth reward function, $\hat{r}$ refers to a learned approximation, and $\tilde{r}$ refers to any reward function.

A policy, denoted by $\pi:S\times A\to\mathbb{R}$, specifies the likelihood of selecting a certain action when in a particular state. The standard RL objective is to find a policy for the MDP that maximizes the expected discounted return from each state. The expected discounted return is formally given by $\sum_{t=0}^{\infty}E[\gamma^t r(s_t,a_t,s_{t+1})]$. In our problem setting, we adhere to the standard RL objective but assume no access to the ground truth reward function. Instead, we must first learn a reward function from human data and then derive a policy using that reward function, for example via RL.

We denote the state-action value function and state-value function for the reward function, $\tilde{r}$, under an optimal policy, π^*, as $Q^*_{\tilde{r}}$ and $V^*_{\tilde{r}}$ respectively.

2.1 Reward learning from pairwise preferences

We can learn a reward function from a dataset of preferences by finding a reward function that maximizes the likelihood, or equivalently minimizes the cross-entropy loss, of the observed preferences. This approach is common in recent work (Christiano et al. [2017], Ibarz et al. [2018], Wang et al. [2022], Lee et al. [2021a], Ouyang et al. [2022]).

Segments. A trajectory segment, denoted as σ, begins at state s^σ_0. Its length, represented by $|\sigma|$, is the number of transitions in the segment. The segment contains $|\sigma|+1$ states and $|\sigma|$ actions, given by $(s^\sigma_0,a^\sigma_0,s^\sigma_1,a^\sigma_1,...,s^\sigma_{|\sigma|})$. In our problem setting, segments do not contain reward information. A segment that ends in a terminal state is referred to as a trajectory. As a shorthand, we present a time-indexed function: $\sigma:\mathbb{Z}\to S\times A\times S$ with $\sigma_t=(s^\sigma_t,a^\sigma_t,s^\sigma_{t+1})$.

Preference Datasets. A preference over a pair of segments yields a sample (σ_1,σ_2,μ) in a dataset $D_\succ$. The vector μ indicates the preference over σ_1 and σ_2:

- If $\sigma_1\succ\sigma_2$, then $\mu=[1,0]$.
- If $\sigma_1\prec\sigma_2$, then $\mu=[0,1]$.
- For $\sigma_1\sim\sigma_2$ (no clear preference), $\mu=[0.5,0.5]$.

Here, μ_1 and μ_2 are the first and second elements of μ, respectively.

Learning from Pairwise Preferences. Our goal is to learn a reward function that maximizes the likelihood of the preferences in dataset $D_\succ$. To do this, prior literature frequently assumes these preferences arise from a preference model, P. The preference model outputs the probability that one trajectory segment is preferred over another given the *ground-truth* reward function, i.e., $P(\sigma_1\succ\sigma_2|r)$. In practice, r is often unobservable. Our objective is to compute $\hat{r}$, an approximation of r, by minimizing the cross-entropy loss per Equation 1.

$$loss(\hat{r},D_\succ)=-\sum_{(\sigma_1,\sigma_2,\mu)\in D_\succ}\mu_1\log P(\sigma_1\succ\sigma_2|\hat{r})+\mu_2\log P(\sigma_1\prec\sigma_2|\hat{r}) \tag{1}$$

We define $P(\sigma_1 \succ \sigma_2 | \hat{r})$ below in accordance to the the partial return preference model and our proposed regret preference model.

2.2 Partial return preference model

All discussed prior work assumes human preferences are generated by a Boltzmann distribution over a segment pair's partial returns. A segment's partial return under some reward function $\tilde{r}$ is denoted as $\Sigma_\sigma \, \tilde{r}. = \sum_{t=0}^{|\sigma|} \gamma^t \tilde{r}(\sigma_t)$

This partial return preference model is given as:

$$P_{\Sigma r}(\sigma_1 \succ \sigma_2 | \tilde{r}) = logistic\Big(\Sigma_{\sigma_1} \tilde{r} - \Sigma_{\sigma_2} \tilde{r}\Big). \tag{2}$$

In this work, we highlight the flaws of this model. We assert that a segment's partial return is not the only decider of human preferences. Going forward, $P_{\Sigma r}$ is interchangeably referred to as the partial return model as defined by Equation 2.

2.3 Regret preference model

We introduce an alternative preference model that assumes human preferences are generated by a Boltzmann distribution over a segment pair's regret. First, to provide some intuition, let's focus on segments with deterministic transitions. Consider a deterministic transition (s_t, a_t, s_{t+1}). The regret for this transition is defined as:

$$regret_{\text{d}}(\sigma_t | \tilde{r}) \triangleq V^*_{\tilde{r}}(s^\sigma_t) - [\tilde{r}_t + V^*_{\tilde{r}}(s^\sigma_{t+1})]. \tag{3}$$

Note that the subscript d in $regret_d$ emphasizes our deterministic transition assumption. For a deterministic segment with multiple transitions, regret is defined as:

$$regret_d(\sigma | \tilde{r}) \triangleq \sum_{t=0}^{|\sigma|-1} regret_d(\sigma_t | \tilde{r}) = V^*_{\tilde{r}}(s^\sigma_0) - (\Sigma_\sigma \tilde{r} + V^*_{\tilde{r}}(s^\sigma_{|\sigma|})), \tag{4}$$

$regret_d(\sigma|\tilde{r})$ measures the difference in the expected return under an optimal policy from the start state, $V^*_{\tilde{r}}(s^\sigma_0)$, and the expected return given the actions taken in the segment, $\Sigma_\sigma \tilde{r} + V^*_{\tilde{r}}(s^\sigma_{|\sigma|})$. An optimal segment, denoted as σ^*, invariably possesses a regret of 0. Conversely, a non-optimal segment, denoted as $\sigma^{\neg *}$, always has a positive regret. Hence, deterministic regret is a measure of deviation from optimal behavior under $\tilde{r}$.

When the environment contains stochastic transitions, we need to reformulate our calculation of a segment's regret in order to retain the following two properties:

- Segments containing transitions that are closer to optimal behavior have lower regret than segments containing transitions that are farther from optimal behavior.
- Optimal transitions have a regret of 0.

The regret for a segment with multiple stochastic transitions is defined as:

$$regret(\sigma | \tilde{r}) = \sum_{t=0}^{|\sigma|-1} regret(\sigma_t | \tilde{r}) = \sum_{t=0}^{|\sigma|-1} \Big[V^*_{\tilde{r}}(s^\sigma_t) - Q^*_{\tilde{r}}(s^\sigma_t, a^\sigma_t)\Big] = \sum_{t=0}^{|\sigma|-1} -A^*_{\tilde{r}}(s^\sigma_t, a^\sigma_t). \tag{5}$$

Where we refer to $A^*_{\tilde{r}}(s^\sigma_t, a^\sigma_t)$ as the optimal advantage function, i.e., the negated regret function. We refer readers to Knox et al. [2022] for a more in-depth explanation and derivation of regret. Note that, with deterministic transitions, $regret(\sigma|\tilde{r}) = regret_d(\sigma|\tilde{r})$. The regret preference model is the Boltzmann distribution over negated regret:

$$P_{regret}(\sigma_1 \succ \sigma_2 | \tilde{r}) \triangleq logistic\Big(regret(\sigma_2|\tilde{r}) - regret(\sigma_1|\tilde{r})\Big). \tag{6}$$

Intuitively, the regret preference model asserts that human preferences are not only decided by the sum of rewards in a segment but also by what happens before and after the segment. Figure 2 provides further intuition for when the regret preference model is superior the partial return preference model. In section 5.1 we present a tractable algorithm for learning with the regret preference model. Going forward, P_{regret} is interchangeably referred to as the regret model as defined by Equation 6.

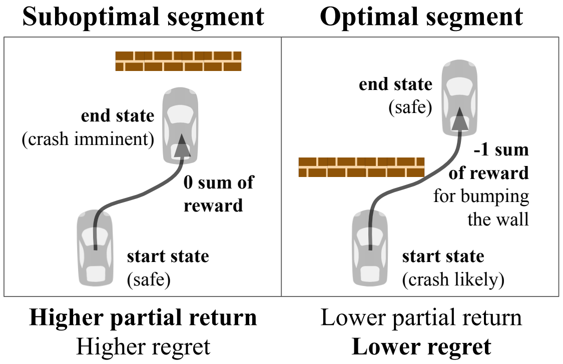


Figure 2: Two car scenarios near a brick wall: the left depicts an impending crash, and the right shows a narrow escape with a minor scrape. Under the partial return preference model, the left (suboptimal) segment with a higher sum of rewards is preferred. Conversely, the regret preference model favors the right (optimal) segment due to it having a smaller regret. We suspect our human readers will also tend to prefer the right segment.

3 Creating a human-labeled preference dataset

We seek to investigate how well our regret preference model does when learning a reward function compared to the commonly used partial return preference model. We collect a dataset of human preferences via Amazon Mechanical Turk to answer the following questions:

1. Is the regret preference model better at predicting human preferences? In other words, is the regret model more *aligned* with how humans generate preferences?

2. Does learning a reward function with the regret preference model induce more performant and human-aligned behavior?

All data collection was IRB-approved, and our collected dataset of human preferences is publically available on the Texas Data Repository Knox et al. [2023b].

3.1 The delivery domain

To investigate the consequences of learning with the regret preference model, we construct a simple grid-world-style game. The game is easy to understand and play, but recognizing *optimal* behavior is intentionally difficult for humans. Our design of the delivery domain purposefully violates the regret preference model's assumption that a human can always estimate optimal behavior. This enables us to study human preferences under more realistic conditions.

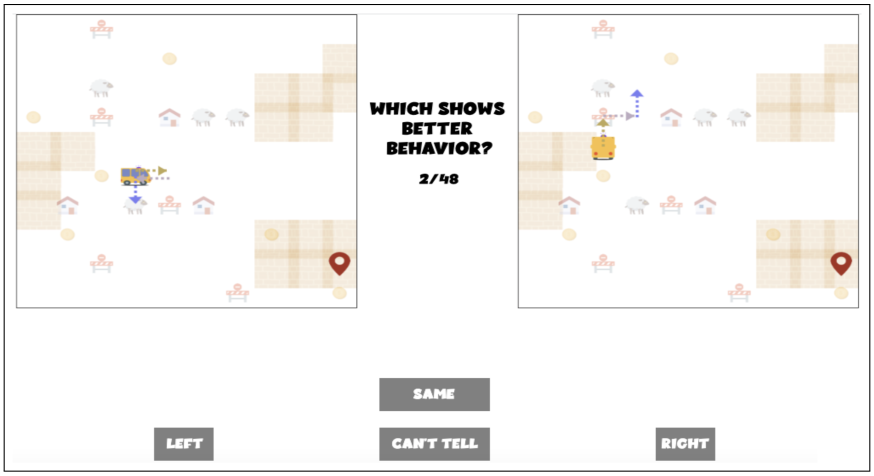
The delivery domain consists of a grid of cells. Each cell has a specific road surface type. The agent's state is its location, and its action space consists of the four cardinal directions. The episode terminates at specific termination cells, which either have a reward of $+50$ (the red marker in Figure 3) or -50 (the sheep in Figure 3). The agent always receives a -1 reward every time it moves, except when it enters a terminal state. Cells with a brick or roadblock surface type incur an additional -1 reward and cells with a coin surface type incur an additional $+1$ reward. None of these surface types disappear; for example, an agent can repeatedly collect the same coin but doing so at best cancels out the cost of moving. The initial state distribution, D_0, is always uniform over all non-termimal states.

3.2 The delivery task

We use one specific instantiation of the delivery domain when collecting human preferences. This MDP is a 10x10 grid shown in Figure 3. In the delivery task, the agent only maximizes its sum of rewards from any given state by reaching a terminal state of reward $+50$.

3.3 The preference elicitation procedure

Before eliciting human preferences, we teach subjects the dynamics and ground-truth reward function for the delivery task. In this work, we seek to recover the ground-truth reward function only from human subjects' preferences over segment pairs. To teach subjects about the MDPs reward function and dynamics, we present them with several instructions specifically describing these components as well as the general domain. We avoid technical jargon by equating "return" and "reward" to equivalent values in US dollars. Rather than telling subjects their objective is to maximize their expected


WHICH SHOWS BETTER BEHAVIOR?

2/48

Figure 4: The interface shown to subjects during preference elicitation.

return, we describe the goal of the task as maximizing their financial outcome in the game. Subjects also play a sequence of practice games in different instantiations of the delivery domain aimed at teaching them specific components of the dynamics and the reward function. A full walk-through of this procedure can be found here.

After each subject is taught to understand the delivery task, we elicit their preferences over 40–50 randomly-ordered segment pairs using the interface shown in Figure 4. The subjects select a preference, no preference (“same”), or “can’t tell”. We exclude responses labeled “can’t tell” when analyzing and learning from the resulting preference dataset.

3.4 Filtering for high quality preferences

We took several steps in order to ensure that our collected dataset of preferences was of high quality. First, we required that all subjects were located in the United States, had completed at least 100 other Mechanical Turk studies, and had an approval rating of at least 99%. This initial filtering resulted in 143 subjects who continued on to complete our study.

After teaching each of these subjects about the delivery domain and eliciting their preferences, we presented them with a comprehension quiz to assess their understanding of the task. We removed the preferences of all workers who scored below a certain threshold, which indicated that those workers may not have fully understood the task. We also removed the preferences collected from all workers who preferred ending in the negative terminal state (with a reward of -50) over not doing so. We interpreted this as a poor understanding of the task. This filtering procedure resulted in a dataset of 1,812 preferences collected from 50 subjects. For more details on our subject filtering procedure, please see Knox et al. [2022] Appendix D.

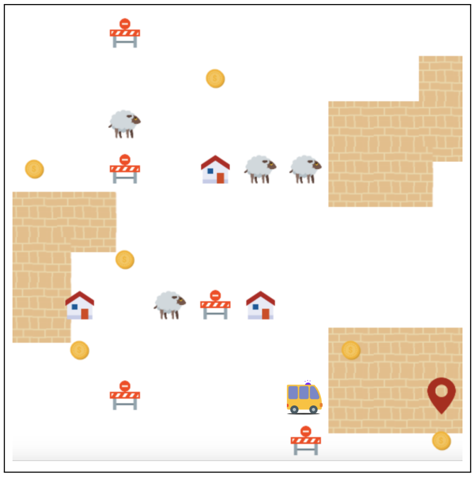


Figure 3: The delivery task used to gather human preferences. The yellow van is the agent and the red marker is the destination.

3.5 Selecting segment pairs for labeling

We collected human preferences in two stages, where we chose the segment pairs shown to subjects in each stage with distinctively different methodologies. In the first stage, we sought to elicit preferences over segment pairs that might highlight the differences between the regret and partial return preference models. We found that when only using these collected preferences, learning with the partial return preference model consistently led to poor policy performance. Note that this was not the case when learning with the regret preference model. Therefore, to aid the partial return model, we collected additional data in a second stage. More details on our data collection procedure can be found in Knox et al. [2022] Appendix D.

4 The regret model better predicts human preferences

To measure how well each model predicts human preferences, we calculate the cross-entropy loss for each model over our preference dataset with Equation 1. Note that this is equivalent to computing the negative log-likelihood of our preference dataset given a preference model. We find that the regret preference model, P_{regret}, achieves a lower mean loss across 10-folds than the partial return model, $P_{\Sigma r}$. Full results are reported in Table 4, indicating that the regret preference model is more aligned with human decision-making.

Preference model	Loss **(n=1812)**
$P(\cdot)=0.5$ (uninformed)	0.69
P_{Σ_r} (partial return)	0.62
P_{Δ_r} (regret)	**0.57**

Table 1: Mean cross-entropy losses on test sets from predicting human preferences. Lower loss is better.

5 The regret model leads to more performant policies

Per section 4, we know that the regret model is better at predicting human preferences. While this is informative, we ultimately aim to use a preference model for learning a reward function. In this section, we investigate the quality of the reward function that is learned when using either P_{regret} or $P_{\Sigma r}$. In all cases, we learn a reward function $\hat{r}$ with Equation 1. In order to find an optimal policy under $\hat{r}$, we then apply value iteration (Sutton and Barto [2018]) to compute $Q^*_{\hat{r}}$, the approximately optimal Q-function under $\hat{r}$. We derive the maximum entropy optimal policy from $Q^*_{\hat{r}}$, which chooses uniformly randomly among all optimal actions. Finally, we evaluate this policy with respect to the ground-truth reward function, r, over D_0. See Figure 5 for details. In this work, we refer to a learned reward function as "more performant" or more "human-aligned" if it induces a policy that performs better with respect to the ground-truth reward function.

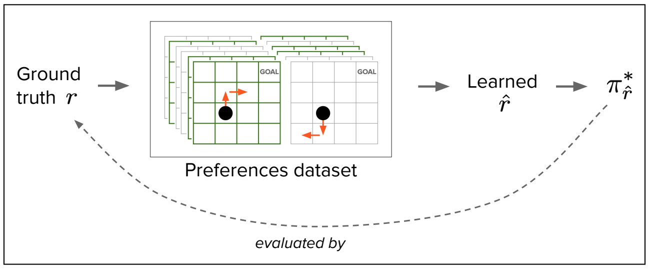


Figure 5: The general procedure used for learning a reward function from preferences and then evaluating that reward function with respect to the ground truth one. A generic gridworld shown is for illustrative purposes only.

5.1 An algorithm for learning reward functions with $regret(\sigma|\tilde{r})$

The regret preference model presented in Equation 6 requires the functions $V^*_{\tilde{r}}$ and $Q^*_{\tilde{r}}$. Therefore, when optimizing $\tilde{r}$, we need a tractable approach to approximating these functions. Below we present such an approach which applies general policy iteration (GPI) with successor features to approximate optimal state and action values for arbitrary reward functions. Our approach builds of the methodology used by Barreto et al. [2016].

Following the notation of Barreto et al., assume the ground-truth reward is linear with respect to a feature vector extracted by $\phi: S \times A \times S \rightarrow \mathbb{R}^d$ and a weight vector $w_r \in \mathbb{R}^d$: $r(s,a,s') = \phi(s,a,s')^\top w_r$. During learning, we seek to learn $\hat{r}$, an approximation for r. Therefore we have $\hat{r}(s,a,s') = \phi(s,a,s')^\top w_{\hat{r}}$ where our goal is to learn $w_{\hat{r}}$.

Our approximation of $V^*_{\tilde{r}}$ and $Q^*_{\tilde{r}}$ relies on *successor features*. Given a policy π, the successor features for (s,a) are the expectation of discounted reward features from that state-action pair when following π: ${\psi_{_Q}}^\pi(s,a) = E^\pi[\sum_{t=0}^{\infty} \gamma^t \phi(s_t, a_t, s_{t+1}) | s_0 = s, a_0 = a]$. Therefore, $Q^\pi_{\hat{r}}(s,a) = {\psi_{_Q}}^\pi(s,a)^\top w_{\hat{r}}$. Additionally, state-based successor features can be calculated from ${\psi_{_Q}}^\pi$ above as ${\psi_{_V}}^\pi(s) = \sum_{a \in A} \pi(a|s) {\psi_{_Q}}^\pi(s,a)$, making $V^\pi_{\hat{r}}(s) = {\psi_{_V}}^\pi(s)^\top w_{\hat{r}}$.

We then assume we have a set of state-action successor feature functions, $\Psi_{_Q}$, and state successor feature functions, $\Psi_{_V}$ for various policies. From [Barreto et al., 2016], we know that $Q^{\pi^*}_{\hat{r}}(s,a) \geq max_{\psi_{_Q} \in \Psi_{_Q}} [{\psi_{_Q}}^\pi(s,a)^\top w_{\hat{r}}]$ and $V^{\pi^*}_{\hat{r}}(s) \geq max_{\psi_{_V} \in \Psi_{_V}} [{\psi_{_V}}^\pi(s)^\top w_{\hat{r}}]$. We use these two maximizations as approximations of $Q^*_{\hat{r}}(s,a)$ and $V^*_{\hat{r}}(s)$, respectively.

Ψ_Q and Ψ_V are computed using an inputted set of randomly generated policies, Π. Note that, in this work, we do not allow an optimal policy for the true reward function π^*_r to be in this set: $\pi^*_r \notin \Pi$. We do this to better explore learning with the regret model under more realistic constraints.

In practice, to enable gradient-based optimization, we replace the maximization function with a softmax function that has a low temperature. For more implementation related details, including how we generate Π, please see Knox et al. [2022] Appendix F. We denote the approximation of $Q^*_{\hat{r}}$ as $\tilde{Q}^*_{\hat{r}}$ and of $V^{\pi^*}_{\hat{r}}$ as $\tilde{V}^*_{\hat{r}}$. Consequently, from Equations 5 and 6, the corresponding approximation $\tilde{P}_{regret}$ of the regret preference model is:

$$\tilde{P}_{regret}(\sigma_1 \succ \sigma_2|\hat{r}) = logistic\left(\sum_{t=0}^{|\sigma_2|\text{-}1}\left[\tilde{V}^*_{\hat{r}}(s^{\sigma_2}_t) - \tilde{Q}^*_{\hat{r}}(s^{\sigma_2}_t, a^{\sigma_2}_t)\right] - \sum_{t=0}^{|\sigma_1|\text{-}1}\left[\tilde{V}^*_{\hat{r}}(s^{\sigma_1}_t) - \tilde{Q}^*_{\hat{r}}(s^{\sigma_1}_t, a^{\sigma_1}_t)\right]\right) \quad (7)$$

We show the reward learning algorithm using $\tilde{P}_{regret}$ in Algorithm 1. Lines 3-6 outline how we generate Ψ_Q and Ψ_V given the inputted set of policies, Π. Lines 8–11 describe the supervised-learning optimization using the approximation $\tilde{P}_{regret}$. Further details on our instantiation of Algorithm 1 for the delivery domain can be found in Knox et al. [2022] 6.1 and Appendix F.1.

Algorithm 1 Linear reward learning with regret preference model (P_{regret}), using successor features

```
 1: Input: a set of policies, $\Pi$
 2: $\Psi \leftarrow \varnothing$
 3: for each reward function policy $\pi_{SF}$ in the input set do
 4:     estimate $\psi^{\pi_{SF}}_Q$ and $\psi^{\pi_{SF}}_V$ (if not estimated already during step 4)
 5:     add $\psi^{\pi_{SF}}_Q$ to $\Psi_Q$
 6:     add $\psi^{\pi_{SF}}_V$ to $\Psi_V$
 7: end for
 8: repeat
 9:     optimize $w_{\hat{r}}$ by loss of Eqn. 1, calculating $\tilde{P}_{regret}(\sigma_1 \succ \sigma_2|\hat{r})$ via Eqn. 7, using $\Psi_Q$ and $\Psi_V$
10: until stopping criteria are met
11: return $w_{\hat{r}}$
```

5.2 Learning from human data

We consider reward learning using the regret and partial return preference models with preferences generated by humans in our delivery task. Results are summarized in Figure 6. We uniformly split our collected dataset of 1,812 preferences into partitions of equal size, resulting in various training datasets. We find that with smaller preference datasets, learning with $\tilde{P}_{regret}$ induces near-optimal performance more often. With larger preference datasets, both $\tilde{P}_{regret}$ and $P_{\Sigma r}$ always induce near-optimal performance. We apply a Wilcoxon paired signed-rank test on normalized mean return to each group with 5 or more partitions. We find that, $p < 0.05$ for all numbers of partitions except 100 and $p < 0.01$ for 20 and 50 partitions.

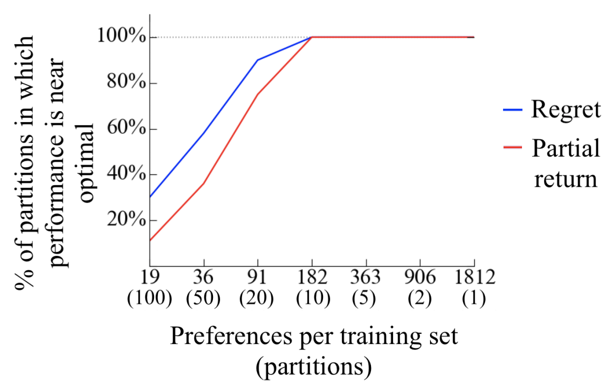


Figure 6: Performance comparison over various amounts of human preferences. Each partition has the number of preferences shown or one less.

With a sufficiently large dataset of preferences, learning with either the regret or partial return preference model always induces near-optimal performance. With smaller datasets of preferences, learning with the regret preference model outperforms learning with the partial return preference model. For more detailed results, including on learning with different types of segment pairs and variations of the regret preference model, please see Knox et al. [2022] Appendix F.3.

5.3 Learning from synthetic data

We consider reward learning using the regret or partial return preference model with synthetic preferences generated by each model. In other words, we investigate how well each preference model does when learning from preferences that perfectly adhere to it. The ground-truth reward function r is used to create these preference datasets, although while learning a reward function $\hat{r}$ we have no access r.

For these results, we use either a stochastic or noiseless instantiation of the partial return or regret preference model to generate synthetic preferences. For the stochastic instantiation, preferences are sampled from the distribution created by Equation 2 for partial return-based preferences or Equation 6 for regret-based preferences. For the noiseless instantiation, preferences arise from a direct comparison of a segment pair's partial return, $\Sigma_\sigma r$, or regret, $regret(\sigma|r)$.

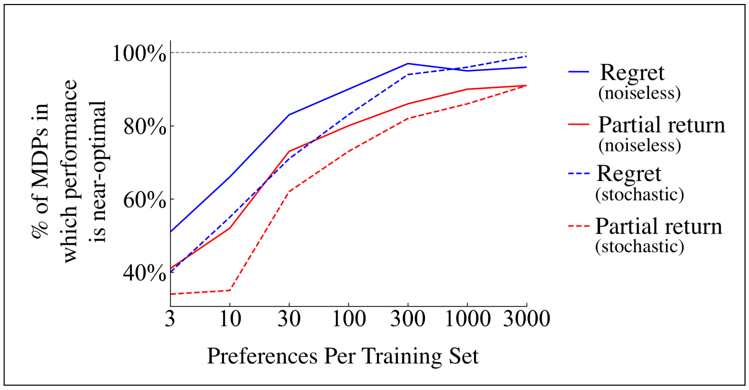


Figure 7: Performance comparison over 100 randomly generated MDPs with deterministic transitions. Each preference model creates its own training dataset and learns from it. Learning with the regret preference model is consistently induces better performance, regardless of training set size or whether preferences are generated stochastically.

We randomly generate 100 MDPs, each of which is a different instantiation of the delivery domain with a different size, reward function, and layout. For the specific parameters used to generate this set of MDPs, please see Knox et al. [2022] Appendix F.2. For each of the 100 MDPs, we randomly generate preference datasets of different sizes for training. The segments in each dataset are constructed by uniformly sampling start states and three subsequent actions. For a set number of preferences, each method has the same dataset of segment pairs. Figure 7 shows the percentage of MDPs in which each preference model results in near-optimal performance. At every dataset size and with stochastic or noiseless preferences, learning with the regret model always outperforms learning with the partial return model. By a Wilcoxon paired signed-rank test on normalized mean returns, $p < 0.05$ for 86% of these comparisons and $p < 0.01$ for 57% of them. These results indicate that, when preferences are generated perfectly in accordance with either preference model, learning with the regret preference model and regret-based preferences is superior.

For more detailed results, including on learning with segments of different lengths, with segments containing stochastic transitions, without segments that terminate before their final transition, and with additional novel preference models, see Knox et al. [2022] Appendix F.2.

6 Summary: the regret preference model

To summarize, we propose the regret preference model as a strong contender to replace the ubiquitous partial return preference model. We design a human subjects experiment and show that regret is a better predictor of real human preferences than partial return. We also show that, when learning a reward function from human or synthetic preferences, using the regret model induces more performant and human-aligned behavior. These results serve as evidence that the choice of preference model impacts reward learning and that using our proposed regret model poses significant advantages to the previously used partial return model.

6.1 Limitations

The regret model has its own set of limitations. It assumes that humans can distinguish between optimal and near-optimal behavior and, like most prior work, that their preferences follow a Boltzmann distribution. We generally assume that humans use a discount factor of 1, and that this should be used by both the RL and reward learning algorithm; this assumption remains uninvestigated. For more discussion on the impact of the discount factor on policy and reward learning, see Knox et al. [2022] Appendix B.2. We also do not consider which segment pairs should be presented to humans for labeling when learning a reward function. However, other research has addressed this problem through active learning ([Lee et al., 2021a, Christiano et al., 2017, Akrour et al., 2011]).

Our instantiation of a regret-based reward learning algorithm assumes that the ground-truth reward function can be expressed as a linear combination of weights and features. This assumption may not hold for all tasks. Additionally, generating candidate successor features for the approximations $\tilde{Q}^*_{\hat{r}}$ and $\tilde{V}^*_{\hat{r}}$ in more complex domains may pose additional challenges. In this work, learning with P_{regret} is more sample efficient by slower than when learning with $P_{\Sigma r}$. Recent work from Hejna et al. (Hejna and Sadigh [2023]) proposes a new preference learning algorithm that uses our regret interpretation of human preferences and overcomes these computational limitations in complex robotics tasks.

7 Learning optimal advantage from preferences and using it as reward

In Section 4 we show that the regret preference model better describes real human preferences, and in Section 5 we show that learning with the regret preference model leads to more performant behavior. However notable recent work has achieved remarkable results when learning with the partial return preference model, which we view as fundamentally flawed and unaligned with human decision-making. Such notable work includes fine-tuning LLMs using human preferences to sound more human-like and produce more helpful responses (OpenAI [2022], Ouyang et al. [2022], Bai et al. [2022], Touvron et al. [2023]). Towards a better understanding of these past approaches, we investigate the consequences of assuming preferences are based upon partial return when they actually arise from regret.

We argue that, when learning with $P_{\Sigma r}$ from preferences generated by P_{regret}, the learned function is an approximation of the optimal advantage function, $\widehat{A}^*_r$, *not* a reward function. Under our interpretation, prior RLHF approaches that use $P_{\Sigma r}$ and assume the learned function is a reward function may still achieve strong performance if a certain pitfall is addressed. Nonetheless, we provide empirical evidence that this incorrect usage of $\widehat{A}^*_r$ is less desirable than the appropriate and simpler approach of greedily maximizing $\widehat{A}^*_r$. Broadly, we present novel insights on why learning with the partial return preference model tends to work so well in practice, despite it conforming poorly to how humans give preferences.

7.1 Learning an optimal advantage function

We have two preference models: Our proposed regret preference model, P_{regret} and the partial return preference model, $P_{\Sigma r}$. Let us unify these two preference models under a single framework:

$$P_g(\sigma_1 \succ \sigma_2|\tilde{r}) \triangleq logistic\Big(\sum_{t=0}^{|\sigma_1|-1} g(\sigma_{1,t}) - \sum_{t=0}^{|\sigma_2|-1} g(\sigma_{2,t})\Big) \tag{8}$$

In the unification above, the segment statistic used by the preference model is represented as the sum of a certain function, denoted as g, which is applied to each transition in the segment. The overall statistic for a segment is calculated by aggregating the outcomes of applying function g to every transition that in that segment: $\sum_{t=0}^{|\sigma|-1} g(\sigma_t) = \sum_{t=0}^{|\sigma|-1} g(s^\sigma_t, a^\sigma_t, s^\sigma_{t+1})$. For the partial return preference model, $g(\sigma_t) = \tilde{r}(s^\sigma_t, a^\sigma_t, s^\sigma_{t+1})$, and the reward function $\tilde{r}$ is learned via Equation 1.

Recall from Equation 5 that the regret function is equivalent to the negation of the optimal advantage function, $-A^*_{\tilde{r}}(s^\sigma_t, a^\sigma_t)$. For the regret preference model, $g(\sigma_t) = A^*_{\tilde{r}}(\sigma_t) = A^*_{\tilde{r}}(s^\sigma_t, a^\sigma_t)$ and the parameters of this optimal advantage function can be learned directly, also via Equation 1.

In this work, we argue that when preferences are assumed to arise from partial return but actually arise from regret, an approximation for the optimal advantage function, $\widehat{A}^*_r$, is learned rather than a reward function. Once learned, we can derive a policy from $\widehat{A}^*_r$ by acting greedily upon it: $argmax_a \widehat{A}^*_r(s,a)$. We call this algorithm $greedy\ \widehat{A}^*_r$. In contrast, prior work mistakes the learned function for a reward function and therefore must perform RL to derive a policy. Notably, $greedy\ \widehat{A}^*_r$ bypasses the need for a separate policy improvement phase and directly utilizes $\widehat{A}^*_r$. No reward function is explicitly learned, though we still assume that preferences were generated by regret under a hidden reward function r.

The rest of this section endeavors to explain why the partial return preference model is highly effective in practical applications, even though it doesn't accurately reflect how humans generate preferences. We investigate the consequences of first using the error-free A^*_r as a reward function ($r_{A^*_r} = A^*_r$), and then of using the approximation $\widehat{A}^*_r$ as a reward function ($r_{\widehat{A}^*_r} = \widehat{A}^*_r$). We refer to the mistaken approach in the first setting as $greedy\ Q^*_{r_{A^*_r}}$ and in the second as $greedy\ Q^*_{r_{\widehat{A}^*_r}}$.

7.2 Using A^*_r as a reward function

Assume preferences are generated by regret. Learning a function from these preferences using the partial return model and treating it as a reward function effectively results in a reward function that is an approximation of the optimal advantage function: $\hat{r} = \widehat{A}^*_r$. Here, we consider what happens when this approximation is perfect: $\widehat{A}^*_r = A^*_r$. In this ideal scenario, $r_{A^*_r}$ is the reward function that emerges when treating A^*_r as a reward function. For a given MDP, we first note that the set of optimal policies with respect to $r_{A^*_r}$ is equivalent to the set of optimal policies that arise when greedily maximizing A^*_r:

In Thereom 7.1 we state that greedy action selection is optimal for an arbitrary reward function, $\tilde{r}$, if the maximum reward in every state is 0. We denote the set of optimal policies with respect to $\tilde{r}$ as $\Pi^*_{\tilde{r}}$.

Theorem 7.1 (Greedy action is optimal when the maximum reward in every state is 0.)**.**
$\Pi^*_{\tilde{r}} = \{\pi : \forall s, \forall a \; [\pi(a|s) > 0 \Leftrightarrow a \in argmax_a \tilde{r}(s,a)]\}$ *if* $max_a \tilde{r}(\cdot,a) = 0$.

For a proof of this thereom, please see Knox et al. [2023a] Appendix A. The underlying intuition is that, if the maximum reward in every state is 0, then the maximum return from any state is also 0. Therefore, greedily selecting the action with the highest reward of 0 is optimal.

By definition of an optimal advantage function, $max_a A^*_{\tilde{r}}(\cdot,a) = 0$. It follows that using A^*_r as a reward function induces the same set of optimal policies as when greedily maximizing A^*_r. From the definition of an optimal advantage function, it also follows that greedily maximizing A^*_r results in the same set of optimal policies that arise from the ground truth reward function, r. Therefore, treating A^*_r as a reward function is principled:

Corollary 7.1 (Policy invariance of $r_{A^*_r}$)**.**
Let $r_{A^*_r} \triangleq A^*_r$. *If* $max_a A^*_r(\cdot,a) = 0$, $\Pi^*_{r_{A^*_r}} = \Pi^*_r$.

A more detailed proof can be found in Knox et al. [2023a] Appendix B. So $r_{A^*_r}$ and r induce the same set of optimal policies, but what is the difference between using these two reward functions when actually learning a policy? Importantly, $r_{A^*_r}$ is a highly shaped reward function and is equivalent to the reward function that results from using the potential-based shaping method proposed by Ng et al. [1999]. Learning using a shaped reward function can be advantageous, as it may reduce the number of environment samples needed compared to learning with the original reward function. Nonetheless, acting greedily over A^*_r rather than treating it as a reward function saves computation and induces the same set of optimal policies.

7.3 Using $\widehat{A}^*_r$ as a reward function

In practice, when learning from a finite dataset of preferences, it is unlikely that any approach will recover the ground-truth optimal advantage function. Therefore, here we consider the more realistic scenario where $\widehat{A}^*_r$ is learned with some error and $\hat{r} = \widehat{A}^*_r$.

Empirically, we find that this error only induces a difference in performance between $greedy\ Q^*_{r_{\widehat{A}^*_r}}$ and $greedy\ \widehat{A}^*_r$ when $max_a \widehat{A}^*_r(s,a) \neq 0$ in at least one state s. To test the assertion, we adjust $\widehat{A}^*_r$ to have the property $max_a \widehat{A}^*_r(\cdot,a) = 0$ by shifting $\widehat{A}^*_r$ by a state-dependent constant: for all (s,a), $r_{\widehat{A}^*_r\text{-}shifted}(s,a) \triangleq \widehat{A}^*_r(s,a) - max_{a'} \widehat{A}^*_r(s,a')$. Note that $argmax_a r_{\widehat{A}^*_r\text{-}shifted}(s,a) = argmax_a \widehat{A}^*_r(s,a)$. In 90 randomly generated MDPs from our delivery domain (detailed in Section 3.1), we observe no performance difference between $greedy\ \widehat{A}^*_r$ and $greedy\ Q^*_{r_{\widehat{A}^*_r}}$ when $\widehat{A}^*_r = r_{\widehat{A}^*_r\text{-}shifted}$. Despite no difference in performance, $greedy\ Q^*_{r_{\widehat{A}^*_r}}$ incurs additional computational cost while policy improvement is executed and environment samples are collected. $greedy\ \widehat{A}^*_r$ does not incur these additional costs. Details on how these 90 MDPs were generated can be found in Knox et al. [2023a] Appendix D.1, and graphed results are in Appendix E.

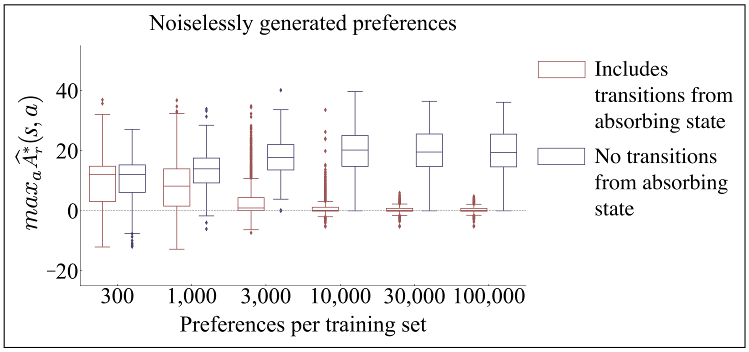


Figure 8: For each state in a set of 30 MDPs, the plots above show the $max_a \widehat{A}^*_r(s,a)$ values when including versus not including segments that contain transitions from the absorbing state. Wilcoxon paired signed-rank tests conducted at each training set size consistently yield highly significant results, with a p-value of less than 10^{-7} in all cases.

Importantly, $max_a \widehat{A}^*_r(\cdot,a) = 0$ is not guaranteed when learning an approximation of A^*_r. We find that including segments in the training dataset that contain transitions from the absorbing state moves $max_a \widehat{A}^*_r(\cdot,a)$ closer to 0 and therefore improves the performance of $greedy\ Q^*_{r_{\widehat{A}^*_r}}$. Intuitively, transitions from the absorbing state are hardcoded to have a reward and advantage of 0 so they provide an "anchor point" for all other preference comparisons. Figure 8 highlights this relationship, where including transitions from the absorbing state results in $max_a \widehat{A}^*_r(\cdot,a)$ being closer to 0.

When $max_a \widehat{A}^*_r(s,a)$ tends to be near 0, we find the performances of $greedy\ Q^*_{r_{\widehat{A}^*_r}}$ and $greedy\ \widehat{A}^*_r$ to be similar, although, $greedy\ \widehat{A}^*_r$ still tends to outperform $greedy\ Q^*_{r_{\widehat{A}^*_r}}$. When the training dataset does not contain segments

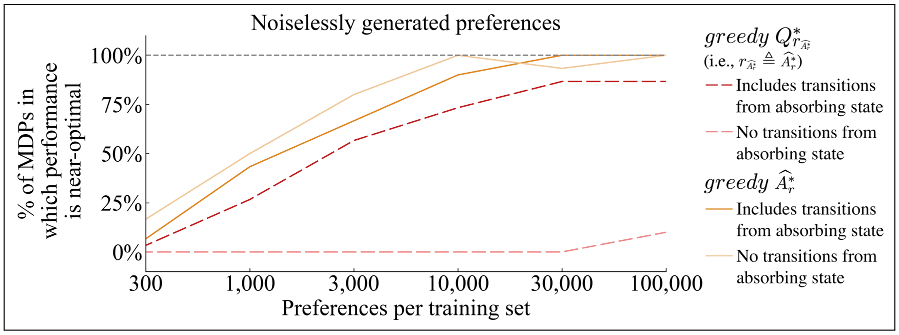


Figure 9: For each state in a set of 30 MDPs, the plots above show the performance when noiselessly generated preference datasets do and do not include segments with transitions from absorbing state. For $greedy\ \widehat{A}^*_r$ (in red) Wilcoxon paired signed-rank tests reveal that including transitions from absorbing state results in significantly higher performance for all training set sizes but the smallest, 300, with $p < 0.0007$. No significant difference in performance is detected for $greedy\ Q^*_{r_{\widehat{A^*_r}}}$ with or without terminating transitions except at 30,000 preferences with a more modest $p = 0.04$.

with transitions from the absorbing state, and therefore $max_a \widehat{A}^*_r(s,a)$ is often not close to 0, $greedy\ Q^*_{r_{\widehat{A^*_r}}}$ significantly underperforms $greedy\ \widehat{A}^*_r$. These results are shown in Figure 9 for 90 randomly generated MDPs with noiseless, synthetic preferences generated by the regret model. These results empirically support our assertion that, if preferences are generated by regret and are learned from using the partial return model, the learned function should be treated as an optimal advantage function not a reward function. For results when learning from stochastic synthetic preferences, please see Knox et al. [2023a] Appendix G. In Knox et al. [2023a] Section 3.3 we present a hypothesis aimed at explaining the circumstances under which $greedy\ \widehat{A}^*_r$ outperforms $greedy\ Q^*_{r_{\widehat{A^*_r}}}$ when the maximum value of $\widehat{A}^*_r(s,a)$ tends to be close to zero.

In Section 7.2, we assert that using A^*_r as a reward function results in a highly shaped reward function. We test whether this reward shaping, which may be beneficial, is also present when using the approximation $\widehat{A}^*_r$ as a reward function. Figure 10 shows that policy improvement with the Q learning algorithm (Watkins and Dayan [1992]) is more sample efficient with $r_{A^*_r}$ and with $r_{\widehat{A^*_r}}$ than with the ground truth r, as was expected. To further quantify the difference between using these reward functions, we define AAC as the area above a learning curve and below 1.0. A small AAC indicates better learning performance. For the results plotted in 10, Wilcoxon paired signed-rank tests reveal that Q learning with r (purple) has a larger AAC than with $r_{A^*_r}$ (red), which in turn has a larger AAC than with $r_{\widehat{A^*_r}}$ (both $p < 0.00003$).

To summarize, when $\widehat{A}^*_r$ is learned with some error, we find that it is always better to greedily maximize $\widehat{A}^*_r$ rather than treat it as a reward function. We hypothesize that influential prior work follows the latter approach, while we advocate for the former. Greedily maximizing $\widehat{A}^*_r$ is conceptually simpler, i.e., by avoiding the need to perform RL, and results in better performance. The performance gap is particularly pronounced when $max_a \widehat{A}^*_r(\cdot,a)$ is not close to 0 for at least one state. This pitfall that arises when $\hat{r} = \widehat{A}^*_r$ can be mitigated by including segment pairs in the training dataset that contain transitions from the absorbing state or by manually forcing $max_a \widehat{A}^*_r(\cdot, a) = 0$. Nonetheless, greedily maximizing $\widehat{A}^*_r$ is more principled and performant.

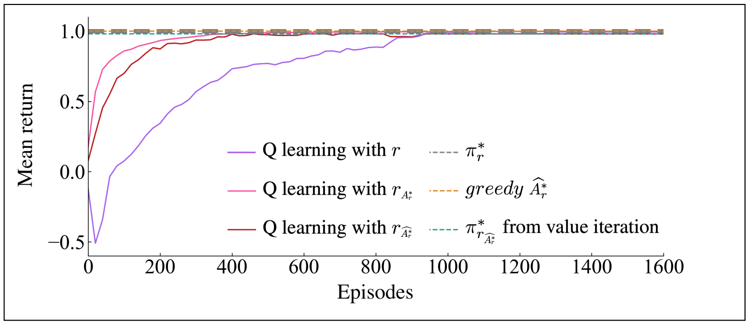


Figure 10: For a set of 100 MDPs, the plots above show the learning curves for Q learning on the ground truth reward function r and on $r_{\widehat{A^*_r}}$. In each MDP, $\widehat{A}^*_r$ was learned with 100,000 noiseless, regret-based preferences. We see that learning is more efficient, indicating that in practice $r_{\widehat{A^*_r}}$ is a helpfully shaped reward function, as is using the true A^*_r as a reward function.

8 Reframing prior work

The partial return preference model has been used in several high-profile applications:

- To fine-tune large language models for text summarization Ziegler et al. [2019].

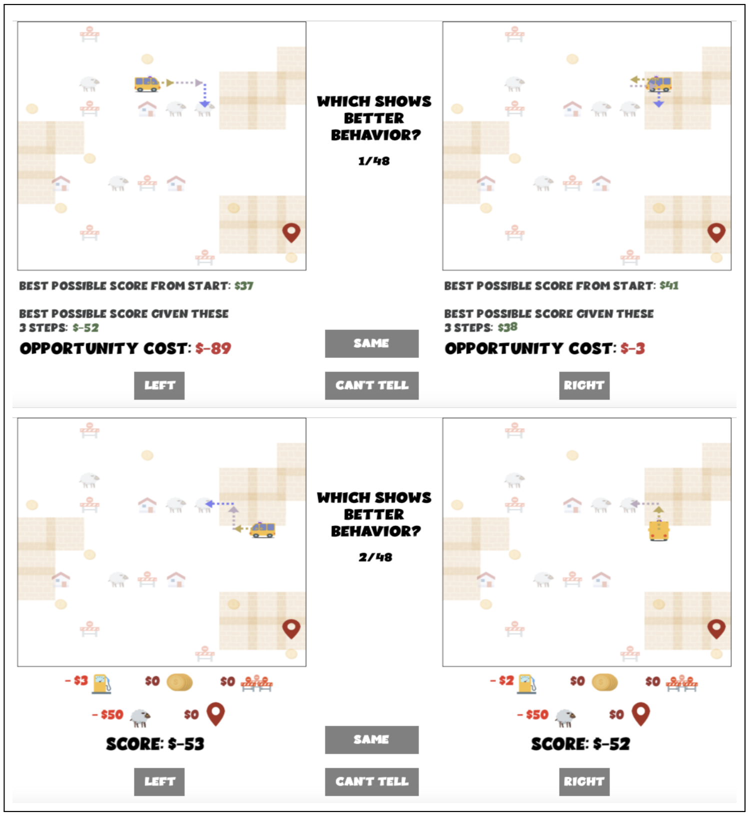
- To create InstructGPT and ChatGPT Ouyang et al. [2022], OpenAI [2022], as well as to fine-tune Llama 2 Touvron et al. [2023].
- To achieve strong performance on a suite of Atari-video games Christiano et al. [2017] and simulated robotics tasks Lee et al. [2021b], Lee et al. [2021a].

The use of the partial return model in these works fortuitously allows **an alternative interpretation of their approach: they are learning an optimal advantage function from regret-based preferences, not a reward function.** This reframing suggests that, instead of treating the learned function as a reward function and performing RL to derive a policy (such as with $greedy\ Q^*_{r_{\hat{A}^*}}$ in Section 7), it should instead be treated as an optimal advantage function and greedily maximized. The consequence of our interpretation yields a family of conceptually simpler and less computationally burdensome approaches to learning from preferences. Our perspective was recently explored by Hejna et al. [2023], resulting in strong performance outside of our grid world domain. For a more detailed discussion on how our interpretation provides unique insights on common approaches to fine-tuning large language models, please see Knox et al. [2023a] Section 4.

8.1 Limitations

Our exploration into learning optimal advantage from preferences and mistaking it for reward has a few limitations. Firstly, we assume that the human preferences used for training by the aforementioned prior works follow the regret preference model. While our empirical results from Section 4 support this assumption, we do not have access to the preference datasets used by these other works. Secondly, our interpretation asserts that a policy should be derived by greedily maximizing the learned optimal advantage function rather than by treating it as a reward function. We do not consider how this can be practically implemented in domains with large or infinite action spaces, although Hejna et al. [2023] takes a first step towards applying our insights to domains with continuous actions.

9 Nudging human preferences toward the regret model



WHICH SHOWS BETTER BEHAVIOR?

1/48

BEST POSSIBLE SCORE FROM START: $37

BEST POSSIBLE SCORE GIVEN THESE 3 STEPS: $-52

OPPORTUNITY COST: $-89

BEST POSSIBLE SCORE FROM START: $41

BEST POSSIBLE SCORE GIVEN THESE 3 STEPS: $38

OPPORTUNITY COST: $-3

WHICH SHOWS BETTER BEHAVIOR?

2/48

-$3 $0 $0

-$50 $0

SCORE: $-53

-$2 $0 $0

-$50 $0

SCORE: $-52

Figure 11: Subjects where shown either the REGRET-UI interface (top) or the Σr-UI (bottom). To avoid technical jargon in the REGRET-UI, $V^*_r(s^\sigma_0)$ is referred to as the "BEST POSSIBLE SCORE FROM START", $V^*_r(s^\sigma_{|\sigma|})$ as the "BEST POSSIBLE SCORE GIVEN THESE 3 STEPS", and $regret_d(\sigma|r)$ as the "OPPORTUNITY COST". In the Σr-UI, Σr is referred to as the "SCORE".

In Section 4 we find that the regret preference model better predicts real human preferences. In Section 5.3 we find that learning with preferences that are generated by the regret model induces better performance than learning with preferences that are generated by the partial return model. While human preferences are more in line with P_{regret}, they do not perfectly follow it. Can we push human preferences closer to the regret model in order to get better performance?

As a first step towards investigating this, we present human subjects with information about the ground truth regret for each segment at preference elicitation time. Specifically, we show subjects each segment's start state value, end state value, and regret. We avoid technical jargon when presenting human subjects with this information, and refer to this condition as the REGRET-UI. We additionally show another group of subjects information about *only* the partial return preference model, i.e. each segment's sum of rewards. We refer to this as the Σr-UI. An example of the interfaces for the REGRET-UI and Σr-UI is shown in Figure 11.

Note that we never explicitly tell human subjects they should use the presented information to inform their preferences. Instead, we include either the ground truth regret or partial

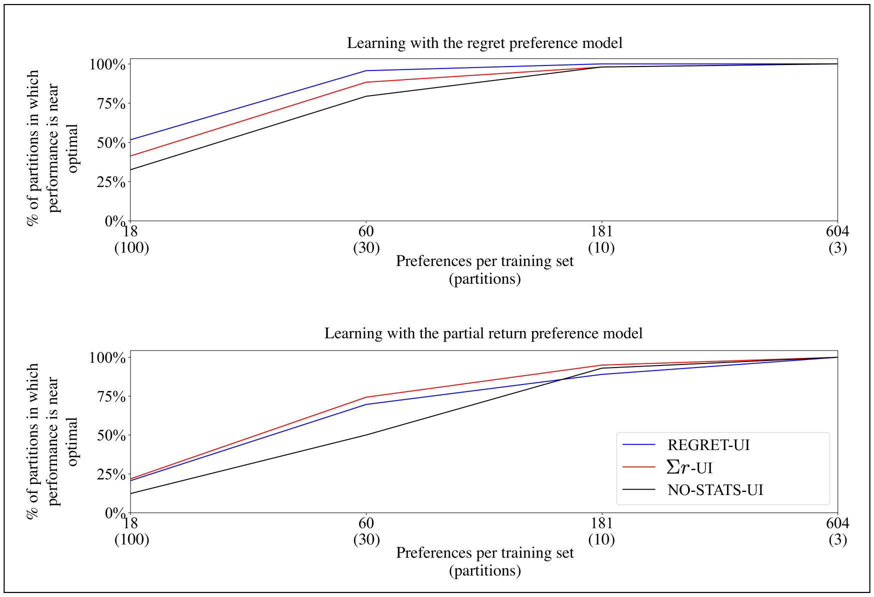


Figure 12: Performance comparisons over various amounts of human preferences from each UI condition. All results are averaged over 10 random seeds. The color of the line indicates under which condition preferences were elicited, while the plot indicates which preference model was used while learning.

return for each segment, allowing participants to independently discern the relevance and significance of this data. The condition where no information is shown to human subjects at preference elicitation time, which is what's used for all prior results and shown in Figure 4, is referred to as the NO-STATS-UI. Note that the only difference between the REGRET-UI, Σr-UI, and NO-STATS-UI is what information is shown at preference elicitation time. For each of the two new conditions, we collect all human preferences in the delivery task presented in Section 3.2. We follow the same elicitation and filtering procedure outlined in Section 3, resulting in $2{,}542$ preferences collected from the REGRET-UI condition and $2{,}545$ preferences collected from the Σr-UI condition.

To assess the effectiveness of P_{regret} and $P_{\Sigma r}$ in predicting preferences from each condition, we compute the cross-entropy loss for each model over the preference datasets collected from each condition. We follow the same methodology presented in Section 4. Unsurprisingly, the regret model best predicts preferences elicited from the REGRET-UI condition and the partial return model best predicts preferences elicited from the Σr-UI condition. These results, shown in Table 9, indicate that showing subjects ground-truth information about a segment's regret or partial return can nudge their preferences toward the regret or partial return preference models respectively.

Preference model	NO-STATS-UI Loss (n=1812)	Σr-UI Loss (n=2545)	REGRET-UI Loss (n=2542)
$P(\cdot) = 0.5$ (uninformed)	0.69	0.69	0.69
$P_{\Sigma r}$ (partial return)	0.62	**0.50**	0.55
P_{regret} (regret)	**0.57**	0.52	**0.44**

Table 2: Mean cross-entropy losses on test sets from predicting human preferences for three preference elicitation conditions. Lower loss is better.

We achieve the best performance when learning a reward function with P_{regret} and preferences collected from the REGRET-UI condition. Interestingly, learning with P_{regret} and preferences collected from the Σr-UI induces similar performance to learning with $P_{\Sigma r}$ and using those same preferences. These results are illustrated in Figure 12 and suggest that 1) we can nudge human preferences towards a certain model, and 2) doing so is beneficial when learning a reward function and policy. In turn, this provides a proof of concept for future work focused on how to nudge human preferences towards P_{regret} or $P_{\Sigma r}$ *without* access to ground-truth reward and value function information at preference elicitation time.

9.1 Limitations

This investigation relies on presenting human subjects with information about the true reward function and value function to nudge their preferences. Future work should address this limitation by designing preference elicitation interfaces for nudging human preferences without access to ground truth information. Additionally, we seek to teach human subjects about the regret or partial return model in a relatively simple grid-world domain. Whether findings in our domain translate to more complex domains requires further exploration.

A critical concern regarding this direction of research is that we assume there is a single true reward function to recover from human preferences. Our objective is to influence human preferences to more accurately recover this reward function. However, this assumption is precarious; if there is not a true reward function shared by all human subjects then nudging human preferences would likely hinder efforts to learn a human-aligned reward function. Consequently, practitioners should be wary of applying related methodologies to real-world problems where individuals may have differing preferences.

10 Directions for future work

This thesis presents the regret model of human preferences, which poses numerous advantages over the partial return model. Below, we outline and elaborate on additional directions for future work outside of the limitations discussed in Sections 6.1, 8.1, and 9.1.

Some prior work has focused on developing preference-based reinforcement learning algorithms with sample efficiency guarantees (Kong and Yang [2022], Pacchiano et al. [2021]). These algorithms and guarantees, however, rely on the partial return assumption. One important direction for future research involves analyzing previously proposed provably efficient reward learning algorithms using the regret preference assumption rather than the partial return preference assumption. Perhaps such an analysis will yield tighter performance bounds.

This work is entirely concerned with offline reward learning. Future research could focus on using the regret preference model to learn a reward function in an active learning setting instead, possibly resulting in increased sample efficiency. For example, Sadigh et al. [2017] introduced a popular method for choosing segment pairs for preference labeling; they select segment pairs that maximize the volume removed from the hypothesis space of possible reward functions. The volume removed, however, depends on human preferences which are assumed to arise from partial return. Reworking the query selection methodology proposed by Sadigh et al. [2017] with the regret assumption may yield a more sample-efficient active learning algorithm.

Additionally, approximating regret under a given reward function requires approximating $\tilde{V}^*_{\hat{r}}$ and $\tilde{Q}^*_{\hat{r}}$ which is costly. We propose a tractable approach to address this problem reliant on successor features and the assumption that reward can be expressed as as linear combination of weights and state-action-features. Future research should focus on other methods for approximating $\tilde{V}^*_{\hat{r}}$ and $\tilde{Q}^*_{\hat{r}}$ that do not require our linearity assumption and can efficiently be extended to more complex domains.

Like the partial return model, the regret preference model assumes humans are Boltzmann rational. As such, P_{regret} and $P_{\Sigma r}$ are parameterized using the logistic function. This assumption is common in the RLHF literature, but remains largely uninvestigated. An important direction of research involves investigating how humans use a segment statistic, such as regret or partial return, to actually generate preferences.

Future research should also focus on how to best elicit human preferences. For example, P_{regret} relies on the assumption that humans can differentiate between optimal and near-optimal behavior. While we purposefully violate this assumption in our delivery task, it may be even more difficult for a human to differentiate between behaviors of varying desirability in more complex domains. Can we design preference elicitation interface tools to address this problem and aid humans in generating preferences? Relatedly, in Section 9, we present initial work aimed at teaching human subjects about a preference model with the objective of nudging their preferences towards that model. An interesting direction of research involves teaching human subjects about a preference model in a simple domain with the objective of nudging their preferences towards that model in a new, more complex domain. For example, it may be straightforward to teach humans that they should account for start and end state values when evaluating a segment's desirability in our grid-world delivery domain. This instruction might then influence their preferences in a more complex setting, like in a robotics manipulation domain.

11 Conclusion

We begin by questioning a ubiquitous assumption made by prior works in RLHF: that a segment's desirability arises solely from its sum of rewards. We call the preference model that uses this assumption the partial return preference model ($P_{\Sigma r}$), but we believe it is incorrect. Instead, we propose the regret preference model (P_{regret}) which rests on the assumption that a segment's desirability arises from its deviation from optimal behavior. Our proposed regret preference model shows numerous improvements over the partial return model:

- The regret model better predicts real human preferences.
- Learning with the regret model from a dataset of human preferences results in more performant learned policies more often.
- When each preference model learns from synthetic preferences that perfectly adhere to it, the regret preference model also outperforms the partial return model.

Our findings indicate that the regret preference model is more successful descriptively, in predicting and learning from human preferences, and normatively, as the model we would prefer humans to adopt if given the choice. We observe, however, that influential prior work achieves remarkable results when learning from human preferences but using the partial return model, which we view as fundamentally flawed. We show that if preferences are generated by P_{regret} and learned from using $P_{\Sigma r}$, then the learned function is actually an optimal advantage function *not* a reward function. We pinpoint the consequences of mistaking an optimal advantage function for a reward function, resulting in a more theoretically principled perspective on prior work. Finally, we provide early evidence that we can nudge human preferences toward P_{regret} or $P_{\Sigma r}$ by modifying the preference elicitation interface, resulting in more performant learned policies.

References

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

A Rupam Mahmood, Dmytro Korenkevych, Gautham Vasan, William Ma, and James Bergstra. Benchmarking reinforcement learning algorithms on real-world robots. In *Conference on robot learning*, pages 561–591. PMLR, 2018.

OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. *arXiv preprint arXiv:1806.10293*, 2018.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems (NIPS)*, pages 4299–4307, 2017.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022.

Dorsa Sadigh, Anca D Dragan, Shankar Sastry, and Sanjit A Seshia. Active preference-based learning of reward functions. *Robotics: Science and Systems*, 2017.

Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. Reward learning from human preferences and demonstrations in atari. *arXiv preprint arXiv:1811.06521*, 2018.

Erdem Bıyık, Dylan P Losey, Malayandi Palan, Nicholas C Landolfi, Gleb Shevchuk, and Dorsa Sadigh. Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences. *The International Journal of Robotics Research*, page 02783649211041652, 2021.

Kimin Lee, Laura Smith, and Pieter Abbeel. Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. *arXiv preprint arXiv:2106.05091*, 2021a.

Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. B-pref: Benchmarking preference-based reinforcement learning. *arXiv preprint arXiv:2111.03026*, 2021b.

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

Xiaofei Wang, Kimin Lee, Kourosh Hakhamaneshi, Pieter Abbeel, and Michael Laskin. Skill preferences: Learning to extract and execute robotic skills from human feedback. In *Conference on Robot Learning*, pages 1259–1268. PMLR, 2022.

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

Amelia Glaese, Nat McAleese, Maja Trębacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Maribeth Rauh, Laura Weidinger, Martin Chadwick, Phoebe Thacker, et al. Improving alignment of dialogue agents via targeted human judgements. *arXiv preprint arXiv:2209.14375*, 2022.

OpenAI. Chatgpt: Optimizing language models for dialogue. OpenAI Blog `https://openai.com/blog/chatgpt/`, 2022. Accessed: 2022-12-20.

W Bradley Knox, Stephane Hatgis-Kessell, Serena Booth, Scott Niekum, Peter Stone, and Alessandro Allievi. Models of human preference for learning reward functions. *arXiv preprint arXiv:2206.02231*, 2022.

W Bradley Knox, Stephane Hatgis-Kessell, Sigurdur Orn Adalgeirsson, Serena Booth, Anca Dragan, Peter Stone, and Scott Niekum. Learning optimal advantage from preferences and mistaking it for reward. *arXiv preprint arXiv:2310.02456*, 2023a.

W Bradley Knox, Stephane Hatgis-Kessell, Serena Booth, Scott Niekum, Peter Stone, and Alessandro Allievi. Reproduction Data for: Models of Human Preference for Learning Reward Functions, 2023b. URL `https://doi.org/10.18738/T8/S4WTWR`.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado Van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. *arXiv preprint arXiv:1606.05312*, 2016.

Riad Akrour, Marc Schoenauer, and Michele Sebag. Preference-based policy learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 12–27. Springer, 2011.

Joey Hejna and Dorsa Sadigh. Inverse preference learning: Preference-based rl without a reward function. *arXiv preprint arXiv:2305.15363*, 2023.

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.

Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. *Sixteenth International Conference on Machine Learning (ICML)*, 1999.

Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8:279–292, 1992.

Joey Hejna, Rafael Rafailov, Harshit Sikchi, Chelsea Finn, Scott Niekum, W Bradley Knox, and Dorsa Sadigh. Contrastive prefence learning: Learning from human feedback without rl. *arXiv preprint arXiv:2310.13639*, 2023.

Dingwen Kong and Lin Yang. Provably feedback-efficient reinforcement learning via active reward learning. *Advances in Neural Information Processing Systems*, 35:11063–11078, 2022.

Aldo Pacchiano, Aadirupa Saha, and Jonathan Lee. Dueling rl: reinforcement learning with trajectory preferences. *arXiv preprint arXiv:2111.04850*, 2021.