# Towards Active Attention-Modified Policy Shaping

Taylor Kessler Faulkner[1], Elaine Schaertl Short[2], Andrea L. Thomaz[2]

***Abstract*— Robots that learn from their environment can also take input from human teachers. However, teaching the robot may not be these humans' only task, in which case the robot cannot rely on constant attention from a teacher. We propose an algorithm that learns a model of a human teacher's feedback and uses this model to plan when to actively ask the teacher for attention. This algorithm will allow the teacher to take breaks from teaching the robot to complete other tasks and enables the robot to ask for attention in areas of the human model in which it is confused. The goal of this algorithm is to balance human effort and time with robot learning.**

## I. Introduction

Robots that learn by exploring their environment can also leverage input from humans as added data. However, robots that learn from humans over extended periods of time cannot always expect a human teacher to be paying attention to them. Teachers may have other tasks to complete, other robots to oversee, or take breaks from supervising the robot. We consider robots using reinforcement learning (RL) with policy shaping [1]. This method enables a robot to both learn from interacting with its environment and receive feedback on its actions from people. Since the robot can learn from the environment as well as a human, the robot can continue learning while no human is giving feedback.

Policy shaping allows human teachers to give binary positive or negative feedback to a robot performing reinforcement learning [1]. Rather than being taken as rewards, this feedback directly influences the action policy of the robot, and is only interpreted as a positive or negative decision on a single state and action rather than a reward. When using RL with policy shaping, a robot can either passively wait for a teacher's attention or actively ask for attention. Attention-Modified Policy Shaping (AMPS) is a method of policy shaping that changes learning methods depending on whether a teacher is paying attention [2]. If there is not attention, the robot prioritizes actions that the teacher has previously approved. Otherwise, the robot prioritizes actions that the teacher has not seen before or previously approved actions. However, in this scenario the burden is on the teacher to decide when to check in with the learning robot. Actively deciding when to ask for attention can put less stress on the teacher to decide when to check in with the robot and direct the teacher to supervise the robot at more productive times, rather than relying on the human's judgment. However, allowing the robot to interrupt the teacher arbitrarily could become disruptive and prevent the teacher from accomplishing other tasks. Therefore, an algorithm that chooses informative times to interrupt the teacher is desirable.

We propose an algorithm, Active Attention-Modified Policy Shaping (Active AMPS) that allows a robot to actively ask for attention from a human teacher while allowing the human to take breaks from teaching. Active AMPS creates a model of a single teacher's feedback policy, calculating the likelihood that they will give positive or negative feedback to various action choices in the state space. This model is distributed over the entire state space using a similarity function between states. The robot plans to ask for attention in states in which it is uncertain of the teacher's potential feedback, but spaces these attention requests out using a time threshold to avoid asking for attention too often. Using its model of human feedback, the robot can decrease requests for attention as it becomes more confident in its model of the teacher, decreasing the burden on the teacher.

## II. Background

This work builds on previous research in three fields: human multi-tasking, active learning with thresholds for asking questions, and reinforcement learning with human feedback. The main premise for our algorithm is that human teachers should not be interrupted too often from other tasks. Previous work in multi-tasking research shows that frequent interruptions decrease human task performance on complex tasks [3], [4]. Prior research has focused on determining when to interrupt people to maximize task performance or minimize disruption [5], [6]. Our method allows us to do both, while maximizing performance of a learning agent.

There has been previous research on active learning that asks for help until some information threshold has been passed [7], [8], [9]. However, this work does not consider human teachers who have other tasks to complete while teaching the robot. We use reinforcement learning, unlike [7], [8], and unlike [9] we base our threshold directly on previous feedback from the human teacher.

Reinforcement Learning (RL) while learning from human feedback has been previously studied as well [10], [11], [1]. These methods do not actively ask for attention or help from the human teacher, and the teacher is not assumed to take breaks to complete other tasks. Our work extends the current work on RL with human feedback to long-term

This material is based upon work supported by the Office of Naval Research award numbers N000141612835 and N000141612785, National Science Foundation award numbers 1564080 and 1724157, and the National Science Foundation Graduate Research Fellowship Program grant number DGE-1610403.

[1]Department of Computer Science, University of Texas at Austin, Austin, TX, USA taylor.k.f@utexas.edu

[2]Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX, USA athomaz@ece.utexas.edu, elaine.short@utexas.edu

learning environments. Knox and Stone [10] also create a model of human feedback; however, their model interprets human feedback as a reward rather than a binary positive or negative input as policy shaping does. We use the policy shaping method, as previous work shows that people have trouble giving rewards as feedback [12].

## III. Methodology

We first describe the policy shaping algorithm [1], [13], and then show how Active AMPS fits in the policy shaping framework.

### A. Policy Shaping

We use Q-learning with Boltzmann exploration [14], [15] as the base reinforcement learning method for policy shaping [1], [13]. In this framework using Boltzmann exploration, the probability of taking any action $a$ given the learned Q-values is:

$$Pr_q(a) = \frac{e^{\frac{Q(s,a)}{\tau}}}{\sum_{a'} e^{\frac{Q(s,a')}{\tau}}} \qquad \text{[15], [13]}$$

where $\tau$ is a constant. A variable $C$, ranging from zero to one, is assigned to estimate the confidence in human feedback. The probability that an action is a good action based purely on human feedback is:

$$Pr_c(a) = \frac{C^{\Delta_{s,a}}}{C^{\Delta_{s,a}} + (1-C)^{\Delta_{s,a}}} \qquad \text{[1], [13]}$$

where $\Delta_{s,a}$ is the difference between the number of positive feedback and negative feedback received in $(s, a)$. This probability is combined with the probability $Pr_q(a)$ to give the probability of taking any action $a \in A$:

$$Pr(a) = \frac{Pr_q(a) Pr_c(a)}{\sum_{\alpha \in A} Pr_q(\alpha) Pr_c(\alpha)} \qquad \text{[13].}$$

### B. Active Attention-Modified Policy Shaping

We formulate the attention-requesting problem in the following way. We begin with a time threshold $t$, which limits how often the robot can ask for the human teacher's attention. After $t$ has been reached, the robot can ask for attention on up to $n$ states if it needs more information. The threshold $t$ can be set in terms of the number of actions that the robot takes or for some amount of minutes. One option for $t$ is to assign a constant threshold, for example 10 actions, which spaces attention requests evenly over the length of time that the robot learns. Another option is to begin with a low $t$ so that more feedback is received at the beginning, increasing $t$ over time to give the teacher more free time. Previous work has noted that human feedback is most important early in the reinforcement learning process [10], so in this proposal we begin with $t$ set to one action and increase this value over time.

To model the human feedback, we use a classifier with confidence estimations, such as a Support Vector Machine (SVM). We create a binary classifier $\phi(f_s, v_s, a)$ that, given the features of the current state $s$ ($f_s$), the value $v_s$ of the state $s$, and action $a$ predicts whether the teacher will give positive or negative feedback to $(s, a)$. $\phi$ takes $f_s$ as input so that the robot can distribute the model of human feedback around the state space. Rather than attributing feedback only to a specific state, we assume a relation between features of the state and the feedback that people give actions taken from that state. Thus similar states should receive similar feedback patterns.

The classifier $\phi$ input also includes $v_s$, the value learned for $s$ using RL, so that the robot can leverage its knowledge learned through the environment to create a model of probable human feedback. States that have higher value lead to higher reward areas, so often the human feedback for moving into a high value state will be positive. However, we do not want to automatically assume that transitions into high value states will get positive feedback. In some tasks a human teacher may want the robot to take a different path than the highest value path to a goal state. For example, consider a task in which the reward function only gives reward on the goal state, and there is a short but dangerous path to the goal with many obstacles as well as a longer but safe path to the goal. After rounds of learning, the robot may learn a policy that follows the short path to the goal. However, a cautious human teacher may want the robot to take a safer path to the goal, and will give feedback indicating this policy. In this case, the classifier $\phi$ may learn a low correlation between $v_s$ and positive feedback. However, for less cautious teachers $\phi$ may learn a high correlation between $v_s$ and positive feedback, allowing the robot to ask for less attention from the teacher as it learns values through its own exploration.

The classifier $\phi$ begins with a uniform model of the teacher's feedback, so that the probability of positive or negative feedback for any $(s, a)$ is 0.5, and the confidence of $\phi(f_s, v_s, a)$ is zero . Thus the robot holds the belief that there is a 50% chance of receiving positive feedback, $P(f_p|s, a)$, for each state-action pair $(s, a)$ from all states $S$ and all actions $A$. After each positive or negative feedback $f$ received from the teacher in a state-action pair $(s, a)$, the robot updates its beliefs for $P(f_p|s, a)$ by giving the input $(f_s, v_s, a)$ with the label $f$.

We set a confidence threshold $c$ such that if the confidence of $(\phi(f_s, v_s, a)|s \in S, a \in A) \geq c$ the robot will proceed to learn without asking for attention, as it is confident in its human model in all states. If there is an $(s, a)$ such that the confidence of $(\phi(f_s, v_s, a) < c$, the robot identifies the state $s_i$ with the lowest confidence in $\phi(f_{s_i}, v_{s_i}, a)$ and plans an action sequence of at least length $t$ in order to reach this state. If the task is not structured so that an action path of length $t$ is available to $s_i$, the robot can abandon the current run-through of the task and restart. If the confidence of the classifier is above $c$ in every state, however, the robot will continue learning using basic policy shaping, while the model of human feedback continues to give feedback to the robot.

## IV. Evaluation

We will evaluate this method both in simulation and on a physical robot. We will use a manipulation task with multiple objects and actions such as stacking, pushing, etc, in which a large reward is given to the robot upon successful

**Algorithm 1:** Active AMPS

```
t = 1;
S, A = states,actions;
φ = init_classifier();
while learning do
    if min(confidence(φ(f_s, v_s, a))|s ∈ S) > c then
        s_i = argmin(confidence(φ(f_s, v_s, a))|s ∈ S);
        plan_path(t,s_i);
        execute_path();
        s = current state;
        request_attention(n);
        a = argmin(confidence(φ(f_s, v_s, a))|a ∈ A);
        f = get_feedback();
        update_policy_shaping(f);
        update_φ(f_s, v_s, a);
    else
        a = next policy shaping action;
        f = φ(f_s, v_s, a));
        update_policy_shaping(f);
    end
    increase_t();
end
```

completion of a task and small negative rewards are given at each step to encourage reaching the goal quickly. The participant will divide their time between a distractor task and teaching the robot, paying attention to the robot when attention is requested by Active AMPS. The distractor task will be structured so that we can measure how much of the task is completed; for example, solving math problems or labeling images.

In simulation, we will use an oracle to provide feedback to Active AMPS and test the confidence threshold $c$, the classifier $\phi$, and the pattern of requests. The variable $c$ will be varied from 0 to 1 to optimize for robot learning speed while balancing the time cost to a human teacher. The ideal value may be task-dependent, which can be tested over several reinforcement learning tasks. The type of classifier for $\phi$ will also be tested, as well as other metrics for which action will be most informative, such as whether a person has seen a state-action pair $(s, a)$ previously as used in AMPS [2]. The pattern of requests can be varied in many ways, such as randomly timed requests without planning to a high information state (at least $t$ steps apart), keeping $t$ at a constant time versus allowing more requests at the beginning, and asking for attention in blocks ($n > 1$) versus asking for attention in one state at a time ($n = 1$). These options will be compared to baselines in which the robot either never receives attention (reinforcement learning) or always receives attention (policy shaping).

We will use a human study with a physical robot to test human aspects of the algorithm. First, we will validate the benefits of the robot actively asking for attention instead of the teacher deciding when attention is necessary. To do so, we will compare Active AMPS to AMPS, which does not actively ask for attention, instead passively changing learning styles based on the presence of attention from a teacher [2]. AMPS prioritizes actions that have received positive feedback when the teacher is not paying attention and splits priority between actions that have received positive feedback and actions the teacher has not seen when attention is present. Comparing Active AMPS and AMPS will test whether actively asking for attention improves robot learning and the human teacher's ability to complete a distractor task. We will also test this hypothesis by running Active AMPS without requests for attention. In this case, the robot cannot plan to a state with low information, but can take the action with the lowest $confidence(\phi(f_s, v_s, a))$ when the human is paying attention. Second, the time threshold, $t$, and the number of states for which the robot asks for attention, $n$, will also be tested in a human study. Different lengths of time and size of $n$ will be tested across users to determine which combination is the least disruptive to users while still enabling the robot to learn at an acceptable rate.

## V. Discussion

We propose Active AMPS in order to allow robots to actively ask for attention from human teachers while learning. We believe this is an important function for robots to have so that people can take breaks from teaching a robot to complete other tasks while still providing the robot with feedback when it is uncertain. Rather than rely on a distracted person to decide when to pay attention to the robot, enabling active attention requests allows a teacher to focus on other tasks until summoned by the robot and assures that a teacher will give feedback to the robot in uncertain areas of the state space.

In our human studies, we expect to find that people will not want to interrupted frequently by the robot, and would rather give feedback to blocks of actions infrequently than single actions frequently. When given the choice of when to pay attention rather than being actively asked, we expect people to either focus almost entirely on the robot or on the distractor task. This may depend on how familiar a participant is with robots and how interesting the distractor task is. We believe that the ability to actively request help will better balance participant performance on the distractor task with robot learning on the reinforcement learning task.

In general, it is important to take attention from a human teacher into account. If the robot passively continues learning in the same way whether a human is present or not, it may not take as much advantage of human presence as possible when a teacher is available. Particularly in long-term learning environments, counting on the constant presence of a human teacher is not always feasible. Therefore, we propose Active AMPS in order to give breaks to a teacher while asking for attention when uncertain about human feedback. In its current state, Active AMPS does assume that a teacher will be available when attention is requested, which may not be true if no human is present. Future work can extend this method to include how the robot should behave when no person is present.

## VI. Conclusion

In this work, we introduce Active AMPS, an algorithm that models human feedback to policy shaping in order to reduce the amount of total feedback a human teacher must give and allow the teacher to take breaks to complete other tasks. This algorithm learns a model of a single teacher using a binary classifier, although a more general classifier for types of teachers could be learned with enough human subjects. Active AMPS enables the robot to leverage environmental feedback and the human feedback model to become more independent over time, relying less on actual feedback from a teacher as it becomes more confident in the human's answers.


## Acknowledgements

We thank Prof. Guy Hoffman for thoughtful discussions about this work.


## References


[1] S. Griffith, K. Subramanian, J. Scholz, C. L. Isbell, and A. L. Thomaz, "Policy shaping: Integrating human feedback with reinforcement learning," in *Advances in neural information processing systems*, 2013, pp. 2625–2633.

[2] T. Kessler Faulkner, E. S. Short, and A. L. Thomaz, "Policy shaping with supervisory attention driven exploration." to appear in Intelligent Robots and Systems (IROS), 2018 IEEE/RSJ International Conference on. IEEE, 2018.

[3] S. Monsell, "Task switching," *Trends in cognitive sciences*, vol. 7, no. 3, pp. 134–140, 2003.

[4] C. Speier, J. S. Valacich, and I. Vessey, "The influence of task interruption on individual decision making: An information overload perspective," *Decision Sciences*, vol. 30, no. 2, pp. 337–360, 1999.

[5] P. D. Adamczyk and B. P. Bailey, "If not now, when?: the effects of interruption at different moments within task execution," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2004, pp. 271–278.

[6] A. Nigam and L. D. Riek, "Social context perception for mobile robots," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 3621–3627.

[7] S. Chernova and M. Veloso, "Interactive policy learning through confidence-based autonomy," *Journal of Artificial Intelligence Research*, vol. 34, pp. 1–25, 2009.

[8] M. Cakmak, C. Chao, and A. L. Thomaz, "Designing interactions for robot active learners," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 2, pp. 108–118, 2010.

[9] J. A. Clouse, *On integrating apprentice learning and reinforcement learning*. University of Massachusetts Amherst, 1996.

[10] W. B. Knox and P. Stone, "Reinforcement learning from simultaneous human and mdp reward," in *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 2012, pp. 475–482.

[11] W. Saunders, G. Sastry, A. Stuhlmueller, and O. Evans, "Trial without error: Towards safe reinforcement learning via human intervention," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2018, pp. 2067–2069.

[12] A. L. Thomaz and C. Breazeal, "Teachable robots: Understanding human teaching behavior to build more effective robot learners," *Artificial Intelligence*, vol. 172, no. 6-7, pp. 716–737, 2008.

[13] T. Cederborg, I. Grover, C. L. Isbell, and A. L. Thomaz, "Policy shaping with human teachers." in *IJCAI*, 2015, pp. 3366–3372.

[14] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[15] C. Watkins, "Models of delayed reinforcement learning," *PhD thesis, Psychology Department, Cambridge University*, 1989.