Reinforcing safety, with style: exploring reward shaping through human feedback

Cristian Opris - Stanford SCPD
(copris@stanford.edu)

ABSTRACT

"Learning from Human Preferences": recently released OpenAI paper exploring learning a reward shaping function from sparse human feedback expressed as preferences over alternate state-action trajectories encoded as video clips and presented in a web interface

Objective: reproduce the results of the paper in a new setting - simple ball bouncing games based on the Unity3D ml-agents RL framework - and explore an enhancement where we train a reward shaping predictor for a subgoal, which is then reused as an additive reward shaping bias to train a new reward shaping predictor for the end goal.

MODELS

RL Algorithm - OpenAI Proximal Policy Optimization : variant of policy gradient, implemented as 2 x 64 node hidden layer MLP:

$$L(\theta) = \hat{\mathbb{E}}_t[\nabla_\theta log \pi_\theta(a_t | s_t) \hat{A}_t]$$

Reward predictor - predicts latent reward $\hat{r}$ from probabilities P of user preference over alternate trajectories (σ1, σ2):

$$\hat{P}[\sigma^1 \succ \sigma^2] = \frac{\exp \sum \hat{r}(o_t^1, a_t^1)}{\exp \sum \hat{r}(o_t^1, a_t^1) + \exp \sum \hat{r}(o_t^2, a_t^2)}.$$

2 x 64 node MLP with softmax loss over binary preferences μ1 and μ2:

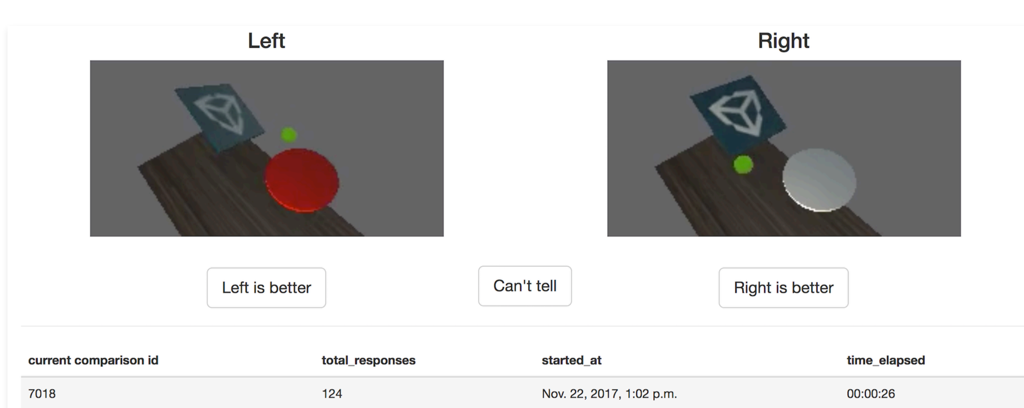
$$loss(\hat{r}) = - \sum_{(\sigma^1, \sigma^2, \mu) \in D} \mu(1) log \hat{P}[\sigma^1 \succ \sigma^2] + \mu(2) log \hat{P}[\sigma^2 \succ \sigma^1]$$

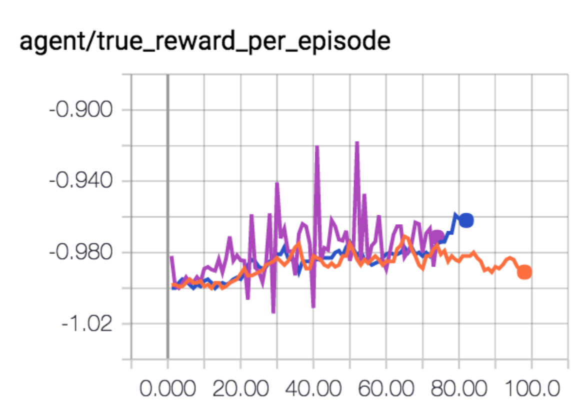
Reward bias: a pre-learned reward function's predictions can be used as additive bias when learning a new reward function:

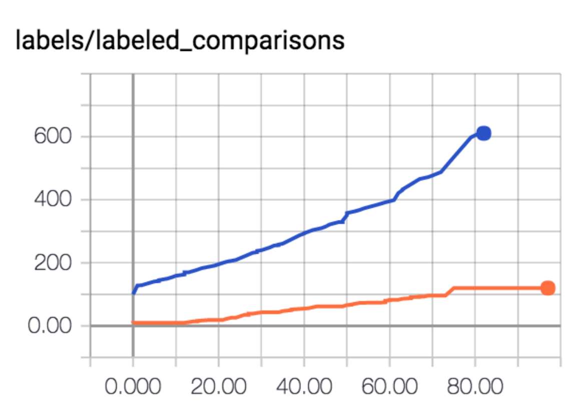
$$\hat{r}_{total}(o_t, a_t) = \hat{r}_{new}(o_t, a_t) + \hat{r}_{bias}(o_t, a_t)$$

BASELINE

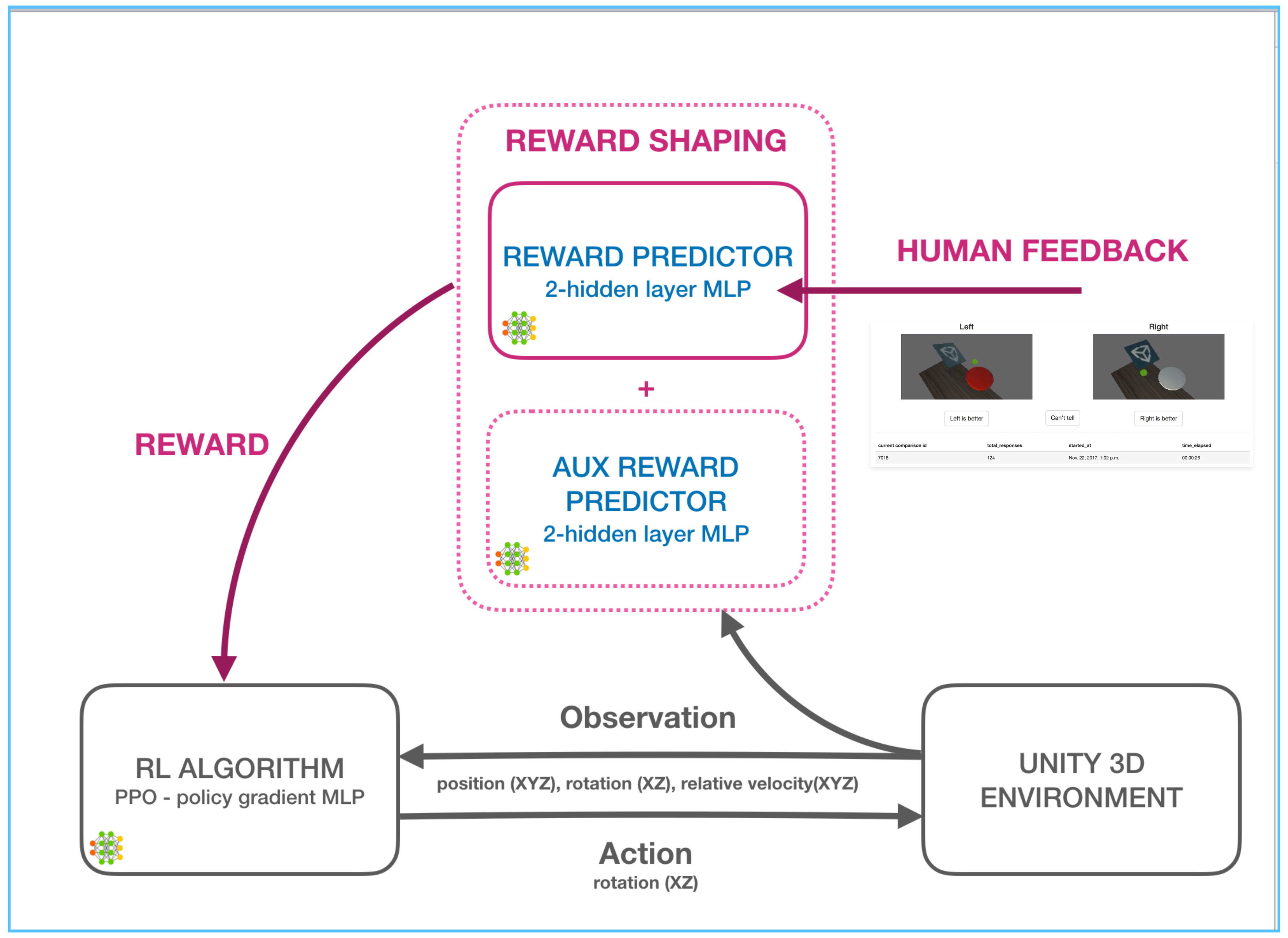
Bounce a ball to target with:
RL = hard-coded reward
SYNTH = preferences from hard-coded reward
HUMAN = preferences







Similar performance to coded reward achieved with as few as 100 human comparison labels.

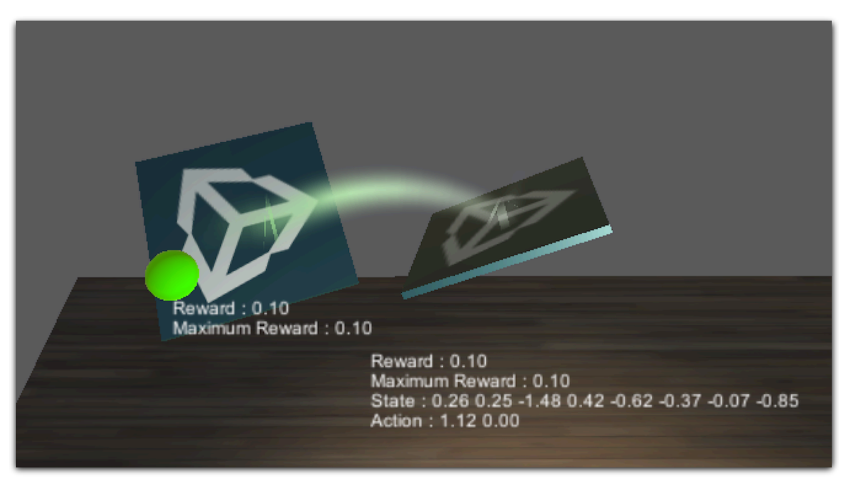


MAIN EXPERIMENT

Bounce ball between platforms

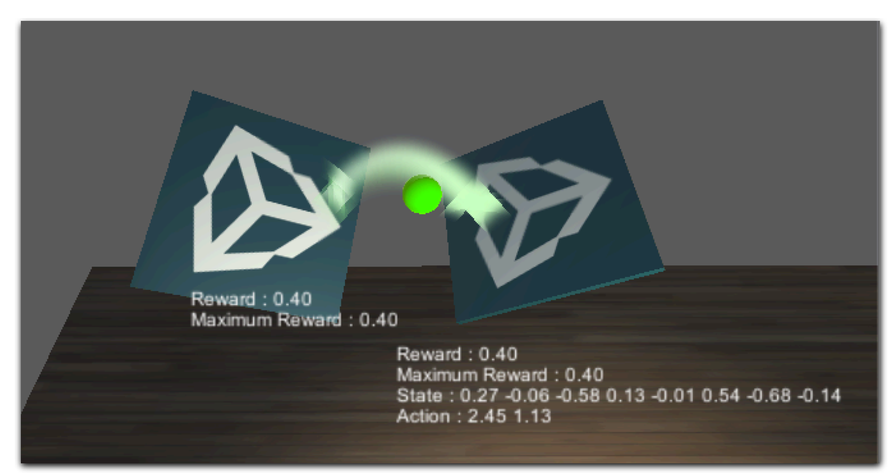
RL = hard-coded reward:
+0.1 if any platform hit, -1 if ball lost

Model learns to infinitely bounce ball on one platform as quickly as possible !



HUMAN1 = train right platform using human feedback to bounce ball towards left platform. Model learns correct behaviour for right platform, but left platform gets stuck into wrong pattern with no good trajectories being offered for human preference selection.

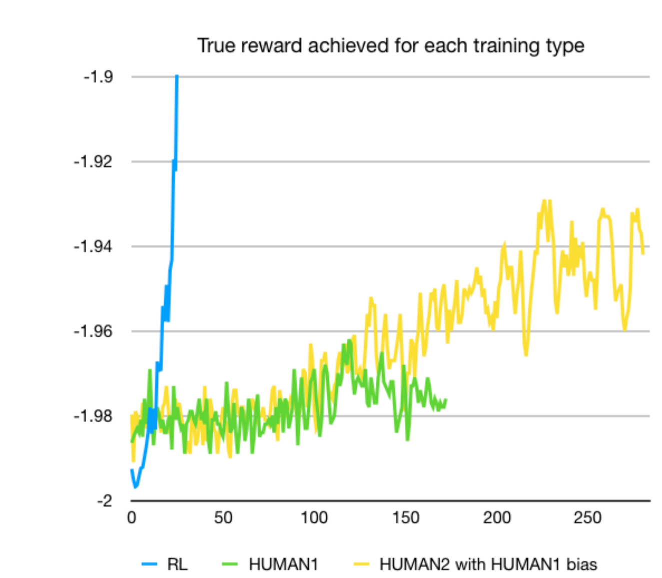
HUMAN2 = reuse the first reward predictor with weights fixed as an additive shaping bias for the reward of a new predictor, which can be trained to drive the correct behaviour for left platform, while the correct behaviour is preserved for the right platform. Success !

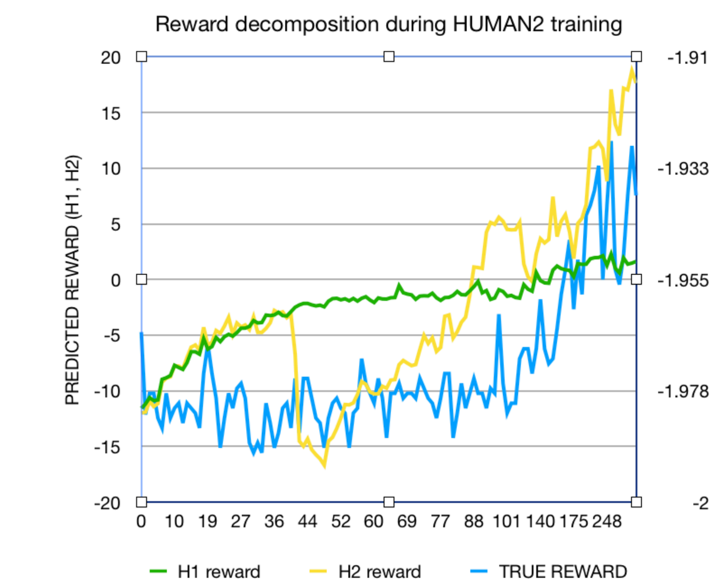


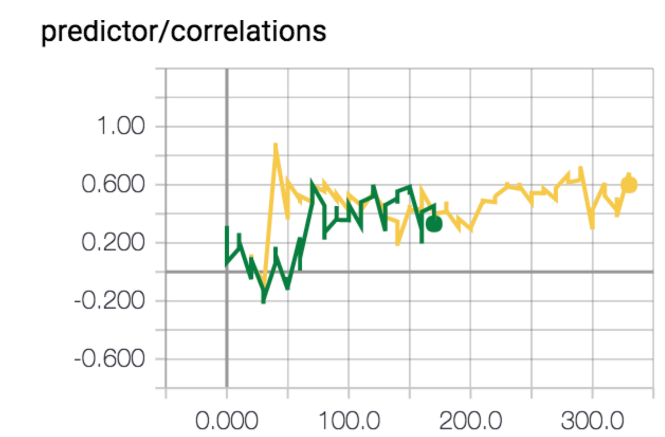
RESULT ANALYSIS

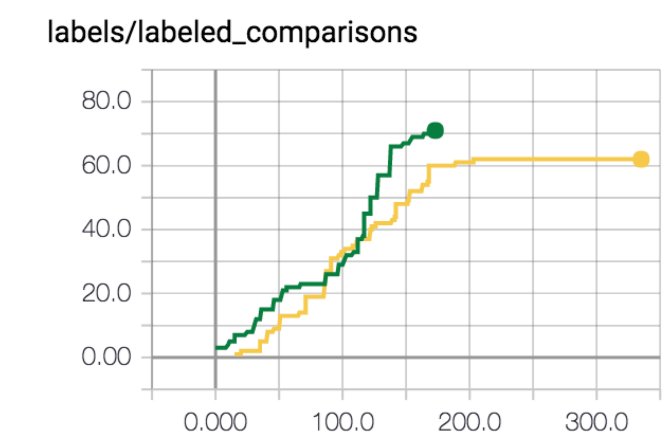
REWARD FUNCTION	MAX TRUE REWARD	TIMESTEPS FOR MAX REWARD	TOTAL HUMAN DECISIVE COMPARISONS
RL	-1.383	31000	n/a
HUMAN1	-1.962	119000	72
HUMAN2 (+ HUMAN1 bias)	**-1.929**	226000	63

RL = potentially infinite reward from optimal but undesired behaviour
HUMAN1 = first reward predictor learns correct behaviour for right platform, accumulating some true reward
HUMAN 2 = training the new reward function is helped by the previously learned shaping bias: the new reward is maximised while the bias is preserved, while it's predicted reward is also optimised for by the RL model. As few as 60-70 human decisive comparisons are enough to train each reward function, to correlate with accumulating true reward, however with a very different style of behaviour.









CONCLUSION

Using human feedback proves to be a practical approach to training RL systems in novel ways. The success of our relatively crude approach of additive reward shaping suggests the possibility of a future where combining fairly standardized learning models in unsophisticated, albeit creative ways, could yield meaningful results. Given more time, we'd have liked to study and approach reward shaping more rigorously, and apply this on more complex environments, e.g. OpenAI Roboschool.

References:

https://blog.openai.com/deep-reinforcement-learning-from-human-preferences/
https://github.com/nottombrown/rl-teacher/
https://github.com/Unity-Technologies/ml-agents/blob/master/docs/Example-Environments.md
Ng, A.Y., Harada, D., Russell, S.J.: Policy invariance under reward transformations: