Transferring While Playing the RL Agent

Matthias Müller-Brockhausen[1] and Hélène Plisnier[2]

[1]Universiteit Leiden (LIACS)
[2]Vrije Universiteit Brussels
[1]m.f.t.muller-brockhausen@liacs.leidenuniv.nl
[2]helene.plisnier@vub.be

Video: `https://surfdrive.surf.nl/files/index.php/s/EG86cZQzHopsZFn`

Abstract. Transfer Learning, a sub-domain of Reinforcement Learning, allows for an agent to learn new tasks faster by intelligently reusing skills acquired in old tasks. Arguably, to have a chance at life-long learning, future RL algorithms have to integrate some sort of transfer mechanism to exploit relevant knowledge from the past. In parallel, it is paramount to produce demonstrations targeting a non-expert audience, offering a look into AI mechanisms that may become omnipotent in future technology. We believe Transfer Reinforcement Learning to be one of such mechanisms. In this demo, we let volunteers play the role of an RL agent learning Mountain Car, from the Gym, with episodes of varying gravities. The gravity changes occur unknowingly to the participant. Our demonstration is deployed as a web application that not only allows for easy recruitment of a large amount of volunteers, but also records rewards and actions per timestep. The purpose of this demonstration is twofold: i) illustrate a Transfer Learning through changing environmental dynamics application to non-experts, ii) collect data from a large set of volunteers to compare human learning in the above-mentioned TL setting to that of an actual RL agent (such as PPO).

Keywords: Reinforcement Learning · Transfer Learning · Volunteer-Driven Demonstration

Transfer Learning

In the context of Reinforcement Learning (RL), Transfer Learning (TL) aims at limiting the time spent by an agent relearning solutions to problems it may have already encountered in the past, by reusing already available knowledge as much as possible. A TL setting often involves a source task and a target task; the agent first learns the source task from scratch, without using conventional RL, then tackles the target task while applying a specific TL technique exploiting the policy learned in the source task [4, 5]. A crucial element of a TL problem is the way(s) in which the source and target tasks differ. Although there are numerous ways to make environments different, as detailed by [4], the most investigated settings tend to be a change in the transition dynamics of the environment [3]. In

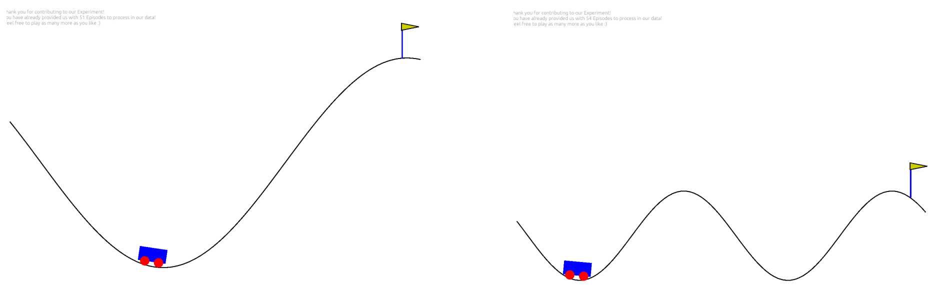

Fig. 1. In addition to the gravity changes, the demonstration includes two different scenarios: the original Mountain Car scenario where the car must drive up one hill, and a second, novel one where the car must drive up two hills to reach the goal. This way, not only must the volunteer transfer across different environment dynamics, but also transfer their skills to new situations.

a navigation task, for instance, one can change the dynamics of an environment by simply increasing or decreasing the value of the gravity parameter.

The demonstration

In this demonstration, we let a human participant learn to solve Mountain Car, an RL environment well-known by the research community. We implemented this environment in Rust [2], inspired by the Gym [1] implementation [1], as a part of a web-application allowing participants to "play" RL environments from their own machine. This helps with the recruitment of volunteers, as participating is as simple as clicking on a link, overall safety of everyone involved, since volunteers do not have to meet us in person to be able to participate. Finally, it allows multiple volunteers to access the demo at the same time, using their own smartphone or laptop. We ask volunteers to play 50 episodes of Mountain Car; the gravity value increases every couple of episodes, unbeknownst to the volunteer. This allows to evaluate the volunteers' ability to transfer skills acquired in previous episodes to new, more challenging ones, and to compare human performance to that of RL algorithms applied to the same problem.

Acknowledgement

The second author is funded by the Science Foundation of Flanders (FWO, Belgium) as 1SA6619N Applied Researcher.

[1] `https://github.com/openai/gym/blob/master/gym/envs/classic_control/mountain_car.py`

References

1. Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., Zaremba, W.: Openai gym. arXiv preprint arXiv:1606.01540 (2016)
2. Hoare, G.: The rust programming language. URL http://www. rust-lang. org (2013)
3. Müller-Brockhausen, M., Preuss, M., Plaat, A.: Procedural content generation: better benchmarks for transfer reinforcement learning. In: 2021 IEEE Conference on games (CoG). pp. 01–08. IEEE (2021)
4. Taylor, M.E., Stone, P.: Transfer learning for reinforcement learning domains: A survey. Journal of Machine Learning Research (2009)
5. Zhu, Z., Lin, K., Zhou, J.: Transfer learning in deep reinforcement learning: A survey (2020)