Discovering Reinforcement Learning Algorithms

Advances in Neural Information Processing Systems 33 (NeurIPS 2020)
17 July 2020

DeepMind
Junhyuk Oh *Matteo Hessel* *Wojciech M. Czarnecki*
Zhongwen Xu *Hado P. van Hasselt* *Satinder Singh, David Silver*

Problem

RL algorithms update the agent's parameters using discovered update rules. Automating the process of update rule discovery from given data can lead to much more efficient algorithms instead of doing so manually. Such automation is known as Learning to Learn. This concept shows that it's possible not only to learn to optimize static well-defined goals but to learn the meta-level information about the goal as well. Learning meta-information is a simple task if the environment domain is static, and the agent will interact with a very close environment to the one it learned to deal with. On the other hand, it's not that simple to generalize on other environments from different domains using the same automated update rule. The problem is to discover general algorithms that are effective for a broader class of agents and environments instead of being adaptive to a particular environment. There have been a few attempts to generalize and meta-learn RL algorithms, such as MetaGenRL. It was proposed to learn a domain-invariant policy update rule, that could generalize from a few MuJoCo environments to other MuJoCo environments. However, no prior work has attempted to discover the full update rule; instead, they all relied on value functions, which is the most fundamental building block of RL bootstrapping. This paper proposes a new methodology called Learned Policy Gradient (LPG) that can generalize to complex Atari games. LPG meta-learns its own mechanism for bootstrapping. This shows the potential to discover general RL algorithms from data in an automated way that can be applied in different domains.

Methodology

The goal is to find the optimal update rule for policy and value functions, parameterized by η, from a distribution of environments $P(\varepsilon)$ and initial agent parameters $P(\theta_0)$: $\eta^* = argmax_\eta(E_{\varepsilon\sim p(\varepsilon)}E_{\theta_0\sim p(\theta_0)}[G])$, where G is the expected return at the end of the lifetime. LPG – the update rule parameterized by η – is proposed to be a backward LSTM network that produces the updates of the policy and the prediction vector – prediction of the state – using the prediction vector of current and previous steps $y_\theta(s_t), y_\theta(s_{t-1})$, and the previous policy $\pi(a_{t-1}, s_{t-1})$. LPG is domain-invariant as it takes neither action space nor observation space as an input, and it's environment-invariant as well, as it meta-trains its parameter η based on the average of the policy gradients generated by different agents in different environments while training the agents using the current belief in an end-to-end manner.

Results

LPG was evaluated on different environments, such as delayed reward, noisy reward, and long-term credit assignment environments, where LPG outperformed A2C on most of them, which shows that LPG is an even better solution than manually designed RL algorithms. Regarding the prediction part, it is implicitly shown that the LPG is asking the agent to predict future rewards and use such information for bootstrapping. This why prediction values converge on a better estimate than in other algorithms.

Comparison to curriculum

Our Curriculum proposes the methodologies in RL in the case that both objective and update rule are static and fully defined. On the other hand, this paper tries to meta-train their parameters to get more efficient and adaptive RL algorithms that generalize to different domains and different environments.

Critique

This paper provides a robust approach to get adaptive RL algorithms. However, it assumes that there are plenty of resources to train multiple agents at the same time and share their gradient information with each other to meta-train the update rule. Another problem is that they assume the samples of environments they take to represent the whole possibilities of environments, which is not the general case for sure.