Summary of Curiosity-driven Exploration by Self-supervised Prediction

Ashutosh Kumar Jha, ME14B148
Mechanical Engineering, IIT Madras
Topics in RL Course

***Abstract*—In this technical report, we summarize the contents and ideas presented in the technical paper titled *Curiosity-driven Exploration by Self-supervised Prediction*[1]. We also present the possible shortcomings of the work.**

***Keywords*—Deep Reinforcement Learning, Exploration-Exploitation Dilemma, Intrinsic Reward**

I. Summary

Non-trivial ways to perform exploration is an active research area in the field of Deep Reinforcement Learning. Such non-trivial exploration techniques are particularly useful in environments where there are sparse rewards. This work provides with an algorithm to do the same by using an **intrinsic** reward.

In a typical policy gradient based RL algorithm, we build a model that can predict the policy of the agent. The basic idea of this paper is to also train two other models: an inverse dynamics model on the go which would predict the action given the present state and the future state *and* a forward model which tries to predict the feature encoding of the future state given the feature encoding of the present state and the action taken (Future being one step ahead). Mathematically the inverse and the forward models are given by:

$$\hat{a_t} = g\left(s_t, s_{t+1}; \theta_I\right)$$
$$\hat{\phi}(s_{t+1}) = f\left(\phi(s_t), a_t; \theta_F\right)$$

where both f and g are deep neural networks with parameters θ_F and θ_I respectively.
The agent's reward function is then augmented with an intrinsic reward using the forward model as follows:

$$r_t = r_t^i + r_t^e$$
$$r_t^i = \frac{\eta}{2}||\hat{\phi}(s_{t+1}) - \phi(s_{t+1})||_2^2$$

Such a intrinsic reward makes sense; If the agent predicts a state based on what it has been seeing so far and the true state is different from the prediction, it implies that the behavior of the agent has been exploratory and such a behavior would also give high intrinsic reward in our current setup.
To put things together, we now have three networks: a policy network , a forward model, and an inverse model parametrized by θ_P, θ_F and θ_I respectively. The losses for each of these networks is given by:

$$L_P = -\mathbb{E}_{\pi(s_t;\theta_P)}\left[\sum_t r_t\right]$$
$$L_F = \frac{1}{2}||\hat{\phi}(s_{t+1}) - \phi(s_{t+1})||_2^2$$
$$L_I = -\log(p(\hat{a_t} = a_t))$$

The losses due to the forward and inverse models quantify the "curiosity" of the agent and if seen carefully, training the forward and the inverse models is supervised learning because the true values of $\phi(s_{t+1})$ and a_t is known to the agent (in fact self supervised because these are obtained by the agent upon actually going out in the environment and obtaining trajectories).
The complete optimization objective is then given by:

$$\min_{\theta_P,\theta_I,\theta_F} \left[-\lambda L_P + \beta L_F + (1-\beta) L_I\right]$$

A. Experiments and Results

The paper shows results on VizDoom and Super Mario Bros, both from OpenAI Gym. The paper shows results only by using Intrinsic Curiosity Module(ICM) with A3C as the base algorithm.
The Baseline Methods that they consider are Vanilla A3C and another model they call ICM + A3C(Pixels) where the inverse model is removed and the function ϕ is an identity over the pixel values. The paper also compares its results against TRPO+VIME which was the state-of-the-art before this paper was published.

In case of Doom the environments are made dense,sparse and very sparse where even though the reward is provided only in the terminal state, the re-spawning of the agent is uniformly random in case of dense; in a fixed far room in case of sparse and in a fixed farther room in case of very sparse rewards.
In all the three cases ICM+A3C performs the best (with the exception of sparse rewards where ICM + A3C(pixels) is the best).

As an analysis the authors also show experiments which test the robustness of the algorithm proposed by augmenting 40% of the agent's vision by white noise. Here they show that while ICM + A3C is robust that is not the case with ICM + A3C(pixels).

The next set of experiments show results in the no reward case where the goal is to let the agent explore as much as

possible. They show results in the no reward case on both Doom and Mario. They show 30-50% coverage on both the environments and is the first paper to show good exploration in the no-reward setting.

The paper also shows results in transfer learning as well where they test the policy learned by maximizing curiosity in Level I of Mario in subsequent levels in three different ways: 1) Apply the policy as it is; 2) Fine tune the policy based on the intrinsic reward; 3) Fine tune the policy to maximize extrinsic reward. In all the three cases, the paper shows good results in Level II and in particular in Level III of the game. In fact in Level 2, the fine tuned policy is better than the one trained from scratch.

II. Shortcomings

In this section we make a note of possible short comings of this paper:

- The equations in the paper are all messed up. The eqn(2) is supposed to be $\hat{a_t} = g\left(\phi(s_t), \phi(s_{t+1}); \theta_I\right)$.
- The authors don't show any discounting factor in the paper, however the code that they have linked to does have discounting.
- The optimization objective is very messed up in the sense that while trying to minimize $-L_P$, we try to maximize $||\hat{\phi}(s_{t+1}) - \phi(s_{t+1})||_2^2$, however while trying to minimize L_F at the same time, we try to minimize the same quantity. A more detailed analysis of the effect of the hyper-parameters would thus be good.
- The paper shows no results on the famous Montezuma's Revenge which has become a standard benchmark for exploration based algorithms.

III. Conclusion

The paper introduces an interesting way of exploration in the Deep RL setting with the help of an intrinsic reward function. While the obvious drawbacks of higher computational time is introduced due to updating parameters of 3 different networks instead of 1, the paper is the first to show how to explore in the absence of rewards (the no-reward setting). The results on the sparse reward settings are good too however results on some standard domains like ALE/MuJoCo would be nice to see.

References

[1] Curiosity-driven Exploration by Self-supervised Prediction, Deepak Pathak, Pulkit Agrawal, Alexei A. Efros and Trevor Darrell. International Conference on Machine Learning (ICML), 2017.