Reinforcement Learning with Visual Inputs

Daeyeop Kim
Luddy School of Informatics, Computing, and Engineering Indiana Univeristy

Abstract

-In Reinforcement Learning a program (aka agent) learns to control the world by trial and error. The agent does not get direct instructions regarding what should be done in every state. Instead the agent has to act in the world and it is given rewards in some situations. The agents goal is to maximize the long term sum of rewards obtained from the environment. Over the last few years RL has shown huge successes, for example winning against the world champion in the game of Go. This project explained the potential sufficiency and RL with simple representations are with visual inputs.

Introduction

What is PPO?

- PPO is stands for Proximal Policy Optimization, which is a reinforcement learning algorithm or policy used to train agents to perform tasks in environments.
- PPO has been shown to be effective in a variety of reinforcement learning tasks, including game playing, robotics, and control. So, we use this PPO in this project.

What is Cartpole?

- This is a simple challenge task that is often used to test RL agents.
- In this environment, There is a stick and base. and the base moves horizontally(left or right)
- The goal of the agent is to balance the pole for as long as possible. (make the stick to stand straight)
- The natural representation of the state is
- x: The horizontal position of the cart. Range: (-2.4 and 2.4.)
- x_dot: The rate of change of the horizontal position of the cart. Range: (-inf and +inf.)
- theta: The angle of the pole with respect to the vertical direction. Range: (-12 and +12 degrees.)
- theta_dot: The rate of change of the angle of the pole. Range: (-inf and +inf.)

Method

What is MLP?

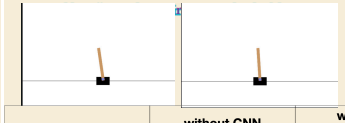
- MLP stands for Multi-Layer Perceptron, which is a type of neural network. It is a feedforward neural network that consists of multiple layers of neurons, including an input layer, one or more hidden layers, and an output layer. Each layer is made up of multiple artificial neurons that are connected to the neurons in the next layer.
- MLPs are widely used in machine learning for tasks such as classification, regression, and pattern recognition. They are particularly useful for tasks involving non-linear relationships between the input and output variables.

What is CNN?

-CNN stands for Convolutional Neural Network, which is a type of deep neural network commonly used for image and video processing tasks.
-CNNs use convolutional layers to filter and extract features from the input images, followed by pooling layers to reduce the size of the output.
- During training, the weights of the filters and other layers are optimized using a loss function and backpropagation.

What is VAE?

-VAE stands for Variational Autoencoder, which is a type of generative model used in unsupervised learning.
- A VAE is a neural network that is trained to encode high-dimensional data, such as images or text, into a lower-dimensional latent space, and then decode the latent space back into the original data.



Research process

Step 1: Experiment 1 Learning with non photographic-dataset

- Incorporated a successful RL agent in a compact and meaningful state description that captures the physical environment.
- Used Gym environment for the experiment and set up the RL algorithm from an existing implementation(model: PPO, environment: Cartpole-v1, policy: MlpPolicy).
- Used numerical data for RL learning in the native space stake.

Step 2: Experiment 2 Learning with photographic-dataset with using CNN method for reinforcement learning. (Before encoding)

- Hid the compact input from the agent and gave it pictures of the scene instead.
- Used the PPO policy with CNN neural network method for RL learning with images representing stake.
- Resized the photo data to 64x64 and compared the CPU and GPU time difference.
- Compared the results of RL learning with and without photographic data.

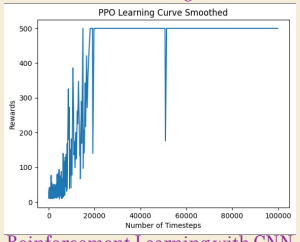
Step 3: Experiment 3 Learning with photographic-dataset with using CNN method for reinforcement learning.(After Encoding)

- Used Mlp with VAE and LinearVAE to encode and decode the photographic data.
- Checked the before and after images from VAE to ensure it was good enough for RL.
- Used the low dimension state encoding compacted by VAE in the PPO RL agent using a MLP.

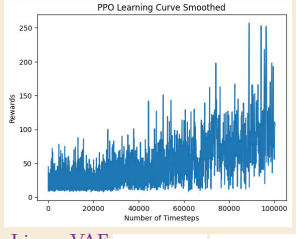
	without CNN	with CNN/ without encoding(CPU)	with CNN/ without encoding(GPU)	with CNN/ with encoding
Learning time(sec)	160.65 sec	1776.07 sec	651.33 sec	1508.73 sec
timesteps	1000000	1000000	1000000	1000000
image	X	64X64	64X64	64X64
Policy	MlpPolicy	Cnnpolicy	Cnnpolicy	Mlppolicy
Model/ Env	PPO / 'Cartpole-v1'	PPO / 'Cartpole-v1'	PPO / 'Cartpole-v1'	PPO / 'Cartpole-v1'

Visualization

Reinforcement Learning without CNN



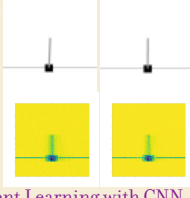
Reinforcement Learning with CNN Before VAE



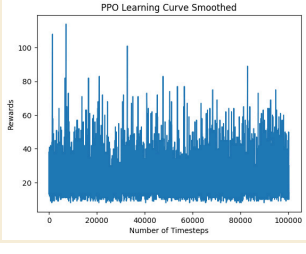
LinearVAE

Before VAE(variational autoencoder)

After VAE(variational autoencoder)



Reinforcement Learning with CNN After VAE



Results

- The research process yielded good quality output.
- Result of step 1 showed that RL is successful in the native space. The run time per steps is relatively fast and PPO consistently achieves the maximum score after learning for 100000 steps.
- In step 2, we explored how well the reinforcement learning agent performed with visual inputs and compared the learning time with native and image spaces. The learning time for the image space was significantly longer than the native space (X_{74} times more), and using a GPU was about X_3 faster than using a CPU to reach 100000 learning steps. The performance and the RL agent improves over time but it does not each the maximum possible even after 1000000steps
- In step 3, we attempted to use Linear VAE to get better results from the image space, but it did not give good output due to the absence of angles in the images.
- In the last step, we used MLP with PPO policy on the encoded data from the Linear VAE. It took a shorter time than step 2, but the performance and the RL agent does not improve over time, most likely due to the low quality representation produced by the VAE.

CONCLUSION

- It's good that you started with the PPO policy and MLP algorithm and got good results in the native space. However, when you tried to deal with the image space using a CNN, you faced challenges in improving the performance of the agent.
- So, trying out different algorithms like LinearVAE and MLP with PPO policy is a good approach. However, it seems like these attempts did not lead to significant improvements in the performance of the agent.

Next step is…

Since Reinforcement Learning is a challenging task that requires a lot of experimentation to find the best policy for the agent. These are the methods to improve

- Increase the number of training steps or episodes
- Consider other policy algorithms
- Try different hyperparameters
- Preprocess the image input
- Use transfer learning
- Use a combination of different algorithms like try with different VAE

Reference

Definition of CNN:
https://en.wikipedia.org/wiki/Convolutional_neural_network
Definition of MLP:
https://en.wikipedia.org/wiki/Multilayer_perceptron
Definition of Model-free RL, Model-based RL:
https://neptune.ai/blog/model-based-and-model-free-reinforcement-learning-pytorch-tf-case-study
Definition PPO:
https://en.wikipedia.org/wiki/Proximal_Policy_Optimization
Cartpole:
https://www.gymlibrary.dev/environments/classic_control/cart_pole/
https://gsurma.medium.com/cartpole-introduction-to-reinforcement-learning-ed0eb5b58288