Programming Assignment 1
Multi-agent PPO in Hanabi

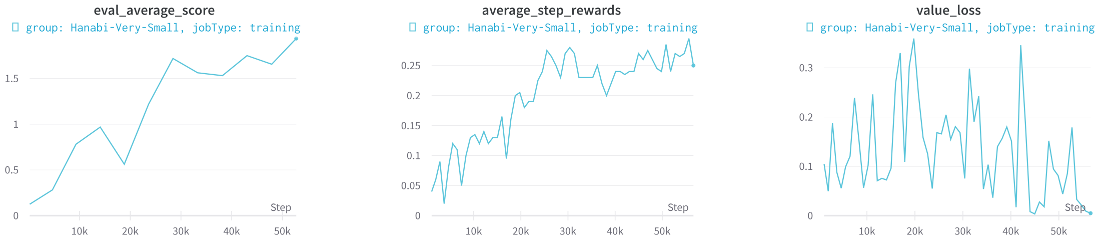
Ji In Kwak
Carnegie Mellon University
jiink@andrew.cmu.edu

October 14, 2022

1 Task 1 : Implementation of MAPPO

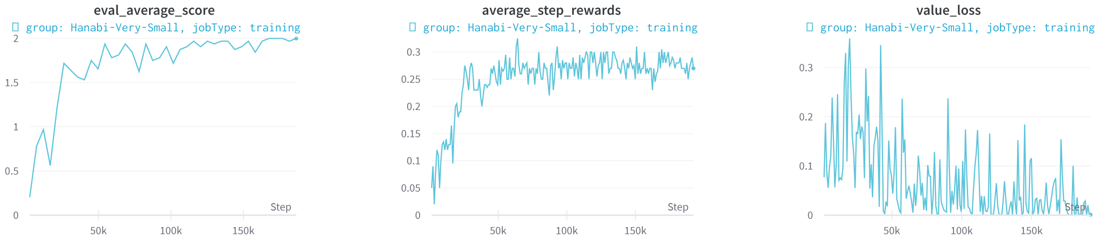
According to the MAPPO algorithm in handout, I reproduced the MAPPO algorithms in the given code. Throught ppo_update function in the code, the actor and critic policy is updated with the given learning rate and computed MSE loss value. With the fixed hyperparameters for the experiment "Hanabi-Very-Small", final evaluation accuracy score was 1.9375, and the policy ratio value was between 0.8 and 1.25 during the training. The training time took about 10 minutes. The accuracy and loss result in wandb is shown in Figure 1.

To improve the training result, the hyperparameter tuning is conducted for the number of epochs, episodes, and so on. When the number of environment step is increased from 30k to 100k, the evaluation score increases to 2.0 and the final step rewards also slightly increases to 0.27. Also, the value loss significantly decreased compared to the previous experiment, from 0.0048 to 0.0010. The reward curves and scores for every environment step is shown in Figure 2.



(a) (b) (c)

Figure 1: The result of evaluation score, rewards and policy ratio for experiment 1



(a) (b) (c)

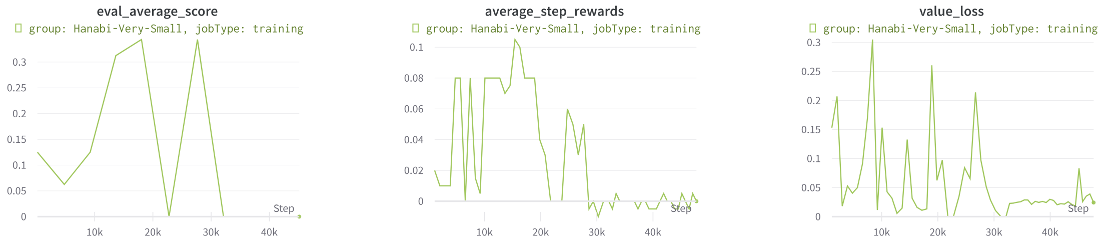
Figure 2: The result of evaluation score, rewards and policy ratio for experiment 1

2 Task 2 : PPO with a KL-penalty coefficient

In the paper, it proposed to use the KL divergence for the MAPPO, especially the adaptive KL. KL divergence is one of the methods to calculate the distance between two different distributions p

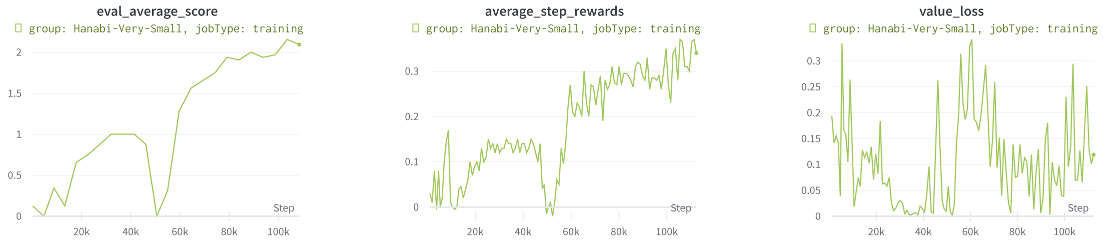
and q. By setting the β coefficient as 1, I conducted the training with KL penalty. With the given hyperparameters, the evaluation score and rewards is worse than the previous task 1, becoming the zero score while training. The results are shown in Figure 3.

Hence, to improve the PPO algorithm with KL coefficient, I changed the value of learning rate and the number of environment step. The learning rate was 7e-5 and the learning rate for critic was set as 1e-4. Also, I trained with 60k number of environment steps. With the same dataset, ”Hanabi-Very-Small”, Figure 4 shows the improved results. The final evaluation average score become above 2, which is 2.15 for the highest value. In my opinion, the smaller learning rate affects significantly to the learning process and final result. But in KL divergence loss setting, the training is unstable as we can see that the value average score and rewards become 0 at around step 50k. Hence, we need to increase the number of steps in this setting.



(a) (b) (c)

Figure 3: The result of evaluation score, rewards and policy ratio for experiment 1



(a) (b) (c)

Figure 4: The result of evaluation score, rewards and policy ratio for experiment 1