Deep Learning Lab Course: Final project

Yufeng Xiong

February 6, 2017

1 Project Purpose

Since in Deep Q-Network (DQN) of assignment 4, we used the same network to evaluate the maximal Q-value when computing the target-Q value, in this way would often overestimate the Q-values of the potential actions to take in a given state.

Thus, in the final project, I would like to implement the improved DQN (so called DDQN, Double Deep Q-Network) to see whether the proformance will be better or not in simple maze task.

2 Technical Disccusion

2.1 Deep Q-Network (DQN)

The mean square error (MSE) loss of vanilla DQN is:

$$I = (r + \gamma * \max_{a'} Q(s', a', w) - Q(s, a, w))^2$$

From the equation above we can see the vanilla DQN uses the same network to evaluate and select the action, this would often lead to overestimation of the actions, thus harm performance.

2.2 Double Deep Q-Network (DDQN)

In DDQN, a target network is proposed to remove the upward bias caused by $\max_a Q(s, a, w)$.

The author uses the primary network to chose an action, and the target network to generate the target Q-value for that action. By decoupling the action choice from the target Q-value generation, the improved DQN can substantially reduce the overestimation, and train faster and more stably.

Below is the new DDQN equation for updating the target value:

$$Q\_target = r + \gamma * Q(s', arg\max_{a'} Q(s', a', w), w^-)$$

Then the MSE loss equation becomes:

$$I = (r + \gamma * Q(s', arg\max_{a'} Q(s', a', w), w^{-}) - Q(s, a, w))^2$$

3 Implementation Details

3.1 Network architecture

In this project, I changed the Deep Q-Network a little bit compared to the network in assignment 4: adding one more convolutional layer and removing max-pooling layers. The detailed layers are:

a. convolutional layer 1
b. convolutional layer 2
c. convolutional layer 3
d. fully connected layer 1
e. fully connected layer 2

3.2 Performance comparisons

Since DDQN is supposed to train faster and more stable, I ran the tasks for four times: DQN with learning rate = 1e-4, 1e-5; DDQN with learing rate = 1e-4, 1e-5. And compared the test accuracy and average steps among all test episodes(30000 steps in total).

The comparison results are as follows:

Network type	Learning rate	Test accuracy	Average steps
DQN	1.E-04	loss explodes	-
	1.E-05	99.97%	8.8
DDQN	1.E-04	100.00%	5.7
	1.E-05	99.79%	7.5

From the table above we can see, in DDQN, if the learning rate is comparably low(like 1e-4), the network still can learn pretty well, and the accuracy is very high and the number of average steps is small. While in DQN, the network will become unstable and the loss will explode.

In conclusion, the improved DQN can achieve some kind of improvements, train faster and learn more stable.