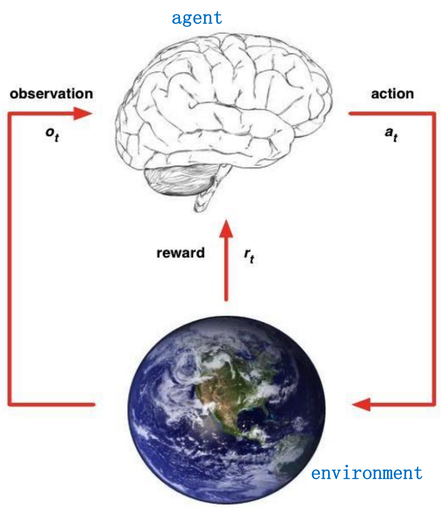
Historical Best Q-Networks for Deep Reinforcement Learning
基于历史最优Q网络的强化学习方法

Wenwu Yu, Rui Wang, Ruiying Li, Jing Gao, Xiaohui Hu
2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI). IEEE, 2018: 6-11.

联系人：俞文武　　联系方式：15652326966（wenwu2016@iscas.ac.cn)

Background

·In Reinforcement Learning (RL) an agent seeks the optimal policies, for sequential decision problems. And at each time step *t*, the agent observes state S_t, takes an action a_t and receive a scalar reward r.



Difficulties

Some features in DRL (Deep Reinforcement Learning):

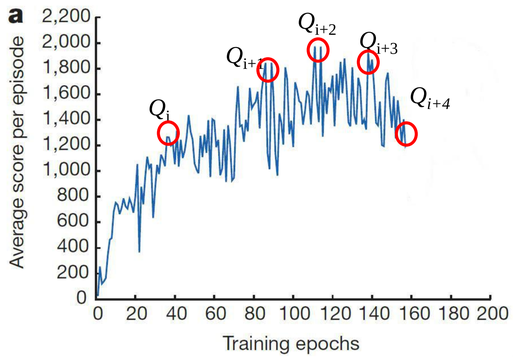
have overestimation phenomena

can not use the samples sufficiently

have one target network: updated by the latest learned Q-value estimat

Can we use the historical Q-networks to generate a new target?

using the formula: $Q_{i+4} \leftarrow \{Q_i, Q_{i+1}, Q_{i+2}, Q_{i+3}\}$



Method

This is the overview (left) of our auxiliary networks for deep learning approach. Our method, named DQN with auxiliary networks, has these networks:

·multiple target networks

·*T* latest previous target networks

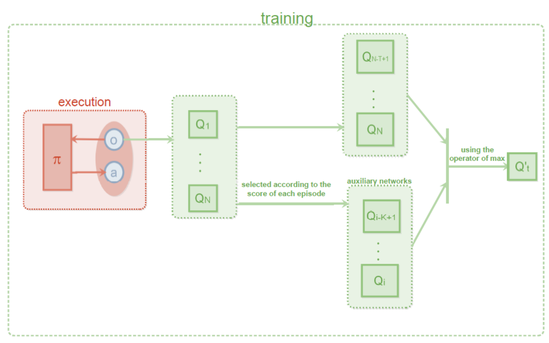
·*K* auxiliary networks

However, the criteria used to determine which network is better as an auxiliary network is indeed a problem. To overcome this issue, naturally, we adopt the following measures:

·according to the score

·using the operator of max

The whole algorithm, which we call DQN with auxiliary networks, is presented in Algorithm (right).

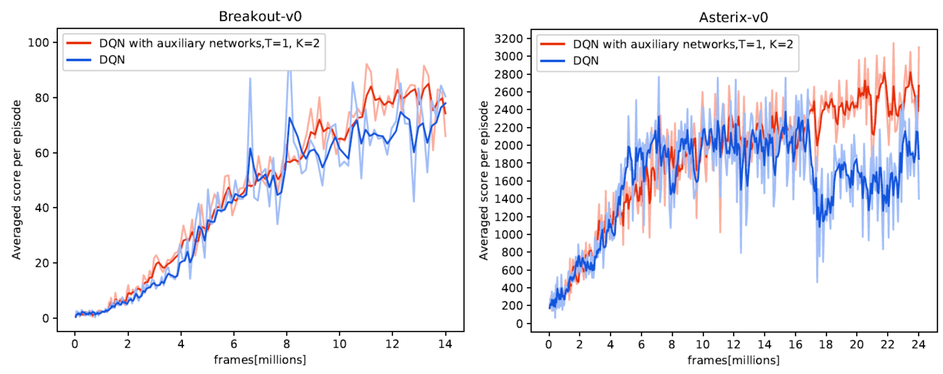


Algorithm 1 DQN with auxiliary networks

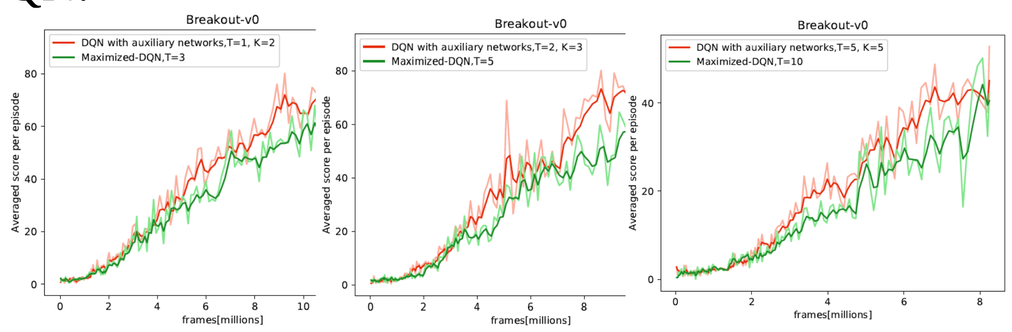
Initialize replay memory D, and buffer B
Initialize Q-value estimate Q with random weights θ
Initialize Q-value target $\hat{Q}_{1\sim T}$ with random weights $\hat{\theta}_{1\sim T}$
Initialize auxiliary networks $Q_{1\sim K}$ with random weights $\theta_{1\sim K}$
Initialize an array $r\_compare$ of 0 with a size of K
for $episode = 1, M$ **do**
  Set variable $reward = 0$
  for $t = 1, T$ **do**
    Explore(.), and execute a_t and observe reword r_t
    Store(.)
    $reward = reward + r_t$
    $Q^{new}_{target}(s', a') = max\{Q(s', a'; \hat{\theta}_1), \cdots, Q(s', a'; \hat{\theta}_T), Q(s', a'; \theta_1), \cdots, Q(s', a'; \theta_K)\}$
    $y = E_B[r + \gamma max_{a'} Q^{new}_{target}(s', a')|s, a]$
    Perform a gradient descent step with loss $||y - Q(s, a; \theta)||^2$
  end for
  Replace the earliest target network in $\hat{Q}_{1\sim T}$ by Q every C steps
  if $reward > min(r\_compare)$ **then**
    $temp = argmin(r\_compare)$
    Replace an auxiliary network with $\theta_{temp+1} \leftarrow \theta$
    Reset $r\_compare[temp] = reward$
  end if
end for

Experiment

·DQN with auxiliary networks compare with DQN:



·DQN with auxiliary networks compare with Maximized-DQN:



Conclusion

choose several historical best networks as our auxiliary networks

use the score of each episode as the criteria

demonstrate that the auxiliary networks play an important role, not the operation of maximizing