Prioritized Experience Replay

Tom Schaul, John Quan, Ioannis Antonoglou, David Silver

Summary

1) DQN/Double DQN에서 Experience Replay부분에 중점을 두고있다.

2) Prioritization에서 오는 bias는 importance sampling weight로 커버했다.

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

Motivation

기존 DQN과 Double DQN의 알고리즘에서 experience sampling부분에서
학습의 효율성을 끌어올릴 수 있는 방법이 없을까?

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

Main Idea : Experience Replay + Prioritized

Experience replay lets online reinforcement learning agents remember and reuse experiences from the past. In prior work, experience transitions were uniformly sampled from a replay memory. However, this approach simply replays transitions at the same frequency that they were originally experienced, regardless of their significance. In this paper we develop a framework for prioritizing experience, so as to replay important transitions more frequently, and therefore learn more efficiently. We use prioritized experience replay in Deep Q-Networks (DQN), a reinforcement learning algorithm that achieved human-level performance across many Atari games. DQN with prioritized experience replay achieves a new state-of-the-art, outperforming DQN with uniform replay on 41 out of 49 games.

- Agent는 특정 몇 개의 transitions로부터 더 많이 배운다.
- 즉, Experience마다 학습효과에 차이가 있다.
- 그러므로 중요한 Experience를 더 많이 뽑히게 해서 효과적으로 학습시키자!

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

Methods : Prioritized Replay

In particular, we propose to more frequently replay transitions with high expected learning progress, as measured by the magnitude of their temporal-difference (TD) error. This prioritization can lead to a loss of diversity, which we alleviate with stochastic prioritization, and introduce bias, which we correct with importance sampling. Our resulting algorithms are robust and scalable, which we

1) TD error
2) Stochastic Prioritization
3) Importance Sampling

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

3 Problems of Prioritized Replay

1) First visit에서 TD-Error가 낮으면 향후 다시 방문하기 힘들어진다.

2) Noisy Spike + Approximation Error로 학습의 불안정을 가중시킨다.
* Noisy Spike : Reward가 Stochatic한 경우에 발생하는 uncertainty를 의미

3) 지나치게 편향된 업데이트로 인해 Overfitting을 발생시킬 수 있다.

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

Stochastic Sampling Method

Stochastic Prioritization

$$P(i) = \frac{p_i^\alpha}{\sum_k p_k^\alpha}$$

0에 가까울수록 Uniform Sampling에 가까워 진다.

p_i : Transition i의 prioritization을 의미.

Proportional $p_i = |\delta_i| + \epsilon$

Rank-Based $p_i = \frac{1}{\text{rank}(i)}$

(말그대로 뭔가 우선순위 개념에 확률의 개념이 더해진 느낌이다.)

α가 커질수록 transition간의 Probability 격차는 점점 커지기 때문에

α를 잘 조절하는 것이 핵심으로 보인다.

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

Weighted Importance Sampling?

The estimation of the expected value with stochastic updates relies on those updates corresponding to the same distribution as its expectation. Prioritized replay introduces bias because it changes this distribution in an uncontrolled fashion, and therefore changes the solution that the estimates will converge to (even if the policy and state distribution are fixed). We can correct this bias by using importance-sampling (IS) weights

$$w_i = \left(\frac{1}{N} \cdot \frac{1}{P(i)}\right)^{\beta}$$

→ 이게 importance sampling term이라고?
*참고 : Mahmood et al., 2014

- Prioritization으로 생기는 bias를 통제하고
- Non-uniform sampling을 하는 것에 대한 보상개념으로 weighted Importance Sampling 사용
- 추가적으로 w_i는 안정적인 업데이트를 위해 (1 / max w_i)로 normalize시킨다.

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

Pseudo Code

Algorithm 1 Double DQN with proportional prioritization

Input: minibatch k, step-size η, replay period K and size N, exponents α and β, budget T.
Initialize replay memory $\mathcal{H} = \emptyset$, $\Delta = 0$, $p_1 = 1$
Observe S_0 and choose $A_0 \sim \pi_\theta(S_0)$
for $t = 1$ **to** T **do**
  Observe S_t, R_t, γ_t
  Store transition $(S_{t-1}, A_{t-1}, R_t, \gamma_t, S_t)$ in $\mathcal{H}$ with maximal priority $p_t = \max_{i<t} p_i$
  if $t \equiv 0 \mod K$ **then**
    for $j = 1$ **to** k **do**
      Sample transition $j \sim P(j) = p_j^\alpha / \sum_i p_i^\alpha$
      Compute importance-sampling weight $w_j = (N \cdot P(j))^{-\beta} / \max_i w_i$
      Compute TD-error $\delta_j = R_j + \gamma_j Q_{\text{target}}(S_j, \arg\max_a Q(S_j, a)) - Q(S_{j-1}, A_{j-1})$
      Update transition priority $p_j \leftarrow |\delta_j|$
      Accumulate weight-change $\Delta \leftarrow \Delta + w_j \cdot \delta_j \cdot \nabla_\theta Q(S_{j-1}, A_{j-1})$
    end for
    Update weights $\theta \leftarrow \theta + \eta \cdot \Delta$, reset $\Delta = 0$
    From time to time copy weights into target network $\theta_{\text{target}} \leftarrow \theta$
  end if
  Choose action $A_t \sim \pi_\theta(S_t)$
end for

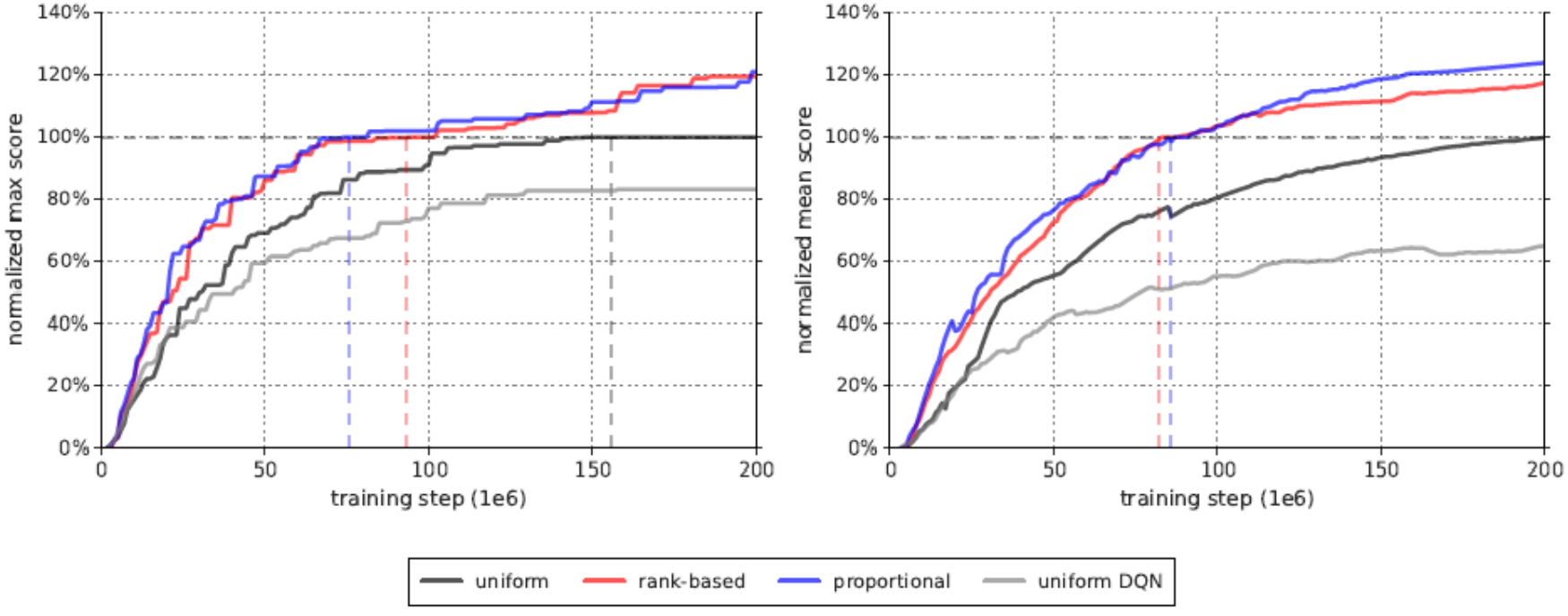
Weight normalizing

∇_θLoss sum

Update parameter θ

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

Result : 그냥 DQN, Double DQN보다 성능이 좋다.



	DQN		Double DQN (tuned)		
	baseline	rank-based	baseline	rank-based	proportional
Median	48%	106%	111%	113%	128%
Mean	122%	355%	418%	454%	551%
> baseline	–	41	–	38	42
> human	15	25	30	33	33
# games	49	49	57	57	57

Table 1: Summary of normalized scores. See Table 6 in the appendix for full results.

100만개의 Transitions

32 Batch size

모든 Experience가 8번 반복됨.

Reward와 TD error는 -1~1사이로 clipping

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

풀리지 않은 의문점

왜 저게 Importance sampling weights인가?

서로 다른 policy에 대해서 그 비율을 return/value에 곱해주는 걸로 알고 있는데..
여기 나온 term은 비율에 해당하는 거 같진 않은데..?

왜 importance sampling ratio를 곱해주면 bias를 잡을 수 있는가?

(Unbiased estimate과 bias estimate에 대한 근본적인 공부가 필요)