성찰적 강화학습: 희소 보상 환경에서의 학습

곽동현 O1 장병탁 1,2
협동과정 뇌과학 1, 컴퓨터공학부 2, 서울대학교
{dhkwak, btzhang}@bi.snu.ac.kr

Introspective Reinforcement Learning: Learning in Sparse Reward Environment

Donghyun Kwak O1 Byoung-Tak Zhang 1,2
Interdisciplinary Program in Neuroscience 1, School of Computer Science & Engineering 2, Seoul National University

요 약

최근 강화학습은 딥러닝을 함수 근사에 사용하면서 탁월한 성과를 보이고 있다. 그러나 강화학습은 문제의 난이도가 매우 높아 모든 state를 탐색하기 어렵거나, reward가 아주 희소하게 주어지는 hard exploration 상황에서는 학습이 잘 되지 않는다. 본 연구에서는 이러한 문제를 해결하는 새로운 방법으로 Inverse Reinforcement Learning을 기존의 강화학습 알고리즘에 응용한 Introspective Reinforcement Learning을 제안한다. Inverse Reinforcement Learning은 전문가의 시연으로부터 reward function를 구하는 연구분야로, 이를 강화학습에 응용하여 agent가 탐색하면서 얻은 경험으로부터 reward function를 구한다. 이렇게 찾아낸 reward function를 사용하면 실제 reward signal을 얻기 힘든 초기 탐색 상황에서도 유의미한 학습이 가능하다. 이를 Mountain Car 환경에서 기존의 강화학습 알고리즘과 성능 비교를 수행하여 제안하는 알고리즘의 학습능력을 보인다.

1. 서 론

강화학습은 최근 발전된 딥러닝 기술을 함수 근사에 사용하면서 기존에 풀기 어려웠던 문제들을 해결하는 탁월한 성과를 보이고 있다. 이러한 연구를 Deep Reinforcement Learning(이하 Deep RL)이라 부르며, Q-value function을 딥러닝으로 근사하는 Deep Q Networks라는 기본적인 형태부터[1] policy나 advantage function을 근사하는 방식의 다양한 구조와 알고리즘이 연구되고 있다[3][4].

그러나 강화학습은 문제의 난이도가 매우 높아 모든 state를 탐색하기 어렵거나, reward를 아주 희소하게 얻을 수 있는 hard exploration 상황에서 학습이 극도로 힘들어지는 문제를 갖고 있다. 이러한 문제를 해결하기 위해, pseudo-reward나 auxiliary task와 같은 방법을 통해 agent의 경험으로부터 학습할 수 있는 요소를 최대한 활용, 보다 빠른 학습 성능을 보이는 연구들이 수행되었다[2][5].

본 연구에서는 이러한 hard exploration 문제를 해결하기 위한 새로운 방법으로 Inverse Reinforcement Learning을 기존의 Reinforcement Learning과 결합한 Introspective Reinforcement Learning(이하 IntroRL)을 제안한다.

역강화학습은 환경으로부터 reward signal이 주어지지 않는 MDP\R 문제에서 전문가의 시연으로 reward 함수를 학습하는 연구 분야이다. 즉 전문가의 시연을 이용해 먼저 reward function을 알아내고, 이렇게 알아낸 reward function을 이용해 강화학습으로 agent를 학습시키는 방법이다. 이때 결과적으로 agent는 전문가의 시연을 모방하는 policy를 학습하게 되어 Learning from Demonstration 혹은 Imitation Learning이라는 연구 분야로도 확장된다[7].

이러한 역강화학습 알고리즘을 전문가의 시연이 아닌 agent가 탐색하면서 얻은 경험들로부터 reward function을 학습하여 사용할 경우, reward signal을 충분히 얻지 못하여 Q-learning 학습이 매우 더딘 초기 탐색 단계에서도 agent를 학습시킬 수 있어, hard exploration 문제를 효과적으로 다룰 수 있다. 또한 기존에 제안된 방법들과 달리 reward function을 학습하기 위해 연구된 역강화학습 알고리즘을 응용했기 때문에 보다 이론적으로 철저하며, 기존에 연구된 여러 가지 역강화학습 알고리즘들을 사용할 수 있는 장점이 있다.

본 연구에서는 이를 Mountain Car 환경에서 기존의 강화학습 알고리즘과의 성능 비교실험 수행하고, 학습 속도의 유의미한 향상을 보인다.

(loop)
1. Initial State S_t
2. Random Exploration으로 action A 선택
   혹은 A = argmax_a $Q(S_t, a)$
3. A 를 실행하여 S_{t+1} 을 탐색하고 R_{t+1} 를 얻음
4. 역강화학습을 사용해 $<S_t, A_t, R_{t+1}>$ 으로부터 subreward function R'을 학습
5. $R^*_{t+1} = R'_{t+1}(S_t, A_t) + R_{t+1}$
6. $<S_t, A_t, S_{t+1}, R^*_{t+1}>$를 이용해 Q-learning 학습
7. $S_t \leftarrow S_{t+1}$

그림 1. Q-learning with Introspective Reinforcement Learning 알고리즘

2. 알고리즘

2.1 Reinforcement Learning의 문제

강화학습은 기본적으로 agent가 환경을 탐색하면서 얻은 경험들로부터, 미래에 얻을 reward의 합에 대한 기댓값을 최대화하는 policy를 구하는 알고리즘이다. 그런데 만약 reward를 얻을 확률이 극히 낮은 문제라면 강화학습은 그 어떤 초기 학습도 이룰 수가 없고, 오히려 무작위적으로 초기화된 Q-value function 에 의한 maximization bias가 발생하여 오히려 성능을 악화시킨다[8].

2.2 Inverse Reinforcement Learning의 응용

역강화학습은 환경으로부터 받는 reward가 존재하지 않을 때, 전문가의 시연으로부터 reward function을 학습하는 연구분야이다. 가장 기본적인 방법으로는 reward function을 linear function으로 근사한 뒤, agent의 경험과 전문가의 시연 사이의 거리를 최대화하는 직선을 구하고 이를 사용하는 방법이 있다[7].

2.3 Introspective Reinforcement Learning

본 논문에서 제안하는 IntroRL은 앞서 설명한 역강화학습을 응용하여, agent가 탐색하여 얻은 경험으로부터 reward function을 알아내고, 이렇게 구한 reward function을 실제로 얻은 reward signal과 더하여 Q-learning에 사용한다[그림 1]. 또는 reward function을 오직 Q-function으로부터 policy를 계산하는 과정에만 사용하여, 보다 빠르고 좋은 exploration을 실행하는 방법도 존재한다[그림 2].

이때 사용하는 역강화학습 알고리즘은 reward function에 대한 가정에 따라 두 가지로 나뉜다. 먼저 linear reward function으로 근사한 경우, agent가 탐색한 경험을 성공한 경험(누적 reward의 합이 양수인 경우)과

(loop)
1. Initial State S_t
2. Random Exploration으로 action A 선택
   혹은 A = argmax_a { $Q(S_t, a)+R'_{t+1}(S_t, a)$ }
3. A 를 실행하여 S_{t+1} 을 탐색하고 R_{t+1} 를 얻음
4. 역강화학습을 사용해 $<S_t, A_t, R_{t+1}>$ 으로부터 subreward function R'을 학습
5. $<S_t, A_t, S_{t+1}, R_{t+1}>$를 이용해 Q-learning 학습
6. $S_t \leftarrow S_{t+1}$

그림 2. Exploration with Introspective Reinforcement Learning 알고리즘

실패한 경험(누적 reward의 합이 0이하인 경우)으로 나누고, 이 둘 사이의 거리를 최대화하는 직선을 closed-form solution으로 풀어내고 그 직선을 reward function으로 사용한다. 반면 reward function을 딥러닝으로 근사한 경우, closed-form solution이 존재하지 않으므로 다음과 같은 Loss를 정의하고 이를 최소화하는 방법으로 학습한다.

$$\sum_{fail}[-1 - R(S,A)]^2 + \sum_{success}[\ 1 - R(S,A)]^2 + \lambda \sum_{all}|R(S,A)|$$

이 Loss의 의미는 성공한 경험에 대해서는 reward function의 output의 합이 1이 되도록 Least Square Error(LSE)를 정의하고, 실패한 경험에 대해서는 -1이 되도록 LSE를 정의한 것이다. 단, 이때 반드시 regularization term을 사용해야만 하는데, 그 이유는 역강화학습 알고리즘이 본질적으로 ill-posed problem에 해당하기 때문에 적절한 regularization을 걸어주어야만 좋은 해를 찾을 수 있기 때문이다.

그런데 linear reward function을 가정하는 알고리즘의 경우, 성공한 경험과 실패한 경험 사이의 거리를 최대화하는 직선을 구하기 위해서는 성공한 경험이 최소한 1개 이상 존재해야 한다는 문제가 있다. 그러나 딥러닝 기반의 reward function은 앞서 정의한 Loss에서

표 1. 실험에서 사용한 상세 하이퍼 파라미터 설정

실험 변수	변수 값
각 에피소드 최대 시간 길이	100
최대 에피소드 반복 횟수	500
Epsilon-greedy가 줄어드는 최대 시간 길이	누적 10000 시간 스텝
Adaptive Learning Rate 알고리즘	Adam[6]
Discount Factor	0.99
Q-value function approximation	3 layer DNN

성공한 경험이 없을 경우 나머지 부분에 대해서만 Loss를 계산하여 학습하면 되기 때문에 실패한 경험만으로도 reward function 학습이 가능하다.

3. 실험

실험에서 비교로 사용한 알고리즘은 모두 [표 1]의 동일한 하이퍼 파라미터를 설정하고, 핵심적인 알고리즘 부분만 바꾸어 총 5가지 비교 실험을 수행하였다. 가장 먼저 Q-learning을 이용한 일반적인 강화학습, linear reward function을 학습하고 policy에만 적용한 exploration with linear IntroRL, deep reward function을 학습하고 policy에만 적용한 exploration with deep IntroRL, linear reward function을 학습하고 Q-learning에 적용한 Q-learning with linear IntroRL, 마지막으로 deep reward function을 학습하고 Q-learning에 적용한 Q-learning with deep IntroRL이 있다. 5가지 알고리즘에 대한 실험결과 [표 2]와 같이 Q-learning with deep IntroRL의 성능이 가장 좋았으며, linear reward function을 가정한 알고리즘은 모두 학습에 실패하였다.

Linear reward function은 가장 이론적으로 검증된 역강화학습 알고리즘임에도 불구하고 학습에 실패한 이유는 끊임없이 성공 데이터가 생성되는 것 때문으로 보인다. (기존 역강화학습에서 성공 데이터는 초기에 전부 주어지고, policy 탐색 경험만 추가되어 학습한다.) 그런데 linear function은 closed-form solution에 의해 계산되므로 새로운 성공 데이터가 추가될 때마다 큰 폭으로 직선의 각도가 변하게 되어, reward function이 쉴 새 없이 바뀌게 된다. 이를 Q-learning에 적용하면 Q-function이 불안정한 학습을 하게 되고, 결국 아이에 학습에 실패하는 것으로 보인다. 반면 deep reward function을 가정하면 gradient descent에 의해 점진적인 함수의 변화가 일어나므로 이런 문제를 겪지 않는 것으로 보인다. 단, 딥러닝을 사용할 경우 강력한 regularization을 주지 않으면(λ 0.5 이상), 학습에 실패하는 경우가 종종 발생하였다.

4. 논의 및 결론

본 연구는 선행 연구[10]와 달리 전문가의 데이터가 없는 상황에서 reward function을 학습하는 새로운 방법이다. 기존 연구가 학습된 reward와 실제 환경의 reward를 단순히 더했을 때 성능증가를 분석한 것에 비해, 이는 훨씬 더 발전된 방법으로 현존하는 모든 강화학습 알고리즘에 적용되어 성능을 상향시킬 수 있는 방법이다. 후속 연구에서는 이를 더 복잡한 이미지 기반의 게임과 실세계 문제 등에 적용하고, Loss를 수학적으로 정의하는 연구를 진행할 계획이다.

표 2. 알고리즘 비교 실험 결과

알고리즘	전체 에피소드의 평균 시간 스텝	최근 30개 에피소드의 평균 시간 스텝
Q-learning	67.960	44.644
Exploration with Linear IntroRL	100(학습 실패)	100(학습 실패)
Exploration with Deep IntroRL	60.073	42.687
Q-learning with Linear introRL	100(학습 실패)	100(학습 실패)
Q-learning with Deep introRL	53.79	39.54

위의 실험은 500번의 에피소드가 끝난 시점에서 각 알고리즘이 Mountain Car 문제를 푸는데 걸린 평균 시간 스텝을 측정함. 4-frame skipping을 사용한 경우로 전문가는 평균 30스텝으로 문제를 해결. 또한 위 비교실험은 각 알고리즘당 30회의 실험을 반복을 통해 비교적 정확한 성능을 측정함.

감사의 글

이 논문은 2017년도 정부 (미래창조과학부)의 재원으로 정보통신기술진흥센터(R0126-16-1072-SW스타랩), 한국산업기술평가관리원(10044009-HRI.MESSI, 10060086-RISF)의 지원을 일부 받았음.

참고문헌

[1] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." Nature 518.7540 (2015): 529-533.

[2] Jaderberg, Max, et al. "Reinforcement learning with unsupervised auxiliary tasks." arXiv preprint arXiv:1611.05397 (2016).

[3] Mnih, Volodymyr, et al. "Asynchronous methods for deep reinforcement learning." arXiv preprint arXiv:1602.01783 (2016).

[4] Wang, Ziyu, Nando de Freitas, and Marc Lanctot. "Dueling network architectures for deep reinforcement learning." arXiv preprint arXiv:1511.06581 (2015).

[5] Konidaris, George, and Andrew G. Barto. "Skill discovery in continuous reinforcement learning domains using skill chaining." Advances in neural information processing systems. 2009.
reinforcement learning." Icml. 2000.

[6] Kingma, Diederik, and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).

[7] Abbeel, Pieter, and Andrew Y. Ng. "Apprenticeship learning via inverse reinforcement learning." Proceedings of the twenty-first international conference on Machine learning. ACM, 2004.

[8] Sutton, Richard S., and Andrew G. Barto. Introduction to reinforcement learning. Vol. 135. Cambridge: MIT Press, 1998.

[9] Bellemare, Marc G., Joel Veness, and Michael Bowling. "Investigating Contingency Awareness Using Atari 2600 Games." AAAI. 2012.

[10] 곽동현, 이상우, 이성태, 장병탁. (2016). 모방적 강화학습. 한국정보과학회 학술발표논문집, 666-668.