강화학습을 통한 죄수의 딜레마 게임 대응전략 연구*

주정호 이석원
아주대학교 소프트웨어학과
jujeongho@ajou.ac.kr leesw@ajou.ac.kr

A Study on the Performance of the Strategies for the Prisoner's Dilemma Game using Reinforcement Learning

Jeong-Ho Ju Seok-Won Lee
Department of Software and Computer Engineering, Ajou University

요 약

게임 이론의 대표적인 예시인 죄수의 딜레마는 2명이 참가하는 비제로섬 게임의 일종으로, 정치학자 로버트 액설로드는 연속된 죄수의 딜레마 게임에서 최선의 전략을 확인하기 위해 이를 게임의 형태로 만들었다. 이때, 자주 활용되는 7가지 전략이 있는데, 그 중 현존하는 최강의 전략이라고 평가받는 것은 Copycat이다. 처음에는 협력으로 시작하지만, 그 다음부터 상대의 이전 선택을 그대로 따라하는 것이다. 본 논문에서는 현재 게임 상태, 협력 및 배신, 획득 점수를 강화학습의 State, Action, Reward로 활용하여 모델이 Copycat을 뛰어넘는 전략을 펼칠 수 있는지 연구한다. 이를 위해 간단한 Fully Connected Layer 구조와 REINFORCE 알고리즘을 이용하는 방법을 제안한다. 또한, 강화학습을 게임 이론에 적용하여 사회 과학, 경제학, 정치학 등의 문제 해결로 나아갈 수 있는 발전 가능성을 확인하고자 한다.

1. 서 론

게임 이론이란 상호 의존적이고 이성적인 의사결정에 관한 수학적 이론이다. 이는 참가자들이 상호작용하면서 변화해 가는 상황을 이해하고 분석하여 사회 과학, 경제학, 정치학 등에 적용되고 있다.

게임 이론의 대표적인 예시인 죄수의 딜레마는 2명이 참가하는 비제로섬 게임의 일종이다. 이는 공범으로 의심되는 두 명의 용의자를 따로따로 수사실로 불러 자백할 기회를 준다. '둘 다 자백하지 않으면 징역 1년(무슨 일이 있었는지 모르므로), 둘 다 서로의 죄를 자백하면 징역 3년(자백의 효과가 없으므로), 둘 중 한 명은 자백하고 다른 한 명이 자백하지 않는다면, 자백한 쪽은 석방, 자백하지 않은 쪽은 징역 10년에 처하게 된다'는 상황에서 용의자는 자백하는 것이 이득인지, 아니면 자백하지 않는 것이 이득인지 따진다. 두 사람이 각자의 이익을 위해서 이성적으로 행동한다고 가정하면, 상대방이 취하는 행동과 무관하게 자신이 자백하는 것이 이득이므로 둘 다 자백을 택하게 된다. 그 결과 둘 다 3년의 징역을 살게 된다. 즉, 각자가 최선의 이익을 보려는 행동으로 인해서 모두가 오히려 큰 손해를 본다. 반대로 두 사람 모두 공공의 이익을 위해 개인의 이익을 포기한다면 결과적으로는 두 사람 모두 적지 않은 이익을 볼 수 있다.

정치학자 로버트 액설로드는 연속된 죄수의 딜레마 게임에서 최선의 전략을 확인하기 위해 이를 게임의 형태로 만들었다. 각 참가자는 특정 전략을 취하고 있고, 대결을 하는 두 참가자의 행동에 따라 규칙을 기반으로 점수를 획득한다. 특정 라운드가 끝나면 하위 점수의 특정 참가자들이 탈락하고 상위 점수의 특정 참가자들을 재생산한다. 이를 반복하여 최선의 전략을 찾는다. [1]

표 1 죄수의 딜레마 게임의 점수 획득 규칙

	협력	배신
협력	+2 / +2	-1 / +3
배신	+3 / -1	0 / 0

본 논문에서는 위 게임에서 자주 활용되는 7가지 전략의 상대방과의 매칭에서 행동과 이에 따른 보상을 기반으로 강화학습을 통해 모델을 학습하고, 모델이 펼치는 전략을 살펴본다. 최종적으로, 강화학습을 게임 이론에 적용하여 사회 과학, 경제학, 정치학 등의 문제 해결로 나아갈 수 있는 발전 가능성을 증명하고자 한다.

표 2 죄수의 딜레마 게임의 전략

All Cooperate	항상 협력한다.
All Cheat	항상 배신한다.
Copycat	첫 번째 수는 협력으로 시작한다. 그 다음에는 상대가 바로 전 판에서 한 선택을 따라한다.
Grudger	항상 협력한다. 하지만, 상대가 한 번이라도 배신하면 끝까지 배신으로 보복한다.
Detective	협력-배신-협력-협력의 수로 시작한다. 이때 한 번이라도 상대가 배신한다면, Copycat과 똑같이 행동한다. 그렇지 않다면, All Cheat과 똑같이 행동한다.
Copykitten	첫 번째 수는 협력으로 시작한다. 상대가 두 번 연속으로 배신할 때만 배신한

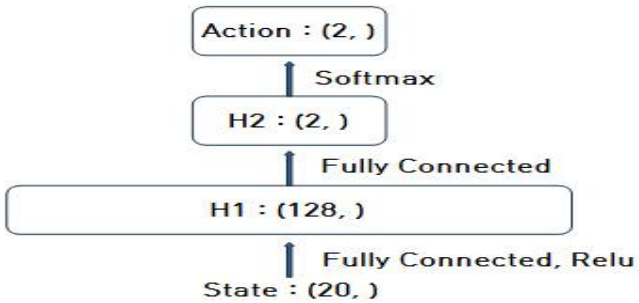
* 본 연구는 2022년 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학사업의 연구결과로 수행되었음(2022-0-01077).

	다.
Simpleton	첫 번째 수는 협력으로 시작한다. 상대가 협력하면 내 마지막 수와 같은 수를 둔다. 상대가 배신하면 내 마지막 수와 반대로 둔다.

2. 죄수의 딜레마 게임과 강화학습 연동

게임에서 각 상대방과의 매칭은 10번의 라운드로 진행된다. 이는 강화학습에서 상태(State)로 활용되며 −1, 0, 1을 이용한 배열로 나타내었다. −1은 게임 시작 전, 0은 배신, 1은 협력을 의미한다. 즉, 초기 상태는 [−1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1]이고, 첫 라운드에 모델이 배신하고 상대방이 협력하였다면 [0, 1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1, −1]이 될 것이다. 모델은 배열을 통해 게임 상태를 이해하고, 이를 기반으로 배신과 협력에 대한 기댓값을 통해 행동(Action)을 선택해야 한다. 이를 위해 그림 1과 같이 Fully Connected Layer를 이용한 간단한 구조의 모델을 제안한다.

그림 1 모델의 구조



게임이 시작되면 7가지 전략 중 무작위로 하나의 전략을 가진 상대방과 매칭이 된다. 게임 상태와 모델을 통해 행동이 선택되고, 이는 현재 라운드에서 모델의 행동을 결정할 수 있다. 또한, 이전 라운드들의 게임 상태와 상대방의 전략을 통해 현재 라운드에서 상대방의 행동을 결정할 수 있다. 이를 통해 게임 상태를 갱신할 수 있고, 다음 게임 상태가 된다. 보상(Reward)은 모델이 선택한 행동과 참가자의 행동을 기반으로 표 1과 같은 점수 획득 규칙을 통해 결정된다. 하나의 라운드는 이처럼 진행되고, 한 참가자와는 총 10번의 라운드를 진행하게 된다.

표 3 Copycat 전략의 상대방과 매칭 예시

라운드	게임 상태	모델	상대방	보상
1	[-1,-1,...]	배신	협력	+3
2	[0,1,-1,-1,...]	협력	배신	-1
3	[0,1,1,0,-1,-1,...]	협력	협력	0
...	...	...	...	...

모델의 학습은 REINFORCE 알고리즘 방식을 따른다. 이는 하나의 에피소드(Episode)를 끝까지 수행하며 각 상태에 대한 보상을 기억하였다가 에피소드가 끝난 이후에 총 보상을 계산하여 파라미터 업데이트를 실시하는 방법이다. [2, 3] 즉, 우리는 매 라운드마다 게임 상태, 모델의 행동 및 보상을 저장하고, 10번의 라운드가 종료되면 해당 에피소드의 총 보상을 계산해 파라미터 업데이트를 하게 된다.

그림 2 REINFORCE 알고리즘 Method

function REINFORCE
Initialise θ arbitrarily
for each episode $\{s_1, a_1, r_2, \ldots, s_{T-1}, a_{T-1}, r_T\} \sim \pi_\theta$ **do**
for $t = 1$ to $T - 1$ **do**
$\theta \leftarrow \theta + \alpha \nabla_\theta \log \pi_\theta(s_t, a_t) v_t$
end for
end for
return θ
end function

3. 모델이 펼치는 전략의 성능 및 기존 전략과의 비교

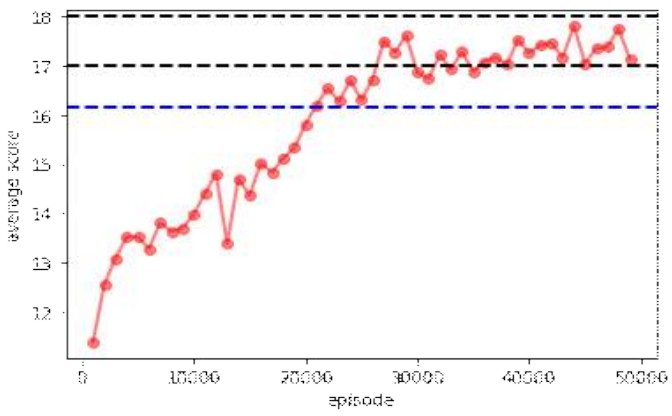
표 2에서 제시한 7가지 전략의 참가자들을 서로 매칭시켰을 때, 각 참가자가 얻게 되는 점수는 표 4와 같다. 또한, 로버트 액설로드가 현존하는 전략 중 최강의 전략이라고 제시한 Copycat이 제일 높은 점수를 기록한 것도 확인할 수 있다.

표 4 각 전략의 참가자가 획득한 점수

All Cooperate	11.5.
All Cheat	11.0
Copycat	16.16
Grudger	14.33
Detective	11.16
Copykitten	13.5
Simpleton	14.33

7가지 전략 중 하나의 전략을 가지는 참가자와 10번의 라운드를 진행하는 것을 하나의 에피소드라 하였다. 하나의 에피소드를 5만 번 시행하고, 1000개의 에피소드를 기준으로 모델이 획득한 평균 점수 추이를 살펴보았고, 그림 3과 같다. Copycat의 획득 점수인 16.16(Blue Line)을 넘어 17.80에 수렴하는 것을 확인할 수 있다.

그림 3 모델이 획득한 평균 점수



4. 모델이 펼치는 전략 분석

모델이 펼치는 전략을 분석하기 위해 7가지 전략에 대해서 어떻게 대응하고, 획득한 점수는 어떠한지 살펴보았고, 이는 표

5와 같다.

표 5 전략에 대한 모델의 대응 및 획득 점수

전략	게임 상태	점수
All Cooperate	[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1]	21/17
All Cheat	[1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]	-2/6
Copycat	[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1]	21/17
Grudger	[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1]	21/17
Detective	[1,1,1,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,1]	19/15
Copykitten	[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1]	21/17
Simpleton	[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1]	21/17

먼저, 상대의 전략에 상관없이 모델은 협력-협력으로 시작한다는 것을 알 수 있다. 이때, 상대방이 두 번째 라운드에 배신하거나 모두 배신한다면, 배신으로 대응한다. 또한, Copycat이 우수한 전략이라는 것을 증명하듯 모델도 이와 유사하게 상대방이 먼저 배신하지 않는다면 계속 협력하고, 이전 라운드에 배신한다면 배신으로 맞대응하는 모습을 보인다. 이러한 특징으로 인해 Copycat이 All Cheat에 취약한 것처럼 모델의 전략도 All Cheat에 취약하다. 마지막으로, 모델은 학습 과정에서 게임이 10번의 라운드로 구성되어있는 점을 알게 되고. 상대방의 전략과 상관없이 마지막 라운드에 무조건 배신하는 것을 확인할 수 있다.

5. 결론 및 향후 연구

본 논문에서는 강화학습을 기반으로 게임 이론의 대표적인 예시인 죄수의 딜레마에 대응하는 최선의 전략을 찾는 것을 목적으로 한다. 이를 위해 강화학습의 알고리즘 중 REINFORCE 알고리즘을 활용하고, 게임 상태를 이해하기 위해 간단한 Fully Connected Layer 구조를 사용하였다. 결과적으로, 강화학습을 통해 현존하는 최강의 전략인 Copycat을 뛰어넘는 전략을 세울 수 있다는 것을 확인하였고, 강화학습의 알고리즘과 모델의 구조에 따라 게임 상태에 대한 이해와 모델이 펼치는 전략의 성능이 더욱 향상될 수 있다는 가능성을 포함한다. 또한, 모델의 전략을 통해 참가자들이 자신의 이익을 극대화하는 방법은 결국 협력이라는 것, 게임의 마지막을 인지할 수 있으면 배신으로 게임을 마무리하는 것이 이득이라는 것을 확인하였다. 최종적으로, 강화학습을 게임 이론에 적용하여 사회 과학, 경제학, 정치학 등의 문제 해결로 나아갈 수 있는 발전 가능성을 증명하였다. 향후 본 연구에서는 다양한 강화학습의 알고리즘과 모델의 구조를 활용하여 더 나은 전략의 성능 연구를 진행하고 실제로 특정 분야에 적용하여 문제를 해결하고자 한다.

참고문헌

[1] 로버트 액설로드. 『협력의 진화』. 이경식(역). 시스테마, 2009.

[2] Williams R. J. Simple statistical gradient-following altorithms for connectionist reinforcement learning. *Mach. Learn,* 1992.

[3] Sutton R., McAllester D., Singh S. & Mansour Y. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, 2000.