CNN구조를 이용한 Starcraft II 미니게임 모방학습

정의진[1,O], 김경중[2,*]
건설환경공학과, 세종대학교[1]
융합기술원, 광주과학기술원[2]
jin.Prelude@gmail.com kjkim@gist.ac.kr

Behavior Cloning of StarCraft II Mini Games using CNN Architecture

Euijin Jeong[O,1], Kyung-Joong Kim[2,*]
Department of Civil and Environmental Engineering, Sejong University[1]
Institute of Integrated Technology, Gwangju Institute of Science and Technology[2]

요 약

강화학습에서 모방학습(behavior cloning)은 에이전트의 초반 탐험(exploration) 문제를 해소하려는 방법 중 하나로 쓰여지며, 모방학습으로 학습된 가중치를 이용해 강화학습 에이전트의 신경망을 초기화하는 방법으로 많이 사용된다. 본 논문에서는 StarCraft II 미니게임 환경에서 CNN 구조를 이용하여 모방학습을 진행할 것이다. 위 과정에서 프레임쌓기(frame stack), 오버샘플링, 드롭아웃 기법을 적용해보았고, 기법 적용의 유무에 따라 성능의 차이가 나타나는 것을 확인해본다. 실험 결과 드롭아웃의 적용이 성능을 향상시켰으며, 프레임 쌓기를 하지 않는 것이 성능향상에 도움이 되었고, 오버샘플링에 대한 성능의 변화는 크게 없어 데이터 불균형 문제에 견고함을 보였다. 또한 지도학습의 일반적인 평가지표인 loss와 accuracy 측정이 모방학습 에이전트가 게임을 얼마나 잘 플레이하느냐와는 큰 연관성이 없음을 확인하였다.

1. 서 론

심층 강화학습은 짧은 기간동안 눈부신 성장을 이뤄왔다. 2013년 딥마인드에서 DQN을 공개함을 통해 이미지와 같은 고차원의 입력이 요구되는 환경에서도 강화학습이 잘 작동함을 보였고[1], DDPG(Deep Deterministic Policy Gradient)를 통해 높은 행동 차원에서도 좋은 성능을 보이며 매우 빠르게 발전하였다[2][3]. 하지만 아직까지 여러 현실적인 문제에 직면해 있으며, 그중 하나가 탐험 문제이다. 강화학습 모델, 즉 에이전트는 더 높은 점수를 얻기 위해 최대한 많은 방법을 시도해보아야 하며, 만약 적절한 탐험이 이뤄지지 않는다면 보상을 한번도 얻지 못하고 학습이 종료될 가능성이 매우 높다. 그러므로 강화학습 탐험 문제를 다루는 많은 논문이 나왔고, 그중 비중 있게 다뤄지는 방법이 바로 모방학습이다. 모방학습이란 지도학습과 유사한 방법으로 인간 혹은 문제 해결을 위해 참고할 수 있는 다양한 자료의 정책 또는 행동을 모방하는 학습 방법으로써, 강화학습의 경우 에이전트에게 인간이 문제를 풀어나가는 방법을 모사하도록 모방학습을 먼저 학습시키고 강화학습을 적용하여 일종의 가이드라인을 제시하는 용도로 모방학습을 이용하게 된다. 본 논문에서는 CNN 구조의 모델을 이용하여 Starcraft II 미니게임을 모방학습 시켜 볼 것이다. 그리고 프레임 쌓기, 오버샘플링, 드롭아웃 세가지 기법의 적용에 따라 성능이 어떻게 변화하는지 알아볼 것이다.

2. 실험 환경

논문에서 쓰일 Starcraft II DefeatRoaches라는 미니게임은 7~9개 사이에서 랜덤으로 생성되는 마린 케릭터를 이용하여 네마리의 바퀴(적군) 케릭터를 이기는 방식으로 진행된다. 마린 캐릭터 하나가 죽을 때마다 보상 -1을 받으며, 10의 보상을 받는다. 그리고 바퀴 네 마리를 모두 이기게 되면 게임이 계속 이어지게 되는데, 이때는 마린 캐릭터 개수가 처음 시작 개수보다 하나 줄어든 채로 게임이 진행되고 하나의 에피소드로 간주한다. 예를 들어 첫 번째 게임에서 이기고 21점을 얻고 두 번째 판에서 져서 11점을 받게 된다면 해당 에피소드는 하나로 보고 총 32점의 보상을 얻게 된다. 액션은 이동과 공격으로 이루어져 있으며, Observation은 게임 플레이 화면의 RGB 값이다.

그림 1. 미니게임 플레이 화면

3. 모델과 학습

신경망은 [4]에서 소개된 FullyConvNet의 구조를 따른다. 화면(데이터)을 입력으로 받게 되면 합성곱 신경망 네 개를 거치게 되고, 이때 인풋 화면의 크기를 보존하기 위해 padding을 입혀준다. 그리고 마지막으로 채널이 1인 합성곱 신경망을 거쳐 좌표값을 근사하도록 하고, 한편으로는 또다시 여러 합성곱 신경망을 거치되 MaxPooling을 지나며 차원을 축소시킨 후, 신경망을 거치도록 하여 행동을 결정한다. 모델의 인풋은 192x256 크기의 RGB 화면을 흑백으로 전처리한 입력이 들어가게 되고, 프레임 쌓기 기법의 경우 시간순으로 4 스텝의 프레임이 쌓여 4x192x256 크기의 입력이 들어가게 된다. 그리고 3가지 액션, no_op, 이동, 공격을 출력한다. 총 3251개의 프레임 데이터를 모았다.

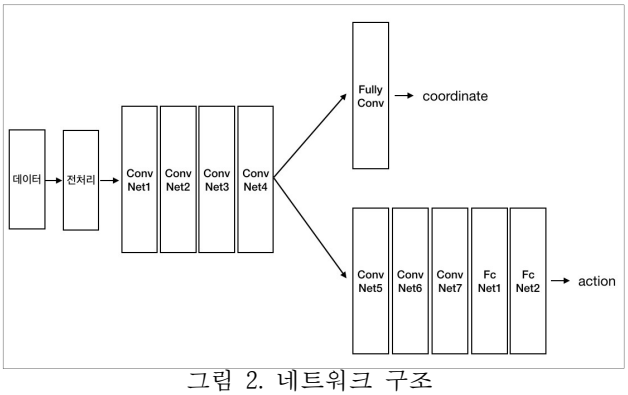

그림 2. 네트워크 구조

no_op이라는 액션은 아무 행동도 수행하지 않는다는 명령으로 따로 좌표값이 필요하지 않은 행동이다. 이 경우에는 좌표값에 대한 loss는 따로 구하지 않고 행동에 대한 loss만을 계산해 역전파를 하도록 하였다. 나머지 두 행동(움직이기, 공격하기)은 행동과 좌표값 두 가지에 대한 loss를 모두 구한 다음 더하여 같이 역전파 시켜준다.

4. 적용 기법

모델 학습에 적용할 첫 번째 기법은 프레임 쌓기로, 에이전트에게 현재의 게임 상황만을 보여주는 방법과 이전 프레임 3개을를 같이 보여주는 방법을 비교하여 어떤 방법이 모방학습의 성능향상에 어떤 영향을 미치는지 분석한다. 두 번째는 균형 잡힌 데이터에 따른 차이이다. 게임 플레이 데이터의 경우 불균형 데이터 문제(imbalanced data)로 나타나는 경향이 잦은데, 지도학습의 경우 인위적으로 소수의 데이터 개수를 늘리는 오버 샘플링(over-sampling)기법으로 모델이 학습 시 모든 데이터를 같은 비율로 학습할 수 있도록 해준다[6]. 불균형 데이터 실험에서는 이러한 방법이 게임 모방학습의 성능평가에 어떤 영향을 미치는지 분석한다. 마지막으로 드롭아웃의 영향력을 분석한다. 드롭아웃은 지도학습에서 과적합을 막기 위한 기법으로 제시된 알고리즘인데[5], 모방학습 데이터의 과적합을 줄이는 방법이 test 데이터의 loss만이 아닌 실제 학습된 네트워크의 게임 플레이 성능향상에도 영향을 미치는지 분석한다.

데이터는 약 4천 프레임의 미니게임 플레이를 통하여 직접 생성하였는데, 원래 24fps인 게임을 1프레임 단위로 액션을 추출하다 보니, 행동을 하지 않는 no_op의 비율이 압도적으로 높아졌다. 이로 인한 데이터 불균형을 조금이나마 해소하고자 8프레임마다 행동을 선택하고 기록하였다.

결과 분석은 loss, 액션 선택 정확도, 좌표 선택 정확도, 그리고 실제 게임 스코어 총 4가지를 통해 분석하게 된다. 정확도는 총 데이터셋 액션의 개수 중 맞춘 액션의 개수 비율을 나타낸 것이고, 좌표 선택 정확도는 액션 개수 중 맞춘 좌표값의 비율을 말한다. 모델은 매 에폭(epoch)마다 loss와 두 정확도와 같이 저장되고, 스코어의 경우 저장된 모델마다 5번 게임을 하게 한 다음 평균 스코어를 기록하는 방식으로 신뢰성을 높였다.

5. 실험 결과

프레임 쌓기의 경우 일반적으로 지도학습에서 학습 정도의 척도로 삼는 loss, accuracy를 보았을 때는 프레임을 쌓은 모델이 확연히 좋은 결과를 보였지만, 게임 플레이 스코어를 보았을 때는 프레임을 쌓지 않은 모델이 더 좋은 성능을 내는 것을 볼 수 있었다.

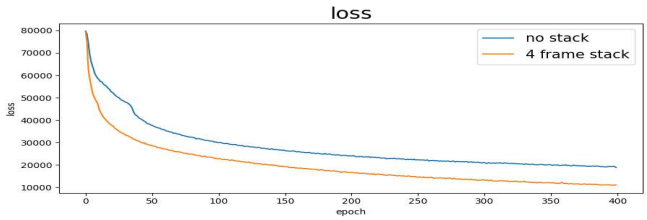

그림 3. 프레임 쌓기 loss 그래프

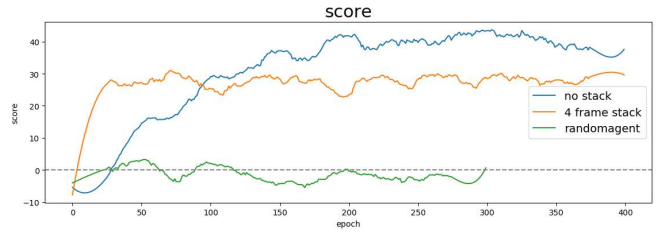

그림 4. 프레임 쌓기 스코어 그래프

데이터 오버샘플링 후 학습을 한 모델과 오버샘플링을 거치지 않고 학습을 한 모델(불균형 데이터)과의 비교를 하였고, 이때 공평한 비교를 위해 오버샘플링을 거치지 않은 모델도 랜덤하게 데이터 샘플을 복제하여 오버샘플링을 거친 모델의 데이터 개수와 맞추어 주었다. 그렇게 실험을 진행한 결과 불균형 데이터로 학습한 모델이 loss도 더 낮은 위치에서 시작하고 액션 정확도도 더 빠르게 올라갔는데, 이는 no_op 데이터의 개수가 나머지 선택지보다 확연히 많은 만큼 좌표값 loss가 포함되는 비중도 적고 데이터 불균형에 의해 정확도도 빨리 높아지기 때문에 불균형 데이터 문제에서 loss와 정확도는 좋은 성능평가 지표로 보기 어렵다(그림 5).

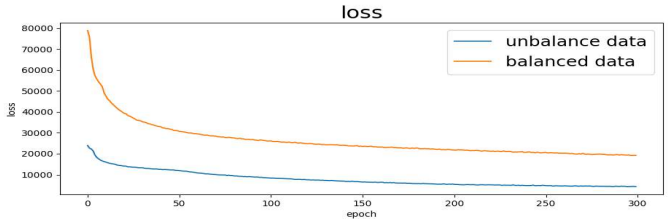

그림 5. 불균형 데이터 loss 그래프

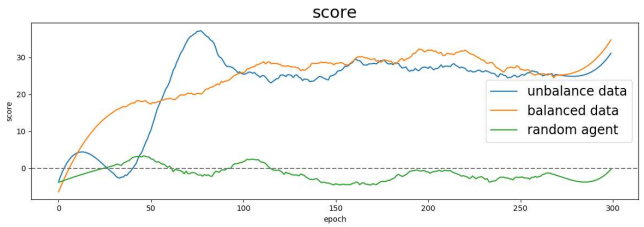

그림 6. 불균형 데이터 스코어 그래프

그러나 게임 스코어를 비교해본 결과 불균형 데이터와 오버샘플링 데이터 모델간의 성능차이가 크지 않음을 확인하였다. 잠시 불균형 데이터 모델이 오버샘플링 데이터 모델의 성능을 뛰어넘는 구간도 있었지만 두 모델 모두 최종 스코어는 비슷한 수치에서 머물렀다(그림 6).

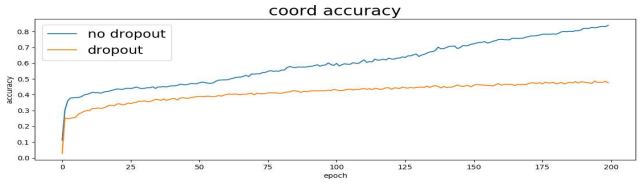

그림 7. 드롭아웃 좌표 정확도 그래프

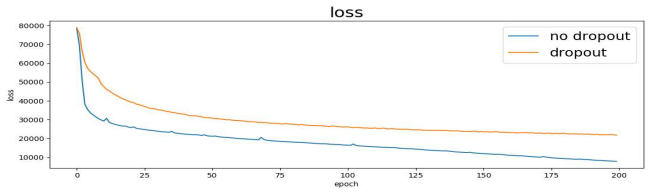

그림 8. 드롭아웃 loss 그래프

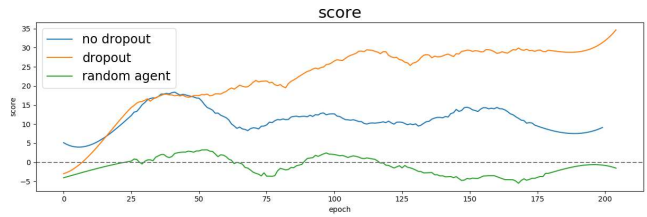

그림 9. 드롭아웃 스코어 그래프

마지막으로 드롭아웃을 적용한 모델과 그렇지 않은 모델과의 결과를 비교하였다. 드롭아웃 미적용 시 학습 데이터에 과적합 되기 쉬운 만큼 loss와 accuracy 측정치는 미적용 모델이 확실히 높았다(그림 7, 8). 하지만 스코어 그래프를 비교하면 드롭아웃을 적용한 모델이 두 배 이상 좋은 성능을 내는 것을 볼 수 있다(그림 9).

6. 결과 분석

위 실험을 통해 지도학습의 일반적인 성능평가지표로 잘 사용되는 loss와 accuracy만으로는 모방학습 모델의 성능을 보장할 수 없다는 것을 눈에 띄게 확인할 수 있었다. 이러한 현상은 특히 프레임 쌓기와 드롭아웃 실험에서 크게 나타났으며, 드롭아웃의 경우 loss와 accuracy 부문에서 적용 모델이 미적용 모델보다 못미치는 수치가 기록됐음에도 불구하고 스코어에 있어서는 미적용 모델을 2~3배 차이로 앞지르는 결과를 내었다. 프레임 쌓기 실험에 있어서는 드롭아웃과 대조적으로 프레임 쌓기 기법이 loss와 accuracy 부분에서는 좋은 결과를 보였음에도 스코어에 있어서는 프레임 쌓기를 하지 않은 모델보다 뒤처지는 모습을 보였다.

그리고 불균형 데이터 실험을 진행한 결과 불균형 데이터가 모방학습에 끼치는 영향은 그렇게 크지 않음을 볼 수 있었다. 데이터 자체가 불균형 문제를 염두에 두고 8프레임 단위로 액션을 선택했다는 사실이 영향을 미쳤을 수도 있으므로, 임의로 가공되지 않은 데이터로 학습을 진행했을 시에도 위와 같은 결과를 얻을 수 있는지에 대한 추가 연구가 필요해 보인다. 하지만 가공한 데이터 또한 적지 않은 불균형 문제를 가지고 있다는 사실도 무시할 수 없는 만큼(그림 10) 모방학습이 불균형 데이터에 있어서 꽤 견고한 모습을 보임을 확인할 수 있었다.

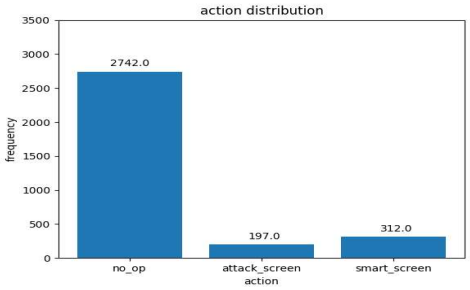

그림 10. 리플레이 액션 분포

그리고 마지막으로 드롭아웃 실험을 통해 모델의 과적합을 막는 것이 실제 모방학습 모델의 성능을 향상시키는데 큰 기여를 함을 알 수 있고, 마찬가지로 드롭아웃 시험의 경우도 마찬가지로 loss와 accuracy를 통해서는 드롭아웃의 효과를 확인하기도 어려움을 확인하였다.

7. 감사의 글

본 연구는 문화체육관광부 및 한국콘텐츠진흥원의 출연금 등으로 수행하고 있는 2019년도 문화기술연구개발 지원사업으로 수행되었습니다.

본 연구 논문은 문화체육관광부 및 한국콘텐츠진흥원의 출연금 등으로 수행하고 있는 한국전자통신 연구원의 R2019020067 위탁연구과제의 연구결과입니다.

* 교신저자

참고문헌

[1] Minh et al, Playing Atari with Deep Reinforcement learning, 2013

[2] Silver et al, Deterministic Policy Gradient Algorithms, 2014

[3] Lillicrap et al, Continuous control with deep reinforcement learning, 2015

[4] Vinyals et al, StarCraft II: A New Challenge for Reinforcement Learning, 2017

[5] Srivastava et al, Dropout: A Simple Way to Prevent Neural Networks from Overfitting, 2014

[6] Johnson et al, Survey on deep learning with class imbalance, 2019