인접 행렬 추상화를 이용한 DGN 모델의 스타크래프트 멀티 에이전트 강화 학습

윤성훈○ 김경중*
세종대학교 컴퓨터공학과
광주과학기술원 융합기술학제학부
kiboyz2@naver.com, kjkim@gist.ac.kr

Multi-agents Reinforcement Learning of DGN using abstraction of adjacency matrix for StarCraft

Seonghun Yoon○ Kyung-Joong Kim*
Department of Computer Science and Engineering, Sejong University
School of Integrated Technology, GIST

요 약

전투 시뮬레이션 게임인 스타크래프트에서는 다양한 유닛들과 이를 통해 적을 이기는 것을 목적으로 한다. 이러한 시뮬레이션 환경에서 여러 에이전트를 움직이는 것을 결정하는 것은 매우 어려운 일이다. 여러 에이전트들의 의사결정을 내리는데 있어서 필요한 것은 각 에이전트 간의 관계와 그로 인해 생기는 충돌과 같은 간섭효과가 발생 시 생기는 문제등을 효과적으로 해결할 수 있는 협동 전략을 배우는 것이다.

본 논문에서는 여러 특징을 지닌 각기 다른 특성의 에이전트들의 관계를 그래프 네트워크를 통해 추상화하여 스타크래프트와 같은 복잡한 게임에서 실시간으로 이루어지는 의사결정을 통해 에이전트들의 협동 전략을 학습하는 강화 학습 모델을 제안한다. 기존의 그래프 네트워크를 이용한 강화 학습 연구들은 인접 행렬을 형성하는데 있어 에이전트들간의 거리만을 고려하였다면[1], 본 연구에서 제안하는 방법은 환경 안에서 받아오는 정보를 모델을 통해 인접 행렬을 구성하여 학습을 진행한다. 실험 결과 제안하는 모델의 성능은 기존의 그래프 네트워크의 방법과 비교하였을 때 더 높은 성능을 보여주었다.

1. 서 론

스타크래프트는 블리자드에서 1998년 제작된 실시간 전략 시뮬레이션 게임(RTS Game)으로 대중적으로 많은 인기를 얻은 게임이다. 알파고와 이세돌의 경기로 인해 인공지능과 딥러닝에 대한 사람들의 관심이 많아짐에 따라, 바둑보다 더 복잡한 상황을 가지고 있는 스타크래프트로 많은 연구자들이 관심을 가지고 있으며, 그로 인해 스타크래프트는 연구자들이 많이 선택하는 환경이다. 실시간 전략 게임은 바둑과는 달리 수행할 수 있는 행동이 크고, 상대방의 정보를 완전히 알지 못하는 불완전 정보게임이며, 실시간으로 의사결정을 다양한 유닛들에게 내릴 수 있어, 바둑과 같은 보드 게임보다 복잡하여 사람의 수준에 해당하는 인공지능을 만드는데 큰 어려움을 가지고 있다.

스타크래프트에서 승부를 가르는 가장 큰 요소 중 하나는 유닛들의 컨트롤이다. 스타크래프트의 유닛은 각 종족마다 세분화되어 있으며, 각 플레이어는 플레이어 당 최대 200개의 개별 유닛에게 실시간으로 명령을 내려야 한다. 이러한 상황에서 상대방 유닛의 조합과 내 유닛간의 관계를 통하여 각 유닛의 움직임에 대한 다양한 명령들을 실시간으로 내려야 하는만큼 변수가 존재하며, 이 변수가 플레이어가 플레이를 하는데 복잡한 상황을 만들기 때문에 멀티 에이전트 환경에서 사람의 수준에 해당하는 인공지능을 만드는 것을 더욱 어렵게 한다.

이를 극복하기 위해, 본 연구에서는 그래프 네트워크를 통해 각 유닛간의 관계를 추상화한 정보로 사용하여 나타낸 인접 행렬 기반의 그래프 네트워크 강화 학습을 제안하였다. 기존의 관련 연구의 경우, 거리를 기반으로 한 인접 행렬을 만들어 사용하였지만, 일반적으로 거리를 기반으로 한 인접 행렬의 경우 환경에 따라 한계가 있는 방법이다. 따라서, 본 논문에서는 스타크래프트의 각 유닛들을 그래프의 노드로 나타내었을 때, 각 유닛간의 특징을 기반으로 한 인접행렬 추상화를 사용한 모델을 사용하였다. 본 연구에서는 스타크래프트 게임 환경을 미니 게임으로 축소시켜 연구를 진행하였다. 이 게임 환경에서는 거리 기반 인접 행렬의 단점과 여러 유닛의 협력을 보여주기 위해, 유닛의 종류를 다양화하였다. 비교 실험 결과, 거리 기반으로 인접 행렬을 표현한 모델보다 본 논문에서 제안하는 각 유닛간의 특징을 기반으로 한 인접 행렬 추상화를 사용한 모델의 성능이 더 좋은 성능을 보여주었다.

2. 관련 연구

2.1 그래프 합성곱 신경망(Graph Convolutional Networks)

그래프 합성곱 신경망(Graph Convolutional Networks)은 데이터 내에서 연결 네트워크를 구성해 이를 입력데이터로

활용하는 인공신경망 기법이다.

그래프는 각 노드 간의 관계를 나타내는 N × N 인접행렬(Adjacency Matrix)과 각 노드를 구성하는 특징을 나타내는 N × F 특징행렬(Feature Matrix)로 구성된다.(N은 노드의 수, F는 특징의 수를 의미한다). 데이터에서 인접행렬을 추출해 이를 그래프 합성곱 신경망을 통해 노드 간 관계를 추출하는 것이 가능하다.

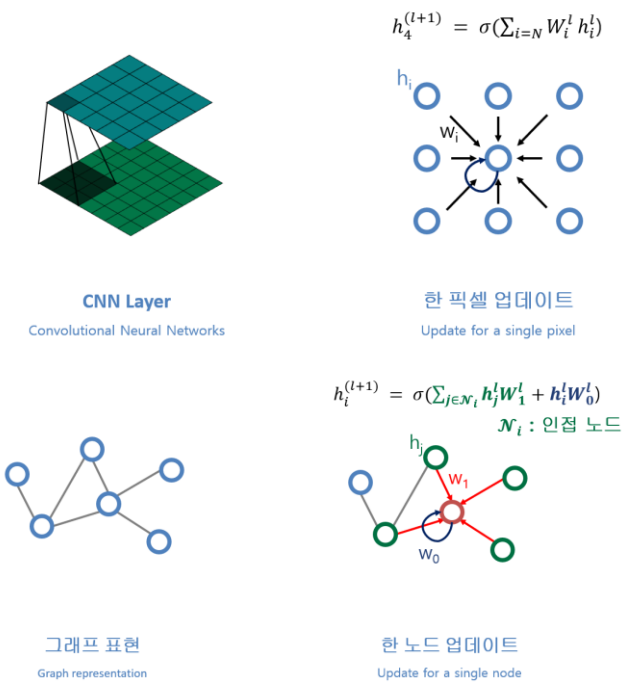

그림 1 합성곱 신경망(CNN)과 그래프 합성곱 신경망(GCN)

그림 1은 합성곱 신경망(CNN)과 그래프 합성곱 신경망(GCN)의 비교를 나타낸 것이다. 그래프를 이용한 환경에서 그래프 합성곱 신경망은 다양한 연구들이 있지만, Jiechuan Jiang et al.은 그래프 합성곱 신경망을 이용한 강화 학습 모델을 제안하였다.[2]

본 연구에서는 강화 학습 모델에서 유닛 간의 특징을 기반으로한 인접 행렬을 활용한 인접 행렬 추상화 강화 학습 모델을 제안한다.

3. 스타크래프트를 위한 그래프 네트워크 강화 학습

제안한 모델이 학습할 환경은 스타크래프트를 축소시킨 미니 게임이다. 미니 게임의 환경 구성과 데이터는 다음과 같이 처리한다.

3.1. 데이터 전처리

스타크래프트를 학습시키기에 앞서, 스타크래프트에서 넘어오는 데이터들을 전처리하는 과정이 필요하다. 이를 위해, 스타크래프트의 환경 데이터를 벡터화 시키는 과정을 진행한다. 총 9개의 요소를 가진 벡터형식의 데이터로, 그 구성은 탑승 여부(1), 지형(1), 공중 유닛 여부(1), 플레이어 소유 여부(1), x 좌표(1), y 좌표(1), 각 유닛과의 거리 정보(3)이다.

3.2 스타크래프트 미니 게임

스타크래프트 미니 게임은 적과 서로 제한된 환경 내에서 목표를 달성하면 보상을 주는 환경이며 제한된 시간 내에 적을 모두 섬멸하는 것을 목적으로 한다.

유닛 간의 관계를 나타내는 것이 목적이므로, 서로 다른 3개의 유닛을 통해 유닛이 관계를 학습하는지의 여부를 판단할 것이다. 그림 2에서 각 유닛은 공중 수송 유닛(노란색), 지상 공격 유닛(파란색), 적 유닛(빨간색)의 3가지의 유닛으로 구성되며, 각 유닛의 관계를 고려하여 지상 유닛이 도달할 수 없는 언덕에 위치한 적 유닛을 공중 수송 유닛을 통하여 언덕에 올라가 처치하는 것을 목적으로 한다.

환경의 보상은 매 지상 유닛이 아래 언덕에 위치해 있을 경우, 스텝 당 -0.001의 보상을 주며, 지상 유닛이 공중 수송 유닛에 탑승 시 스텝 당 0의 보상을, 지상 유닛이 언덕 위에 있을 경우 스텝 당 +0.001의 보상을 주도록 되어 있다. 적 유닛을 처치 시 +1의 보상을, 60초가 지났을 경우에는 -1의 보상을 주며 환경을 초기화 시킨다.

각 유닛은 네 방향의 행동과 정지 행동을 할 수 있으며, 수송 유닛의 경우, 유닛을 태우거나 내리는 행동을 할 수 있다.

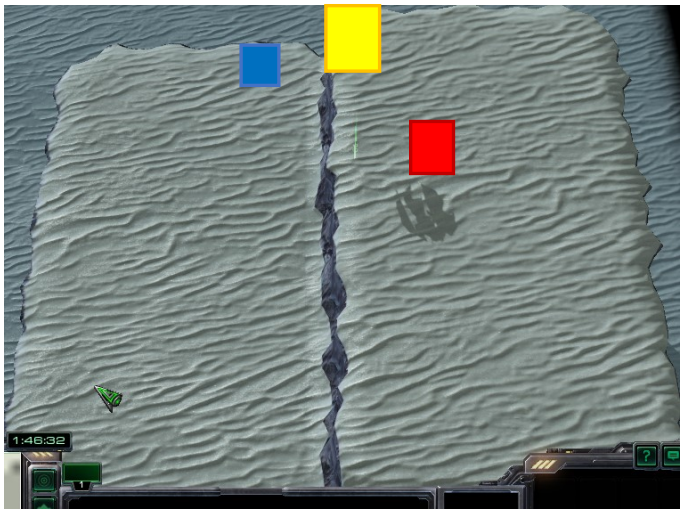

그림 2 스타크래프트 미니게임 화면

3.3 인접 행렬 추상화 그래프 네트워크 강화 학습

그래프 네트워크 강화 학습은 데이터 전처리 과정에서 설명한 각 유닛 별 9개의 요소를 가진 벡터를 생성한 후, 이를 이용하여 특징 벡터(F)와 인접 행렬(C)을 구성하기 위한 인공 신경망의 입력 값으로 사용한다.

인접 행렬을 추상화하기 위하여, 한 유닛의 특징 벡터를 제외한 나머지 유닛들의 특징 벡터를 입력 값으로 한 뒤, 제외된 유닛의 특징벡터를 나머지 유닛들의 특징 벡터와 결합하여 이를 이용하여 각 유닛 별 인접 행렬을 생성한다.

인공 신경망의 출력 값으로 나온 각 에이전트의 인접행렬(C)은 encoder를 거쳐 특징 벡터(F)와 관계 행렬을 구성하며, 이를 Graph Convolution Layer[1]를 통하여 각 에이전트의 그래프 관계를 출력한다. 그림 3은 제안하는 인접 행렬 추상화를 사용한 그래프 네트워크 강화 학습의 모델 구조이며, 학습을 진행할 시에는 강화 학습을 위한 업데이트 모델과 타겟 모델 두 개의 네트워크를 생성한 후 학습을 진행한다. 일정 스텝마다 학습되고 있는 네트워크의 가중치를 복사한 후, Soft Target update를 수행하여 네트워크의 가중치를 그대로 복사하지 않고 tau 값만큼의 비율만 반영하여 가중치를 타겟 네트워크에 업데이트 하는 지수 평균 이동을 사용한다.

$$\theta' \leftarrow \tau\theta + (1-\tau)\theta'$$

네트워크에서 출력된 Q는 DDDQN(Dueling Double DQN)을 사용하여, 각 노드의 Q 값의 MSE를 측정하여, 학습을 진행한다.

$$Q(s,a;\alpha,\beta) = V(s;\theta,\beta) + \left(A(s,a;\theta,\alpha) - \frac{1}{|A|}\sum_{a'} A(s,a';\theta,\alpha)\right)$$

$$\text{Loss}^{DDDQN} = \left\| y_j - Q(s,a;\theta) \right\|^2$$

3. 실험 결과

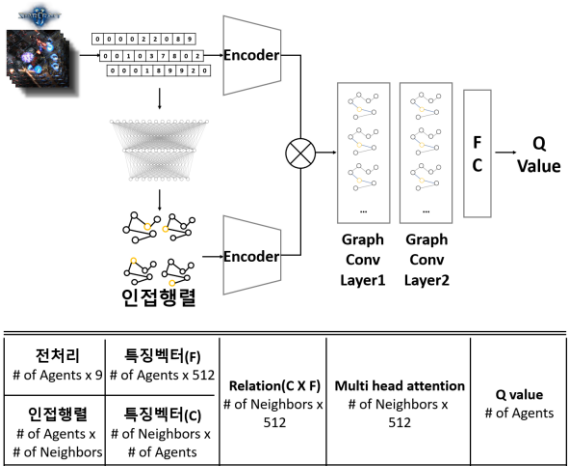

그림 3 인접 행렬 추상화를 사용한 그래프 네트워크 강화 학습 모델 구조

스타크래프트 미니 게임은 위에서 제시된 환경에서 다음과 같은 두 가지 방식을 비교 실험 진행을 한다. 거리 기반 인접 행렬 그래프 강화 학습을 통해 학습하는 방법과 제안하는 모델을 통해 인접 행렬을 생성하여 학습을 하는 두 가지 방법을 비교한다.

3.1 실험 평가

실험은 학습 데이터를 learning rate는 0.0001, target update step은 10000, tau는 0.01로 학습한 것이다. 그림 4를 보면 인접 행렬 그래프의 보상 변화를 에피소드 별로 확인할 수 있다. X축은 시간에 따른 에피소드이며, Y축은 환경에서 받는 보상의 합이다. 파란색의 그래프가 인접 행렬 추상화를 사용한 제안된 모델의 성능이며, 자주색 그래프는 거리 기반 인접 행렬을 사용한 모델의 보상 그래프이다. 파란색 그래프의 보상이 더 높은 것으로 보아 제안된 인접 행렬 추상화 모델의 성능이 더 뛰어나다는 것을 확인할 수 있다. 즉, 에이전트 별 관계를 고려하여 유닛 간 협동을 학습한 것을 의미한다.

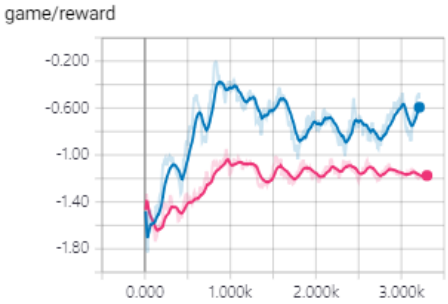

그림 4 학습 과정에서 에피소드 별 보상 변화량

4. 결론 및 향후 연구

현재 연구의 실험 환경은 직접 제작한 스타크래프트의 미니게임 환경으로 진행하였다. 기존 그래프 네트워크 기반 강화 학습의 연구에서 사용한 거리 기반 인접 행렬 그래프 네트워크 강화 학습 모델이 아닌, 환경 안에서 받아오는 정보를 모델을 통해 인접 행렬을 구성하는 모델을 사용하였다. 인접 행렬 추상화를 사용한 제안된 모델의 성능은 기존의 거리 기반 인접 행렬 강화 학습과 비교하였을 때, 좀 더 높은 성능을 보여주었다.

그래프 네트워크 강화 학습은 에이전트 별 관계를 고려하여 유닛 간 관계를 통해 협동을 학습하는 것을 목적으로 한다. 추후, 좀 더 복잡한 환경에서 유닛 간의 협동을 학습하는 모델을 시도할 예정이며, 이를 위해 더 복잡한 미니 게임 환경과 모델 구조 개선이 필요할 것으로 보인다.

5. 감사의 글

이 논문은 2017년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초 연구 사업임 (2017R1A2B4002164). * 교신 저자

참 고 문 헌

[1] Jiechuan Jinang, Chen Dun, Zongqing Lu, “Graph Convolutional Reinforcement Learning for Multi-Agent Cooperation,” *Arxiv*, 2019

[2] Wang, Ziyu, et al. "Dueling network architectures for deep reinforcement learning." arXiv preprint arXiv:1511.06581 (2015)

[3] Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep reinforcement learning with double q-learning." Thirtieth AAAI Conference on Artificial Intelligence. 2016.