

강화 학습을 활용한 크리티컬 노드 탐색

이태홍, 오형국, 노영태

한국에너지공과대학교, 국방과학연구소, 한양대학교

etehong@kentech.ac.kr, hyungkook_oh@add.re.kr, youngtaenoh@hanyang.ac.kr

Critical Nodes Detection with Reinforcement Learning

TaeHong Lee, Hyungkook Oh, Youngtae Noh

요약

본 논문은 기존의 Closeness Centrality, Betweenness Centrality 그리고 PageRank를 통하여 얻어낸 노드 중요도 지표를 활용한 노드 제거와 심층 강화학습 (Deep Reinforcement Learning)을 통하여 얻어낸 노드 중요도 지표를 활용한 노드 제거를 통해 그래프의 연결성을 얼마나 빠르게 약화시키는 지 조사함으로써 기존의 중요도 지표와 심층 강화학습 중요도 지표의 성능을 비교하여 심층 강화학습이 네트워크 분해에 중요한 영향을 주는 노드 (Critical node) 탐지에서 얼마나 우수한 성능을 보여줄 수 있는지 조사한다.

I. 서론

그래프 내에서 노드들은 각각의 구조에 따라 다양한 기능을 수행할 수 있으며, 이로 인해 각 노드들의 중요도는 서로 다르다 할 수 있다. 노드의 중요도를 평가하기 위한 Degree centrality, Closeness centrality, Betweenness centrality, PageRank centrality [1], Percolation centrality 등 다양한 기준이 존재하며, 이러한 기준을 바탕으로 중요도가 높은 노드, 즉 Critical 노드를 식별할 수 있다. Critical 노드는 그래프 내 다른 노드들에 비해 상대적으로 중요도가 높기 때문에, 이러한 노드들을 제거함으로써 그래프의 구조를 효과적으로 분해하거나 약화시킬 수 있다. 그러나 그래프를 가장 크게 손상시킬 수 있는 최적의 노드 집합을 찾는 문제는 일반적으로 NP-hard 문제로 분류되며 다항 시간 복잡도로 해결하기 어려워, 대부분의 기존 기술은 높은 계산 비용으로 인해 대규모 그래프에 대해서는 비실용적이다. 최근에는 그래프 신경망(GNN)과 그래프 컨볼루션 네트워크(GCN)의 등장으로 이러한 문제 해결에 큰 도움이 되고 있다. GNN과 GCN은 그래프를 효과적으로 인코딩할 수 있으며, 이러한 인코딩을 통해 생성된 저차원 벡터를 활용하여 중요도를 평가하면 계산 비용을 상당하게 절감할 수 있다. 임베딩된 벡터에 심층 강화학습을 적용함으로써 최적의 크리티컬 노드 집합을 찾는데 효과적인 알고리즘을 개발하는데 용이하며, 기존 방식보다 훨씬 효율적이며 실제 대규모 그래프 환경에서 적용 가능하다.

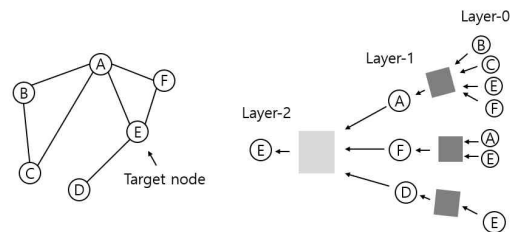
II. 본론

2-1 노드 임베딩 (Node embedding)

노드 임베딩의 경우 Shallow embedding과 Deep embedding으로 나눌 수 있으며 Shallow embedding의 대표적인 예로는 Deepwalk와 node2vec 알고리즘이 있다. Deepwalk의 경우 그래프상의 Random walk를 통해 노드 시퀀스를 생성한 후 이들의 유사도가 원래 그래프의 유사도와 비슷하도록 학습하며, Node2vec의 경우 Random walk를 진행할 때 너비 우선 탐색(BFS)과 깊이 우선 탐색(DFS) 사이에서 균형을 잡을 수 있도록 하여 국소적인 구조와 전역적인 구조 모두를 반영하게 만든 알고리즘이다. 그러나 이러한 Shallow embedding의 경우 각각의 노드에 고유한

임베딩 벡터를 배정하므로, 학습되지 않은 새로운 그래프에 적용하기 힘들다는 단점이 있다. 추가적으로, 두 노드의 구조적 유사성 (Structural similarity)이 높음에도 불구하고 Random walk에 의해 선택되지 않거나 두 노드간에 경로가 존재하지 않는 경우, 두 노드의 임베딩이 상이하게 학습될 수 있어 구조적 유사성을 제대로 포착한다고 하기 힘들다. 또한, 이 방식들은 노드의 Feature를 활용하지 않기 때문에 Feature 정보를 놓치게 된다는 단점이 있으며, 더 나아가 Shallow embedding은 입력벡터에 순서에 직접적인 영향을 받기 때문에 치환 불변 (Permutation Invariance) 특징을 가지고 있는 그래프에 사용하기에 제약이 많다. 그에 비해 GCN에 사용되는 Deep embedding 접근 방식은 CNN(Convolutional Neural Network)의 구조를 이용하여 이웃 노드들의 정보를 통합하여 임베딩 벡터를 생성한다. 이 방식은 새로운 그래프에 적용하기 용이하고 구조적 유사성을 효과적으로 포착하며, 그래프의 치환 불변성 (Permutation Invariance) 특징을 보존하는 장점을 가진다. 따라서 그래프 노드 임베딩을 수행할 때 Shallow embedding을 사용하는 것 보다 GCN과 같은 Deep embedding 기법을 적용하는 것이 효과적이다.

본 논문에서는 여러 가지 GCN 방식 중 하나인 GraphSage [2]를 사용한다.



[그림 1] 그래프 구조에 따른 노드 임베딩의 구조

2-2 심층 강화학습 (Deep Reinforcement Learning)

강화학습 (Reinforcement Learning)이란 에이전트 (Agent)가 환경 (Environment)과의 상호작용을 통해 보상을 얻고, 이 보상을 최대화 하기 위해 학습 알고리즘을 사용하는 것을 의미한다. 강화학습에서 핵심적인 Off-policy 알고리즘의 예로, 행동정책 (Behavior policy)에는 greedy-policy, 타겟 정책 (Target policy)에는 ϵ -greedy policy를 사

용하는 Q-learning이 있다. 심층 신경망을 사용하지 않은 상태에서 Q-learning은 Q-Table을 아래 수식과 같은 방식으로 업데이트를 점진적으로 하며 결과적으로 Q-value는 참값으로 수렴함이 알려져 있다.

$$Q^{new}(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha(R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t))$$

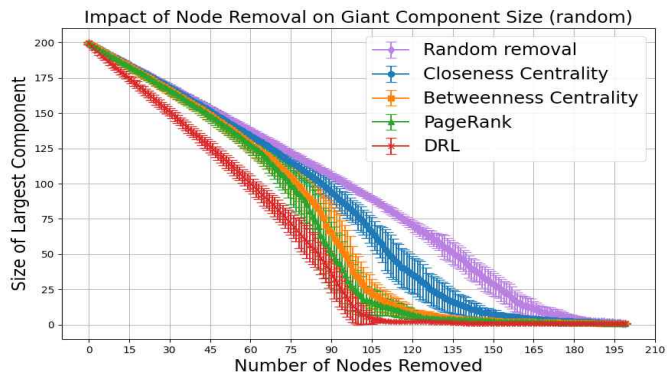
그러나 상태공간과 행동공간이 매우 커질 경우 Q-table을 저장하는 공간을 확보하기 어려워지며, 더 나아가 상태공간 또는 행동공간이 연속적인 무한집합이 될 경우 더 이상 Q-table을 사용 할 수 없다. 상태공간, 행동공간의 차원이 커짐에 따라 발생하는 문제점들은 심층 신경망을 이용하여 해결할 수 있다. 2015년 구글의 딥마인드사는 Q-learning 알고리즘에 CNN을 적용하여 상태공간의 차원을 줄이는 방식으로 작동하는 DQN[3]을 제안하였다. 본 논문에서는 Multistep DQN을 통하여 Critical Node를 찾도록 학습시켰다. 노드 임베딩을 통하여 얻은 벡터를 상태공간(State space)으로 활용하였고 행동공간(Action space)의 경우 제거할 Critical node를 선택하는 것으로 택하였다. Reward의 경우 [4]를 참고하여 일단 ANC(Accumulated normalized connectivity)를 다음과 같이 정의한다.

$$R(v_1, v_2, \dots, v_N) = \frac{\sum_{k=1}^N \sigma(G \setminus \{v_1, v_2, \dots, v_k\})}{\sigma(G)} \quad (\text{여기서 } G \setminus \{v_1, v_2, \dots, v_k\} \text{는 원래의 그래프 } G \text{에서 } v_1, v_2, \dots, v_k \text{를 제거하고 남은 그래프를 의미하며 } \sigma: \{G\} \rightarrow R^+ \text{는 그래프 집합에서 양수 집합으로 가는 함수로서 그래프의 연결성(Giant Component Size 또는 Pairwise connectivity)를 나타내는 함수이다.)}$$

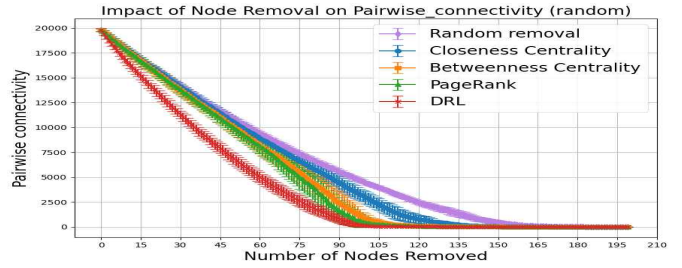
Reward는 ANC의 감소량으로 설정한다. 본 논문에서는 기존에 가장 널리 쓰이는 노드 중요도인 Closeness centrality, Betweenness centrality, 그리고 PageRANK와 비교하여 심층 강화학습 알고리즘의 성능을 비교한다. 비교를 위해 노드들을 중요도 순으로 하나씩 제거한 후 Giant component size와 Pairwise connectivity를 비교하여 그래프 전체의 연결성을 비교해 각 노드들의 중요도에 따른 노드 제거가 그래프를 얼마나 빨리 악화시키는지 비교한다. Giant Component Size란 그래프 상에서 가장 큰 연결 성분을 의미하며 Pairwise Connectivity란 그래프 상의 서로다른 노드 u, v에 대해 경로가 존재한다면 1, 그렇지 않다면 0을 더한다. 두 지표 모두 값이 클수록 그래프의 연결성이 높다고 할 수 있다.

비교에 사용될 그래프는 노드가 33개인 소규모 그래프인 Zachary's Karate club[5], 대규모 그래프로 노드의 개수 200개, 연결 확률 p=0.03인 에르되시 레니 랜덤 그래프, 노드의 개수 150개, 추가 연결선수 m = 2인 바라바시 알버트 그래프이다.

2-3 비교 결과 (에르되시 레니 랜덤 그래프)



[그림 2] 노드 제거에 따른 Giant Component Size의 변화



[그림 3] 노드 제거에 따른 Pairwise connectivity의 변화

III. 결론

본 논문에서 비교해본 결과 대규모 에르되시 레니 랜덤 그래프에서 DRL의 성능이 눈에 띄게 좋았으며 바라바시 알버트 그래프의 경우 근소하게 DRL의 성능이 다른 중요도에 비해 우수하였다. 소규모 그래프인 Zachary's Karate club의 경우 Giant Component Size에서는 가장 우수한 성능을 보여줬으나 Pairwise connectivity에서는 Betweenness Centrality와 Page rank에 근소하게 성능이 낮게 측정되었다. 그러나 Zachary's Karate club은 노드의 개수가 33개인 소규모 그래프이며 규모가 큰 그래프인 에르되시 레니 랜덤 그래프와 바라바시 알버트 그래프의 경우 확실히 성능이 우수한 점으로 미루어 보았을 때 대규모 그래프에서 DRL의 성능은 Closeness Centrality, Betweenness Centrality, PageRank보다 우수하다고 할 수 있으며 앞으로 DRL을 활용한 Critical 노드 탐색은 그 잠재력이 충분하다고 보여진다. 추후 연구를 통해 대규모 그래프상의 Critical 노드 탐색 연구는 복잡한 그래프 구조의 이해를 돕는데 기여할 것으로 기대된다.

ACKNOWLEDGMENT

이 논문은 방위사업청의 재원으로 국방과학연구소의 지원을 받아 수행된 연구임 (411JJ5-912967201)

참 고 문 헌

- [1] Brin, Sergey, and Lawrence Page. "The anatomy of a large-scale hypertextual web search engine." Computer networks and ISDN systems 30.1-7 (1998): 107-117.
- [2] Hamilton, Will, Zhitao Ying, and Jure Leskovec. "Inductive representation learning on large graphs." Advances in neural information processing systems 30 (2017).
- [3] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." nature 518.7540 (2015): 529-533.
- [4] Fan, Changjun, et al. "Finding key players in complex networks through deep reinforcement learning." Nature machine intelligence 2.6 (2020): 317-324.
- [5] Zachary, Wayne W. "An information flow model for conflict and fission in small groups." Journal of anthropological research 33.4 (1977): 452-473.