

Controlling Graph Dynamics with Reinforcement Learning and Graph Neural Networks

Abs.

- 그래프에서 제한적인 interventions에 의해 부분적으로 관측된 dynamic process를 컨트롤 하는 문제를 고려해본다.
- 예를 들어, 바이러스가 퍼지는 것을 막기 위해 검사를 스케줄링 하는 경우, 제품 광고를 위해 타겟 마케팅을 하는 경우, 그리고 social networks에서 가짜 뉴스가 퍼지는 것을 막기 위해 포스트를 검사하는 경우 등의 경우들이다.
- 논문에서는 이러한 경우를 "a sequential decision problem over a temporal graph process"로 설정한다.
- 두 가지 문제에 대하여 성공적으로 이 framework를 적용시켰다.
 1. epidemic을 막기 위해 꼭 검사를 받아야하는 노드(사람)
 2. 그래프에서 영향력이 큰 노드(사람)

Intro.

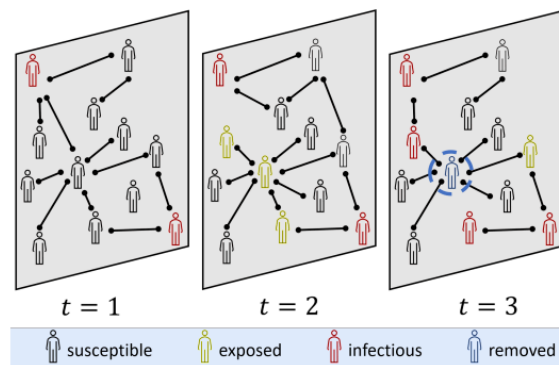


Figure 1. A viral infection process on a graph and an intervention aimed to stop its spread. Here, graph nodes represent people and edges represent interactions. At $t = 1$ only two people are infected (red). At $t = 2$ several interactions resulted in new *exposed* people (yellow); At $t = 3$ the blue node was selected to be quarantined to stop the viral spread. This paper presents a general framework for learning how to control such dynamic processes on graphs.

- 사람들 속에서 epidemic spreading하는 상황을 가정해보면, 질병을 가지고 퍼져나가는 것을 막기 위해서 infected carriers를 찾고 격리시키는 것이 중요하다. 하지만 모든 사람들을 한꺼번에 테스트하는 것은 불가능 하다.
- 이런 경우 제한된 자원에서 가장 epidemic carriers에 가까운 사람 먼저 찾아 내야한다.
- 또 다른 상황을 가정해보면, 제품을 광고하여 이 정보가 social graph에서 퍼져 나가길 원한다고 해보자.
- 만약 impactful node가 확정된다면, 이 노드는 자신의 의견을 다른 노드에 영향을 끼칠 것이다. 즉 information diffusion이 발생한다.
- 이러한 경우 agent가 작은 수의 노드들의 states를 modify하는 interventions를 이용하여 시스템의 dynamics를 핸들링할 수 있게된다.
예를 들어, 특정 감염된 node를 자가 격리 시키거나, 타겟 twitter에게 쿠폰을 보내는 등.
- 그러나 가장 어려운 부분은 current state는 완전히 observed된 것이 아니라는 것이다. 예를 들어 우리는 그래프에서 모든 노드들의 ground truth infection status를 알지 못 한다.
- 그래프는 $G(t) = (V, E(t))$ 로 나타내고 시간에 따라 구조가 변화한다.

- 매 state마다 agent는 nodes의 subset을 고르고 약간의 변화를 주게 된다.
- 목표는 각 state마다 노드의 개수에 달려있는 objective를 최소화 하는 것이다. 예를 들어 가장 영향력이 있는 node를 찾아서 프로모션을 하고 싶다고 하면, agent는 set of seed nodes를 선택하고 그 노드에게 정보를 주거나 광고를 하는 등의 영향을 끼칠 것이다. 만약 이 노드들이 영향력이 있는 노드라면 정보가 퍼져나갈 것이고 이런 경우 목적은 영향을 받은 노드의 수를 최소화 하는 것이다.
- localized interventions를 사용하여 시스템의 dynamics를 컨트롤 하는 문제는 매우 어려운데, 3가지 이유가 있다.
 1. 복잡한 dependencies를 가지면서 연속적으로 변화하는 환경에서 decision making을 해야한다.
 2. 이 문제를 풀기 위해서는 영향을 받은 특정 노드의 잠재적인 downstream ripple effect를 측정할 수 있어야한다. 그리고 node가 실제로 영향을 받을 확률과 균형을 맞춰야한다.
 3. 모델은 noise와 partial observability를 다룰 수 있어야한다.
- 최근의 방법들은 크게 두 가지로 나뉘어 진다.
 1. Monte Carlo simulation
이 방법은 각 decision의 utility를 평가한다. 이 방법은 작은 사이즈의 그래프에서 좋은 성능을 보이지만 큰 규모의 그래프에서는 그렇지 않다.
 2. Heuristics based on topological properties of the known graph
degree가 큰 node는 영향력이 크다고 직관적으로 생각할 수 있을 것이다. 물론 이런 방법은 매우 큰 그래프에서 적용이 잘 될지 모르겠지만, sub-optimal하다.
- 논문에서는
 1. controlling a diffusive process on a temporally evolving graph => a partially-observed Markov decision process (POMDP)
 2. Formulate the problem of selecting a subset of nodes for dynamical intervention as a ranking problem
 3. design an actor-critic RL algorithm to solve it.
- 이 방법의 메인 challenge는 dynamics가 직접적이고 완전하게 관측되지 않는다는 것이다. 대신, 몇몇 노드들의 상태에 대한 부분적 정보가 각 point time 별로 주어진다.

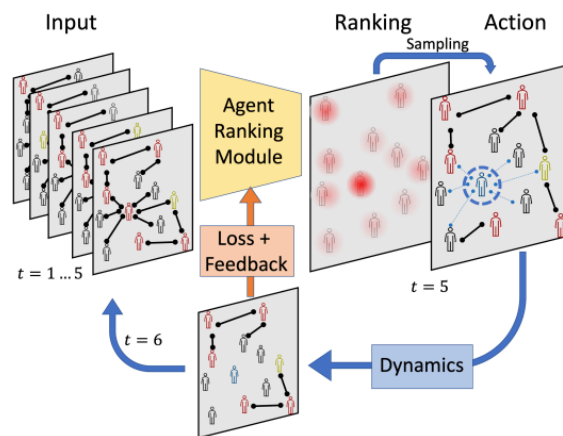


Figure 4. Schematic of our approach. The Ranking module receives as input a sequence of graphs and outputs scores over nodes. Scores are then used to sample actions, selecting nodes for intervention. Here, the person circled in blue is selected for quarantine and its connections are canceled (dashed blue lines). The downstream effect on epidemic progression is then fed as a loss to the ranking module.