Northeastern University
Network Science Institute

POTION: Optimizing Graph Structure for Targeted Diffusion

Sixie Yu, **Leo Torres** (leo@leotrs.com), Scott Alfeld, Tina Eliassi-Rad, Yevgeniy Vorobeychik

Yu, Torres, et al. **POTION: Optimizing Graph Structure for Targeted Diffusion.** SIAM SDM'21.
https://bit.ly/3wXInFw

Abstract

Controlling diffusion processes on networks has many applications. For example, an adversary may wish to attack a pre-specified **targeted subgraph** of the network, while limiting the impact on the rest of the network.

We present, **POTION**, a model in which the principal aim is to **optimize graph structure to achieve such targeted attacks**.

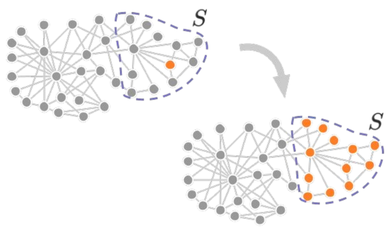
Our algorithm, **POTION-ALG**, solves this problem at scale, using a gradient-based approach that leverages Rayleigh quotients and pseudospectrum theory.

Targeted Diffusion in Cybersecurity

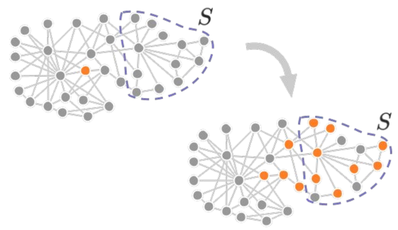
In a cybersecurity setting, an attacker wants to release a computer virus over a computer network G. **There exists a subgraph S of G that the attacker wants to harm with the virus.** S may consist of a set of high-profile targets within the computer network. The virus spreads following SIR dynamics. The attacker may manipulate the weights of edges in G. Let $\mathbf{A}$ be the adjacency matrix of G, and let $\tilde{\mathbf{A}}$ be the adjacency matrix after the attacker has manipulated some weights. The attacker has four objectives.

Attacker Objectives

If the virus starts in S, it should create an epidemic: The attacker wants to decrease the epidemic threshold of SIR dynamics in S. This is approximated by $1/\lambda(\mathbf{A})$, so the attacker wants to **maximize** $\lambda(\mathbf{A})$.



If the virus starts out of S, it should reach S: The attacker wants to **maximize** $\phi(S)$, **the normalized cut between S and $G \setminus S$.**



Attacker Objectives (cont'd)

Limit the harm to $G \setminus S$: We argue that maximizing the likelihood of infection of nodes in $G \setminus S$ is equivalent to **maximizing $\sigma(S)$, the sum of eigenvector centralities of nodes in S.**

Remain within a budget: The budget is given in terms of the **difference in the eigenvalues between the original and attacked matrices:**

$$|\lambda_i(\tilde{\mathbf{A}}) - \lambda_i(\mathbf{A})| \leq \epsilon,\ i = 1, \dots$$

POTION's Objective Function

The attacker's objective function contains all four objectives plus an additional restriction that **guarantees that the solution is a valid adjacency matrix.**

$$\max_{\tilde{\mathbf{A}}} \quad \alpha_1 \lambda_1(\tilde{\mathbf{A}}_S) + \alpha_2 \sigma(\mathcal{S}) + \alpha_3 \phi(\mathcal{S})$$

$$s.t. \quad \tilde{\mathbf{A}} \in \mathcal{P} = \left\{ \tilde{\mathbf{A}} \;\middle|\; \begin{array}{l} |\lambda_i(\tilde{\mathbf{A}}) - \lambda_i(\mathbf{A})| \leq \epsilon,\ i = 1, \dots, n, \\ \tilde{\mathbf{A}} = \tilde{\mathbf{A}}^\top, \tilde{\mathbf{A}}_{ii} = 0,\ \forall i = 1, \dots, n \end{array} \right\}$$

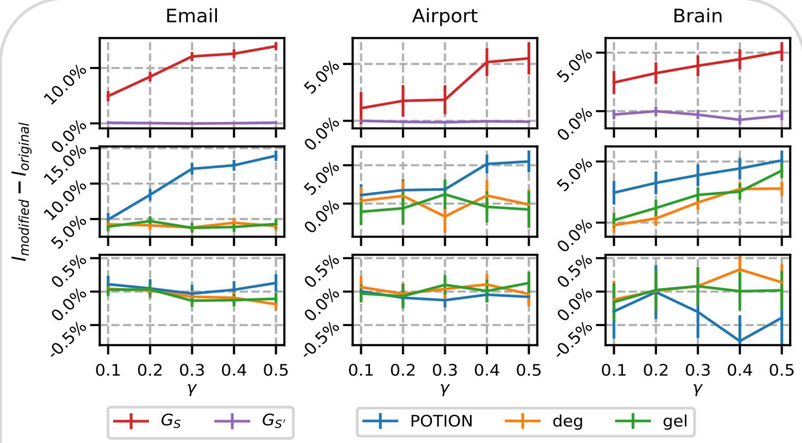
POTION-ALG Algorithm

Our algorithm exploits **Rayleigh quotients and pseudospectrum theory** to express the attacker's optimization problem as a **differentiable function of $\tilde{\mathbf{A}}$**. Once in this form, the differentiation can be handled by standard packages such as PyTorch.

Input: Adjacency matrix $\boldsymbol{A}$, error tolerance ϵ, schedule of step sizes $\{\eta_i\}_i$.
Output: Modified adjacency matrix $\tilde{\boldsymbol{A}}$

i $\leftarrow 1$, $\tilde{\boldsymbol{A}}_1 \leftarrow \boldsymbol{A}$, $B_i \leftarrow 0$
while *True* **do**
  $\boldsymbol{\Delta}_i \leftarrow$ the gradient of $\alpha_1 \lambda_1(\tilde{\boldsymbol{A}}_S) + \alpha_2 \sigma(\mathcal{S}) + \alpha_3 \phi(\mathcal{S})$ w.r.t. to $\tilde{\boldsymbol{A}}_i$
  diag $(\boldsymbol{\Delta}_i) \leftarrow \mathbf{0}$
  if $\|\boldsymbol{\Delta}_i\| = \mathbf{0}$ **then**
    return $\tilde{\boldsymbol{A}}_i$
  if $B_i + \|\eta_i \boldsymbol{\Delta}_i\|_2 \leq \epsilon$ **then**
    $\tilde{\boldsymbol{A}}_{i+1} = \tilde{\boldsymbol{A}}_i + \eta_i \boldsymbol{\Delta}_i$, $B_{i+1} = B_i + \|\eta_i \boldsymbol{\Delta}_i\|_2$, $i = i + 1$
  else
    return $\tilde{\boldsymbol{A}}_i$

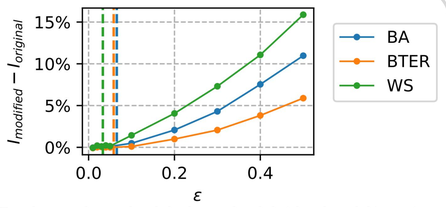
Experimental Results



Top: Difference in number of infected before and after applying **POTION** in S (**red**) and in $G \setminus S$ (**purple**). Higher is better. **Middle:** Comparison of **POTION** (**blue**) against baselines (**orange** and **green**) in S. Higher is better. **Bottom:** Comparison against baselines in $G \setminus S$. Lower is better.

Certified Robustness

Theorem. If the attacker's budget ϵ is lower than a certain threshold, the attack will be unsuccessful.



Budget threshold threshold (dashed lines) and the attack's effectiveness (increase in final number of infected, in solid lines) in three different random graph models.

Conclusions and future work

1. Is the problem of targeted diffusion relevant for other settings different than cybersecurity? **In epidemiology,** the targeted subgraph may be a group of vulnerable individuals and the objective would be to protect them rather than target them.

2. We have focused on modifying the epidemic threshold, which neglects the history of spread as it happens. Can we design and develop **algorithms that modify the structure of the network in real time,** as the dynamics unfolds?

Partially supported by the National Science Foundation (IIS-1741197) and by the Combat Capabilities Development Command Army Research Laboratory (under Cooperative Agreement Number W911NF-13-2-0045).