Efficient Influence Maximization in Signed Networks with Positive Influence Increasing and Negative Influence Decreasing Simultaneously via Edge-view Influence Estimation

Fu-Kai Chang[1], Shiou-Chi Li[1,2], and Jen-Wei Huang[1]

[1] Department of Electrical Engineering,
National Cheng Kung University, Tainan, Taiwan
omnom.chang@gmail.com, jwhuang@mail.ncku.edu.tw
[2] Institute of Computer and Communication Engineering,
Department of Electrical Engineering,
National Cheng Kung University, Tainan, Taiwan
sp822543@gmail.com

Abstract. Influence maximization in signed social networks is one of the influence maximization problems that involve competitive positive influence and negative influence. Most works focus on either positive influence maximization or negative influence blocking problems. Some works take both positive influence and negative influence into consideration, but their proposed methods take a lot of time. We proposed a time-saving framework, Edge-view Signed Influence Maximization (ESIM). ESIM estimates the power of influence flow on edges to make a balance between positive influence maximization and negative influence blocking. The experiment shows that our framework is not only efficient but also effective on real world signed social networks.

Keywords: Influence Maximization · Edge-view Influence · Signed Network · Polarity.

1 Introduction

The influence spread in real-world social networks is complicated. In addition to positive influence which relates to promotion or correct information, there exists negative influence which relates to discouragement or rumor. Some works [2,6,7] focus on influence blocking problems to minimize the spread of negative influence. Moreover, the received influence might be distorted due to the friend-foe or trust-distrust relationship in social networks. A signed network is a network where relationships between vertices are represented by positive and negative signs. While the positive edge maintains the sign of influence, the negative edge inverts it. There are some works that focus on positive influence maximization in signed networks [1,4,5]. However, the negative influence is not considered by these works. In a signed network, it is possible that the way we maximize the positive influence could lead to huge amount of negative influence. On the other hand, when we try to block negative influence, positive influence might decrease simultaneously.

Sung *et al.* [9] proposed Sign-aware Competitive Independent Cascaded (SCIC) model with polarity dominance mechanism and proved that the greedy method does not provide an approximation-guaranteed solution to their proposed objective function, which contains positive and negative influence. Though the authors showed the effectiveness of S-CMIA in terms of the proposed objective function, the proposed algorithm is time-consuming and not applicable to large social networks. Therefore, we aim to propose an efficient and effective framework that tackles binary competitive influence in signed networks.

Previous works usually calculate scores of vertices that represent how influential they are to create viral cascades. They usually apply breadth-first search for evaluation, which turns out to be time-consuming when they need to obtain marginal gain of vertices during seed selection. None of them take edge-view perspective to see influence spread as a kind of flow that gives edges influence power. In this work, we propose a scalable heuristic, Edge-view Signed Influence Maximization (ESIM) that estimates "edge-view influence" to leverage polarity inversion in signed network.

It evaluates positive and negative edge-view influences of each edge iteratively, which represents the scale of positive influence and negative influence that may pass through. Edge-view influences on out-edges of each vertex will then be aggregated respectively and form the level of influences each vertex may cause. ESIM uses not only influence power on vertices but also those on edges to find out the suitable vertex order for seed selection. The experiment results show that our framework is more applicable in terms of time while preserving the quality of solution.

The main contributions of this work are as follows: (1) ESIM views influence calculation from a different perspective that the influence power is stored on each edge. (2) ESIM is competitive with other methods under the influence spread metric. (3) ESIM achieves faster computation compared to other works.

2 Related Works

Table 1 shows the comparison of related works. ESIM, proposed in this paper, is the scalable framework that tackles positive influence and negative influence simultaneously in signed networks.

3 Edge-view Signed Influence Maximization

3.1 Overview of Proposed Scheme

Fig. 1 shows the workflow of ESIM. First, ESIM calculates the edge-view influence iteratively according to the weight of edges and the structure of the network. Then, it aggregates the edge-view influence to form the influence power of each

Table 1: Comparison of Related Works

Algorithm	Positive	Negative	Signed	Scalable
[5,8]	✓		✓	
[4]	✓		✓	✓
[9]	✓	✓	✓	
ESIM	✓	✓	✓	✓

vertex. Next, ESIM iteratively selects the vertex with the highest influence power as the seed and updates the influence power of the rest. This process continues until ESIM obtains enough number of seed.

Our framework can be divided into two parts, "Edge-view Influence Calculation" and "Seed Selection and Influence Power Update". In the following sections, we will describe each part of the algorithm in detail.

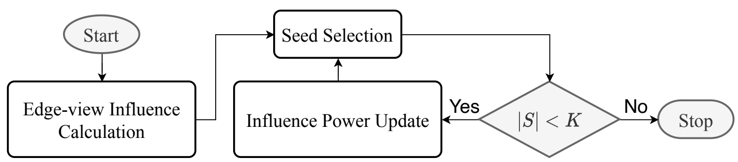


Fig. 1: Flowchart of ESIM

3.2 Edge-view Influence Calculation

$Flow^+_{(i,j)}$ and $Flow^-_{(i,j)}$ are used to denote the positive and negative edge-view influences on edge (i,j) respectively. They represent the level of positive and negative influence that vertex i may contribute through vertex j in the absence of seed set S.

To calculate $Flow^+_{(i,j)}$ and $Flow^-_{(i,j)}$ iteratively, ${Trans^+_{(i,j)}}^{(t)}$ and ${Trans^-_{(i,j)}}^{(t)}$ are defined as positive and negative transient edge-view influences on edge (i,j) at time t. Here we replace $Flow^+_{(i,j)}$ and $Flow^-_{(i,j)}$ with ${Flow^+_{(i,j)}}^{(t)}$ and ${Flow^-_{(i,j)}}^{(t)}$ respectively. The calculation is as follows:

$$ {Flow^+_{(i,j)}}^{(t)} = \sum_{\ell=1}^{t} {Trans^+_{(i,j)}}^{(\ell)} \quad and \quad {Flow^-_{(i,j)}}^{(t)} = \sum_{\ell=1}^{t} {Trans^-_{(i,j)}}^{(\ell)} \tag{1} $$

The calculation of ${Trans^+_{(i,j)}}^{(t+1)}$ and ${Trans^-_{(i,j)}}^{(t+1)}$ is given by

$$ \begin{aligned} {Trans^+_{(i,j)}}^{(t+1)} &= w_{(i,j)} \cdot (1 - r_j) \\ &\cdot \left(\left(1 - \mathbb{1}_{(i,j)}\right) \cdot \sum_k {Trans^+_{(j,k)}}^{(t)} + \mathbb{1}_{(i,j)} \cdot \sum_k {Trans^-_{(j,k)}}^{(t)} \right) \end{aligned} \tag{2} $$

$$
\begin{aligned}
&Trans^{-\,(t+1)}_{(i,j)} = w_{(i,j)} \cdot (1 - r_j) \\
&\cdot \left(\left(1 - \mathbb{1}_{(i,j)}\right) \cdot \sum_k Trans^{-\,(t)}_{(j,k)} + \mathbb{1}_{(i,j)} \cdot \sum_k Trans^{+\,(t)}_{(j,k)} \right)
\end{aligned} \tag{3}
$$

$\mathbb{1}_{(i,j)}$ is the indicator that indicates the sign of edge (i, j). That is, $\mathbb{1}_{(i,j)}$ is 1 if the sign of edge is negative, $\mathbb{1}_{(i,j)}$ is 0 otherwise. $w_{(i,j)}$ and r_j are weight and rate of decay respectively. While r_j controls the amount of transferable influence power through vertex j to upstream edges, $w_{(i,j)}$ determines the portion of transferable influence power allocated to edge (i, j) among the other in-edges pointed to vertex j. In particular, we constrain $\sum_i w_{(i,j)} = 1$. To relate these two parameters to topological property of edge (i, j) and vertex j, we use the following equations: $w_{(i,j)} = \frac{p_{(i,j)}}{d_{in}(j)}$ and $r_j = \frac{1}{1+d_{in}(j)}$.

$p_{(i,j)}$ is the probability that influence may propagate from vertex i to vertex j. $d_{in}(j)$ is weighted in-degree of vertex j that calculated by $d_{in}(j) = \sum_i p_{(i,j)}$.

The sums of positive and negative initial transient edge-view influences that originate at vertex i are according to $\sum_j Trans^{+\,(0)}_{(i,j)} = 1$ and $\sum_j Trans^{-\,(0)}_{(i,j)} = 0$. Since positive influence power is 1 and negative influence power is 0 for every vertex when they are selected as seeds.

Finally, when all transient edge-view influences are small enough at time t, then we terminate the iterative calculation of $Flow^{+\,(t)}_{(i,j)}$ and $Flow^{-\,(t)}_{(i,j)}$. The stopping criterion is $\max_i \sum_j \left(Trans^{+\,(t)}_{(i,j)} + Trans^{-\,(t)}_{(i,j)} \right) < \epsilon$ with given ϵ.

3.3 Seed Selection and Influence Power Update

As the iterative calculation ends, the edge-view influences will be aggregated to form the influence power of vertices that represent the level of influence. $Flow^{+}_{(i,j)}$ and $Flow^{-}_{(i,j)}$ represent the final results from Edge-view Influence Calculation. $Power^{+}_{i|S}$ and $Power^{-}_{i|S}$ are used to denote the positive and negative marginal gains of influence power respectively for vertex i in the presence of seed set S. When $S = \emptyset$, the following equations show the initial accumulation process: $Power^{+}_{i|S} = 1 + \sum_j Flow^{+}_{(i,j)}$ and $Power^{-}_{i|S} = \sum_j Flow^{-}_{(i,j)}$.

Then, we select the vertex with the highest Overall Marginal Gain $Overall_{i|S}$ that is defined by $Overall_{i|S} = Power^{+}_{i|S} - Power^{-}_{i|S}$.

Each time we select a vertex as a seed, the marginal gain of its neighborhood vertices has to be updated in case of redundant influence power accumulation. Since the influence power of edges and vertices are calculated “in the absence of seed set S”. While out-neighbors of newly selected seed reduce their marginal gain for being influenced, in-neighbors suffer deduction for being blocked by seed. The following equations show how to update marginal gains for vertex

$i \in \{V \setminus S | (i,j) \in E\}$ and vertex $k \in \{V \setminus S | (j,k) \in E\}$ when vertex j is selected as new seed.

$$Power^{+}_{i|S\cup\{j\}} = Power^{+}_{i|S} - Flow^{+}_{(i,j)} \tag{4}$$

$$Power^{-}_{i|S\cup\{j\}} = Power^{-}_{i|S} - Flow^{-}_{(i,j)} \tag{5}$$

$$Power^{+}_{k|S\cup\{j\}} = Power^{+}_{k|S} - Flow^{+}_{(j,k)} \tag{6}$$

$$Power^{-}_{k|S\cup\{j\}} = Power^{-}_{k|S} - Flow^{-}_{(j,k)} \tag{7}$$

3.4 Time Complexity Analysis

The overall time complexity of ESIM is derived as $O\left((|V| + |E|) \cdot \frac{\log\left(\frac{|V|}{\epsilon}\right)}{\log\left(\frac{1}{\max_v(1-r_v)}\right)}\right)$. The detail derivation is omitted due to the page limit.

4 Experiments

4.1 Datasets

We use BitcoinAlpha dataset from Stanford Large Network Dataset Collection website [3] and prune off disconnected single vertices. Table 2 shows some topological characteristics of the preprocessed network.

Table 2: Dataset Property

Nodes	3783
Edges	24186
Density	1.69E-04
Negative Edges	1536 (6.4%)
Average Deg.	12.8
Average C.C	0.1766
90% Effective Diameter	3.96599

4.2 Experiment Settings

Polarity Setting We consider both positive dominance (P-Dom) and negative dominance (N-Dom) scenarios, where either positive influence or negative influence will dominate the polarity of vertices.

Probability Settings

- **Trivalency (Tri) Model**
  The probability of each edge is assigned randomly from the set $\{0.1, 0.01, 0.001\}$
- **Uniform Cascade (Uni) Model**
  The probability of each edge is uniformly set to a given value $p = 0.01$.
- **Weighted Cascade (WC) Model**
  The probability of each edge (u, v) is equal to $\frac{1}{d_{in}(v)}$

Environment All algorithms and codes for evaluation are implemented with C++17. These codes run on an Intel(R) Xeon(R) E5-2620 v4 @2.10GHz server with 96GB memory under Ubuntu 16.04.7 LTS operating system.

4.3 Evaluation Metrics

- **Objective Influence (Objective)**
  We use $\frac{|P|-\lambda\cdot|N|}{|V|}$ to calculate the value of objective influence. $|P|$ and $|N|$ represent the number of vertices that are positively and negatively influenced respectively. λ is the hyper-parameter that adjusts the penalty from negative influence. If not mentioned, the value of λ is 1 by default. The objective influence is evaluated through Monte-Carlo simulation with 1000 iterations.
- **Execution Time (Time)**
  We record the time whenever an algorithm identifies a seed. Algorithms that exceed the time limit of one day will be terminated.

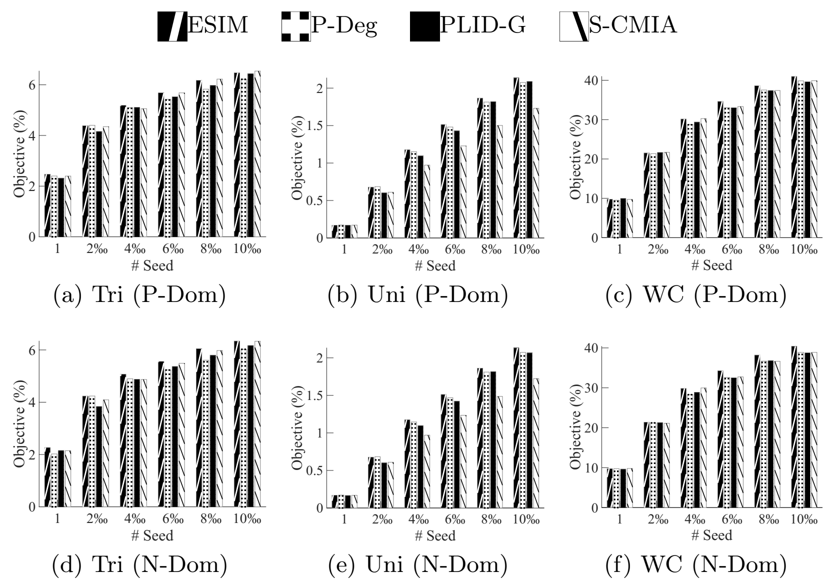
In the following figures, we use "# Seed" to represent the number of seeds. When the label of "# Seed" is X, it means we use X seed(s). When the label of "# Seed" is X‰, it means the number of seeds is X‰ of vertices. In the experiment, the number of seeds is at most 10‰ of vertices.

4.4 Compared Algorithms

- **ESIM**
  The proposed framework. In the experiment, we set $\epsilon = \frac{1}{|V|}$
- **Positive Degree (P-Deg)**
  Baseline approach that sorts the vertices in descending order according to their weighted positive out-degree.
- **PLID Greedy (PLID-G)** [4]
  PLID Greedy aims to solve PIM problem in signed social networks. We maintain its setting in [4] that maximum iteration is 5;
- **S-CMIA** [9]
  It considers the polarity dominance issue and calculates local influence spread to maximize Overall Influence Spread. In the experiment, we set its parameter α in the objective function to 0.5. The value of threshold θ for the maximum influence path is set according to the cascade model. The following shows the criteria.
  - Tri: 0.01
  - Uni: p^2, p is the uniform probability on edge.
  - WC: 0.01

4.5 Evaluation on Polarity Dominance

Fig. 2 show the result under polarity dominance setting. On BitcoinAlpha dataset, there's a mere difference between positive and negative dominance settings. The



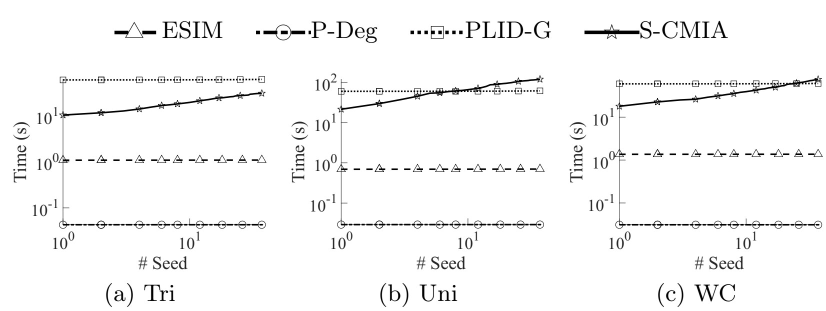
(a) Tri (P-Dom) (b) Uni (P-Dom) (c) WC (P-Dom)

(d) Tri (N-Dom) (e) Uni (N-Dom) (f) WC (N-Dom)

Fig. 2: The Experimental Results of Objective Influence

reason may be the low ratio (6.4%) of negative polarity relations on BitcoinAlpha dataset, which makes negative influence hard to generate. When the seed set size is small, the difference among algorithms is small in general. However, as the number of seeds increases, ESIM is better able to make a balance between positive influence maximization and negative influence blocking. Generally, our framework outperforms other algorithms on the polarity dominance model.

4.6 Execution Time



(a) Tri (b) Uni (c) WC

Fig. 3: The Experimental Results of Execution Time

Fig. 3 show the execution time of algorithms. P-Deg is the fastest among the four algorithms. However, it does not achieve the best performance in terms of objective function. Though PLID-G has steady time consumption, it is one of the most time-consuming algorithms. On the other hand, S-CMIA spends less time than PLID-G when the required number of seeds is small but shows exponential growth. ESIM shows its efficiency that it not only finishes the task within two minutes on different datasets but also preserves near-constant growth. Therefore, our framework is more applicable in terms of time consumption.

5 Conclusion

In this paper, we study the influence maximization problem with binary competitive influences spreading over signed social networks. We design ESIM to leverage the polarity inversion mechanism and make a balance between positive influence maximization and negative influence blocking. The experiment shows that our framework preserves effectiveness and efficiency on the real-world network dataset under several polarity settings.

References

1. Chen, W., Collins, A., Cummings, R., Ke, T., Liu, Z., Rincón, D., Sun, X., Wang, Y., Wei, W., Yuan, Y.: Influence maximization in social networks when negative opinions may emerge and propagate. In: Proceedings of the Eleventh SIAM International Conference on Data Mining. pp. 379–390 (2011)
2. He, X., Song, G., Chen, W., Jiang, Q.: Influence blocking maximization in social networks under the competitive linear threshold model. In: Proceedings of the Twelfth SIAM International Conference on Data Mining. pp. 463–474 (2012)
3. Leskovec, J., Krevl, A.: SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data (Jun 2014)
4. Li, D., Liu, J.: Modeling influence diffusion over signed social networks. IEEE Trans. Knowl. Data Eng. **33**(2), 613–625 (2021)
5. Li, D., Xu, Z.M., Chakraborty, N., Gupta, A., Sycara, K., Li, S.: Polarity related influence maximization in signed social networks. PloS one **9**(7), e102199 (2014)
6. Shi, Q., Wang, C., Ye, D., Chen, J., Feng, Y., Chen, C.: Adaptive influence blocking: Minimizing the negative spread by observation-based policies. In: 35th IEEE International Conference on Data Engineering. pp. 1502–1513 (2019)
7. Song, C., Hsu, W., Lee, M.: Temporal influence blocking: Minimizing the effect of misinformation in social networks. In: 33rd IEEE International Conference on Data Engineering. pp. 847–858 (2017)
8. Srivastava, A., Chelmis, C., Prasanna, V.K.: Social influence computation and maximization in signed networks with competing cascades. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. pp. 41–48 (2015)
9. Sung, C., Ma, H., Huang, J.: Positive influence maximization and negative influence minimization in signed networks under competitive independent cascade model. In: 7th IEEE International Conference on Data Science and Advanced Analytics. pp. 236–244 (2020)