Solving the Contamination Minimization Problem on Networks for the Linear Threshold Model

Masahiro Kimura[1], Kazumi Saito[2], and Hiroshi Motoda[3]

[1] Department of Electronics and Informatics, Ryukoku University
Otsu 520-2194, Japan
kimura@rins.ryukoku.ac.jp
[2] School of Administration and Informatics, University of Shizuoka
Shizuoka 422-8526, Japan
k-saito@u-shizuoka-ken.ac.jp
[3] Institute of Scientific and Industrial Research, Osaka University
Osaka 567-0047, Japan
motoda@ar.sanken.osaka-u.ac.jp

Abstract. We address the problem of minimizing the spread of undesirable things, such as computer viruses and malicious rumors, by blocking a limited number of links in a network. This optimization problem called the contamination minimization problem is, not only yet another approach to the problem of preventing the spread of contamination by removing nodes in a network, but also a problem that is converse to the influence maximization problem of finding the most influential nodes in a social network for information diffusion. We adapted the method which we developed for the independent cascade model, known for a model for the spread of epidemic disease, to the contamination minimization problem under the linear threshold model, a model known for the propagation of innovation which is considerably different in nature. Using large real networks, we demonstrate experimentally that the proposed method significantly outperforms conventional link-removal methods.

1 Introduction

Networks can mediate diffusion of various things such as innovation and topics. However, undesirable things can also spread through networks. For example, computer viruses can spread through computer networks and email networks, and malicious rumors can spread through social networks among individuals. Thus, developing effective strategies for preventing the spread of undesirable things through a network is an important research issue. Previous work studied strategies for reducing the spread size by removing nodes from a network. It has been shown in particular that the strategy of removing nodes in decreasing order of out-degree can often be effective [1, 2, 3]. Here notice that removal of nodes by necessity involves removal of links. Namely, the task of removing links is more fundamental than that of removing nodes. Therefore, preventing the spread of contamination by blocking links from the underlying network is an important problem.

In contrast, finding a limted number of influential nodes that are effective for the spread of information through a social network is also an important research issue in

terms of sociology and "viral marketing" [4, 5, 6]. Widely-used fundamental probabilistic models of information diffusion through networks are the *independent cascade (IC) model* and the *linear threshold (LT) model* [7, 6]. Researchers have recently studied a combinatorial optimization problem called the *influence maximization problem* on a network under these models [7, 8]. Here, the influence maximization problem is the problem of extracting a set of k nodes to target for initial activation such that it yields the largest expected spread of information, where k is a given positive integer. Note also that the IC and LT models are fundamental models of contamination diffusion process on networks [6].

The problem we address in this paper is a problem that is converse to the influence maximization problem. The problem is to minimize the spread of contamination by blocking a limited number of links in a network. More specifically, when some undesirable thing starts with any node and diffuses through the network, we consider finding a set of k links such that the resulting network by blocking those links minimizes the expected contamination area of the undesirable thing, where k is a given positive integer. This combinatorial optimization problem is referred to as the *contamination minimization problem* [9]. For the contamination minimization problem under the IC model, Kimura, Saito and Motoda [9] presented a method for efficiently finding a good approximate solution on the basis of a naturally greedy strategy.

In this paper, we propose a method for efficiently finding a good approximate solution to the contamination minimization problem under the LT model by adapting the greedy method developed for the problem under the IC model. Note here that the IC and LT models considerably differ in quality. First, the LT model is originally a model for the propagation of innovation through the network, while the IC model can be identified with the *SIR model* [10] for the spread of epidemic disease in the network. Moreover, the LT model is viewed as a probabilistic model defined on some continous space, while the IC model is viewed as that on some finite set (i,e., a discrete space) [7, 8]. Therefore, the effectiveness of the greedy method for the problem under the LT model is not self-evident. To compare methods of solving the problem for various networks in performance, we newly introduce the *contamination reduction rate* as a performance measure. Using large real social networks, we experimentally demonstrate that the proposed method significantly outperforms link-removal heuristics that rely on the well-studied notions of betweenness and out-degree in the field of complex network theory.

2 Problem Formulation

In this paper, we address the problem of minimizing the spread of some undesirable thing in a network represented by a directed graph $G = (V, E)$. Here, V and E ($\subset V \times V$) are the sets of all the nodes and links in the network, respectively. We assume the LT model to be a mathematical model for the diffusion process of this undesirable thing in the network, and investigate the contamination minimization problem on G. We call nodes *active* if they have been contaminated by the undesirable thing.

2.1 Linear Threshold Model

We define the *linear threshold (LT) model* on graph G according to [7].

In this model, for any node $v \in V$, we specify, in advance, a *weight* $\omega_{u,v}$ (> 0) from its parent node u such that $\sum_{u \in \Gamma(v)} \omega_{u,v} \leq 1$, where $\Gamma(v)$ is the set of all the parent nodes of v, $\Gamma(v) = \{u \in V; (u, v) \in E\}$. The diffusion process from a given initial set of active nodes proceeds according to the following randomized rule. First, for any node $v \in V$, a *threshold* θ_v is chosen uniformly at random from the interval $[0, 1]$. At time-step t, an inactive node v is influenced by each of its active parent nodes, u according to weight $\omega_{u,v}$. If the total weight from active parent nodes of v is at least threshold θ_v, that is, $\sum_{u \in \Gamma_t(v)} \omega_{u,v} \geq \theta_v$, then v will become active at time-step $t + 1$. Here, $\Gamma_t(v)$ stands for the set of all the parent nodes of v that are active at time-step t. The process terminates if no more activations are possible.

Note that the threshold θ_v models the tendency of node v to adopt the information when its parent nodes do. Note also that the LT model is a probabilistic model associated with the uniform distribution on $[0, 1]^{|V|}$. Thus, the LT model is viewed as a probabilistic model on the continous space $[0, 1]^{|V|}$. Here, $|A|$ stands for the number of elements of a set A.

For an initial active node v, let $\sigma(v; G)$ denote the expected number of active nodes at the end of the random process of the LT model on G. We call $\sigma(v; G)$ the *influence degree* of node v in graph G.

2.2 Contamination Minimization Problem

Now, we give a mathematical definition of the contamination minimization problem on graph $G = (V, E)$.

First, we define the *contamination degree* $c(G)$ of graph G as the average of influence degrees of all the nodes in G, that is,

$$c(G) = \frac{1}{|V|} \sum_{v \in V} \sigma(v; G). \tag{1}$$

For any link $e \in E$, let $G(e)$ denote the graph $(V, E \setminus \{e\})$. We refer to $G(e)$ as the graph constructed by *blocking* e in G. Similarly, for any $D \subset E$, let $G(D)$ denote the graph $(V, E \setminus D)$. We refer to $G(D)$ as the graph constructed by *blocking* D in G. We define the *contamination minimization problem* on graph G as follows: Given a positive integer k with $k < |E|$, find a subset D^* of E with $|D^*| = k$ such that $c(G(D^*)) \leq c(G(D))$ for any $D \subset E$ with $|D| = k$.

For a large network, any straightforward method for exactly solving the contamination minimization problem suffers from combinatorial explosion. Therefore, we consider approximately solving the problem.

3 Proposed Method

We propose a method for efficiently finding a good approximate solution to the contamination minimization problem on graph $G = (V, E)$. We consider adapting the method which we developed for the IC model to the contamination minimization problem under the LT model which is considerably different in nature. Let k be the number of links to be blocked in this problem.

3.1 Geedy Algorithm

We approximately solve the contamination minimization problem on $G = (V, E)$ by the following greedy algorithm:

1. Set $D_0 \leftarrow \emptyset$.
2. Set $E_0 \leftarrow E$.
3. Set $G_0 \leftarrow G$.
4. **for** $i = 0$ to $k - 1$ **do**
5. Choose a link $e_* \in E_i$ minimizing $c(G_i(e))$, $(e \in E_i)$.
6. Set $D_{i+1} \leftarrow D_i \cup \{e_*\}$.
7. Set $E_{i+1} \leftarrow E_i \setminus \{e_*\}$.
8. Set $G_{i+1} \leftarrow (V, E_{i+1})$.
9. **end for**

Here, D_k is the set of links blocked, and represents the approximate solution obtained by this algorithm. G_k is the graph constructed by blocking D_k in graph G, that is, $G_k = G(D_k)$.

To implement this greedy algorithm, we need a method for calculating $\{c(G_i(e)); e \in E_i\}$ in Step 5 of the algorithm. However, the LT model is a stochastic process model, and it is an open question to exactly calcluate influence degrees by an efficient method [7]. Therefore, we develop a method for estimating $\{c(G_i(e)); e \in E_i\}$.

Kimura, Saito, and Nakano [8] presented the bond percolation method that efficiently estimates the influence degrees $\{\sigma(v; \tilde{G}); v \in \tilde{V}\}$ for any directed graph $\tilde{G} = (\tilde{V}, \tilde{E})$. Thus, we can estimate $c(G_i(e))$ for each $e \in E_i$ by straightforwardly applying the bond percolation method. However, $|E_i|$ becomes very large for a large network unless i is very large. Therefore, we propose a method that can estimate $\{c(G_i(e)); e \in E_i\}$ in a more efficient manner on the basis of the bond percolation method.

3.2 Estimation Based on Bond Percolation Method

It is known that the LT model is equivalent to the following bond percolation process [7]: For any $v \in V$, we pick at most one of the incoming links to v by selecting link (u, v) with probability $\omega_{u,v}$ and selecting no link with probability $1 - \sum_{u \in \Gamma(v)} \omega_{u,v}$. Then, we declare the picked links to be "occupied" and the other links to be "unoccupied". Note here that the equivalent bond percolation process for the LT model is considerably different from that of IC model.

In the bond percolation method [8], we efficiently estimate the influence degrees $\{\sigma(v; G_i); v \in V\}$ in the following way. Let M be a sufficiently large positive integer. We perform the bond percolation process M times, and sample a set of M graphs, $\{G_i{}^m = (V, E_i{}^m); m = 1, \cdots, M\}$, constructed by the occupied links. Then, using the strongly connected decomposition of each G_i^m, we efficiently estimate the influence degrees $\{\sigma(v; G_i); v \in V\}$ as

$$\sigma(v; G_i) = \frac{1}{M} \sum_{m=1}^{M} |\mathcal{F}(v; G_i{}^m)|, \quad (v \in V), \tag{2}$$

(see [8] in detail). Here, $\mathcal{F}(v; {G_i}^m)$ denotes the set of all the nodes that are *reachable* from node v in the graph ${G_i}^m$. We say that node u is reachable from node v if there is a path from u to v along the links in the graph.

We are now in a position to give a method for efficiently estimating $\{c(G_i(e)); e \in E_i\}$ in Step 5 of the greedy algorithm. For the LT model, the weights $\{\omega_{u,v}\}$ must be specified in advance. We uniformly set the weights as follows: For any node $v \in V$, the weight $\omega_{u,v}$ from a parent node $u \in \Gamma(v)$ is given by

$$\omega_{u,v} = \frac{1}{|\Gamma(v)| + 1}.$$

Here note that $\sum_{u \in \Gamma(v)} \omega_{u,v} < 1$ for any $v \in V$, that is, there exists a chance such that node v cannot become active even if all the parent nodes of v are active. Then, on the basis of Equations (1) and (2), and the independence of the bond percolation process, we estimate $\{c(G_i(e)); e \in E_i\}$ by

$$c\,(G_i(e)) \;=\; \frac{1}{|\mathcal{M}_i(e)|} \sum_{m \in \mathcal{M}_i(e)} \frac{1}{|V|} \sum_{v \in V} \mathcal{F}(v; {G_i}^m), \qquad (e \in E_i)$$

without applying the bond percolation method for every $e \in E_i$, where $\mathcal{M}_i(e) = \{m \in \{1, \cdots, M\}; e \notin {E_i}^m\}$. Namely, the proposed method can achieve a great deal of reduction in computational cost compared with the coventional bond percolation method.

4 Experimental Evaluation

4.1 Experimental Settings

In our experiments, we employed two sets of large real networks used in [9], the blog and Wikipedia networks, which exhibit many of the key features of social networks. These are bidirectional networks. The blog network had $12,047$ nodes and $79,920$ directed links, and the Wikipedia network had $9,481$ nodes and $245,044$ directed links.

For the proposed method, we need to specify the number M of performing the bond percolation process. In the experiments, we used $M = 10,000$ according to [8].

4.2 Comparison Methods

We compared the proposed method with two heuristics based on the well-studied notions of betweenness and out-degree in the field of complex network theory.

The *betweenness score* $b_{\tilde{G}}(e)$ of a link e in a directed graph $\tilde{G} = (\tilde{V}, \tilde{E})$ is defined as follows: $b_{\tilde{G}}(e) = \sum_{u,v \in \tilde{V}} n_{\tilde{G}}(e; u, v)/N_{\tilde{G}}(u, v)$, where $N_{\tilde{G}}(u, v)$ denotes the number of the shortest paths from node u to node v in $\tilde{G}$, and $n_{\tilde{G}}(e; u, v)$ denotes the number of those shortest paths that pass e. Here, we set $n_{\tilde{G}}(e; u, v)/N_{\tilde{G}}(u, v) = 0$ if $N_{\tilde{G}}(u, v) = 0$. Newman and Girvan [11] successfully extracted community structure in a network using the following link-removal algorithm based on betweeness:

1. Calculate betweenness scores for all links in the network.
2. Find the link with the highest score and remove it from the network.

3. Recalculate betweenness scores for all remaining links.
4. Repeat from Step 2.

In particular, the notion of betweenness can be interpreted in terms of signals traveling through a network. If signals travel from source nodes to destination nodes along the shortest paths in a network, and all nodes send signals at the same constant rate to all others, then the betweenness score of a link is a measure of the rate at which signals pass along the link. Thus, we naively expect that blocking the links with the highest betweenness score can be effective for preventing the spread of contamination in the network. Therefore, we apply the method of Newman and Girvan [11] to the contamination minimization problem. We refer to this method as the *betweenness method*.

On the other hand, previous work has shown that simply removing nodes in order of decreasing *out-degrees* works well for preventing the spread of contamination in most real networks [1, 2, 3]. Here, the out-degree of a node v means the number of outgoing links from the node v. Therefore, as a comparsion method, we consider the straighforward application of this node removal method. Namely, we employ the method of choosing nodes in decreasing order of out-degree and blocking simultaneously all the links attached to the chosen nodes. We refer to this method as the *out-degree method*. Note that the out-degree method can not be applied for all values of k to the contamination minimization problem of blocking k links.

4.3 Experimental Results

We evaluated the performance of the proposed method and compared it with that of the betweenness and out-degree methods. Clearly, the performance of a method for solving the contamination minimization problem can be evaluated in terms of the *contamination reduction rate CRR* that is defined as follows:

$$CRR \;=\; 100\,\frac{c(G) - c(G')}{c(G)}\,,$$

where G' stands for a solution graph constructed by blocking a specified number of links from the original graph G. We estimated the value of c by the bond percolation method with $M = 10,000$ (see Equations (1) and (2)), and computed the value of *CRR*.

Figures 1 and 2 show the contamination reduction rate *CRR* of the resulting network as a function of the *fraction of links blocked*, FLB, for the blog and Wikipedia networks, respectively. Here, the circles, triangles and diamonds indicate the results for the proposed, betweenness and out-degree methods, respectively. In the right figures of Figures 1 and 2, the dashed line indicates the contamination reduction rate of the network obtained by the proposed method when the number of links blocked, k, is 500. Here note that $k = 500$ means $FLB = 0.63\%$ and $FLB = 0.20\%$ in the blog and Wikipedia networks, respectively. We see that the proposed method outperformed the betweenness and out-degree methods for both the blog and the Wikipedaia networks.

These results imply that the proposed method works effectively as expected, and significantly outperforms the conventional link-removal heuristics, that is, the betweenness and out-degree methods. This shows that a significantly better link-blocking strategy for reducing the spread size of contamination can be obtained by explicitly incorporating

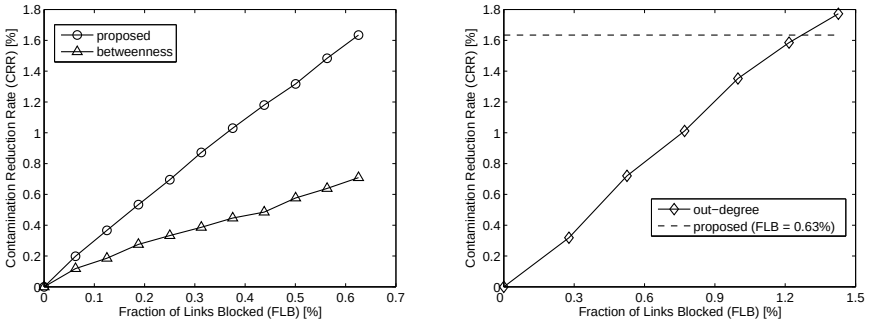

Fig. 1: Performance comparison of the proposed method with the betweenness and out-degree methods in the blog network.

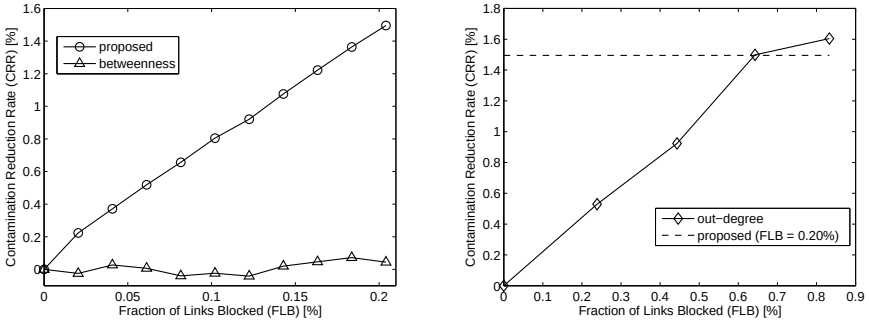

Fig. 2: Performance comparison of the proposed method with the betweenness and out-degree methods in the Wikipedia network.

the diffusion dynamics of contamination in a network, rather than relying solely on structual properties of the graph.

In the task of removing nodes from a network, the out-degree heuristic has been effective since many links can be blocked at the same time by removing nodes with high out-degrees. However, we find that in the task of blocking a limited number of links, the strategy of blocking all the links attached to nodes with high out-degrees is not necessarily effective.

5 Conclusion

In an attempt to minimize the spread of undesirable things, such as computer viruses and malicious rumors, by blocking a limited number of links in a network, we have invesitigated the contamination minimization problem for the LT model that is a fundamental diffusion model on a network. This minimization problem is, not only yet

another approach to the problem of preventing the spread of contamination by removing nodes in a network, but also a problem that is converse to the influece maximization problem of finding the most influential nodes in a social network for information diffusion. We have adapted the method which we developed for the IC model, known for a model for the spread of epidemic disease, to the contamination minimization problem under the LT model, a model known for the propagation of innovation which is considerably different in nature. Using large-scale blog and Wikipedia networks, we have experimentally demonstrated that the proposed method effectively works, and also significantly outperforms the conventional link-removal heuristics based on the betweenness and out-degree.

Acknowledgments

This work was partly supported by Asian Office of Aerospace Research and Development, Air Force Office of Scientific Research, U.S. Air Force Research Laboratory under Grant No. AOARD-08-4027, and Grant-in-Aid for Scientific Research (C) (No. 20500147) from Japan Society for the Promotion of Science.

References

[1] Albert, R., Jeong, H., Barabási, A.L.: Error and attack tolerance of complex networks. Nature **406** (2000) 378–382
[2] Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata, R., Tomkins, A., Wiener, J.: Graph structure in the web. In: Proceedings of the 9th International World Wide Web Conference. (2000) 309–320
[3] Newman, M.E.J., Forrest, S., Balthrop, J.: Email networks and the spread of computer viruses. Physical Review E **66** (2002) 035101
[4] Domingos, P., Richardson, M.: Mining the network value of customers. In: Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. (2001) 57–66
[5] Richardson, M., Domingos, P.: Mining knowledge-sharing sites for viral marketing. In: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. (2002) 61–70
[6] Gruhl, D., Guha, R., Liben-Nowell, D., Tomkins, A.: Information diffusion through blogspace. In: Proceedings of the 13th International World Wide Web Conference. (2004) 107–117
[7] Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network. In: Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. (2003) 137–146
[8] Kimura, M., Saito, K., Nakano, R.: Extracting influential nodes for information diffusion on a social network. In: Proceedings of the 22nd AAAI Conference on Artificial Intelligence. (2007) 1371–1376
[9] Kimura, M., Saito, K., Motoda, H.: Minimizing the spread of contamination by blocking links in a network. In: Proceedings of the 23rd AAAI Conference on Artificial Intelligence. (2008) 1175–1180
[10] Newman, M.E.J.: The structure and function of complex networks. SIAM Review **45** (2003) 167–256
[11] Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Physical Review E **69** (2004) 026113