JMLR: Workshop and Conference Proceedings 29:149–164, 2013 ACML 2013

Predictive Simulation Framework of Stochastic Diffusion Model for Identifying Top-K Influential Nodes

Kouzou Ohara OHARA@IT.AOYAMA.AC.JP
Department of Integrated Information Technology, Aoyama Gakuin University

Kazumi Saito K-SAITO@U-SHIZUOKA-KEN.AC.JP
School of Administration and Informatics, University of Shizuoka

Masahiro Kimura KIMURA@RINS.RYUKOKU.AC.JP
Department of Electronics and Informatics, Ryukoku University

Hiroshi Motoda MOTODA@AR.SANKEN.OSAKA-U.AC.JP
Institute of Scientific and Industrial Research, Osaka University
School of Computing, University of Tasmania

Editor: Cheng Soon Ong and Tu Bao Ho

Abstract

We address a problem of efficiently estimating the influence of a node in information diffusion over a social network. Since the information diffusion is a stochastic process, the influence degree of a node is quantified by the expectation, which is usually obtained by very time consuming many runs of simulation. Our contribution is that we proposed a framework for predictive simulation based on the leave-N-out cross validation technique that well approximates the error from the unknown ground truth for two target problems: one to estimate the influence degree of each node, and the other to identify top-K influential nodes. The method we proposed for the first problem estimates the approximation error of the influence degree of each node, and the method for the second problem estimates the precision of the derived top-K nodes, both without knowing the true influence degree. We experimentally evaluate the proposed methods using the three real world networks, and show that they can serve as a good measure to solve the target problems with far fewer runs of simulation ensuring the accuracy if N is appropriately chosen, and that estimating the top-K nodes is easier than estimating the influence degree, which means one can identify the influential nodes without knowing exactly their influence degree.

Keywords: Predictive simulation, Influence degree, Information diffusion

1. Introduction

The emergence of Social Media such as Facebook, Digg, Twitter, Weblog, Wiki, etc. has provided us with the opportunity to create large social networks. Once an article is posted on social media, it can rapidly and widely spread through these networks and can be shared by a large number of people. Thus, it has a large influence on our thought and decision making. This phenomenon has attracted the interest of many researchers from diverse fields, *e.g*, sociology, psychology, economy, computer science (Kleinberg, 2008). In view of the importance of this phenomenon it is becoming pressingly important that we are able to efficiently analyze this huge amount of information and estimate its influence.

Among a large number of studies on modeling how information propagates through a social network (Yang and Counts, 2010; Yang and Leskovec, 2010; Bakshy et al., 2011; Cui et al., 2011; Guille and Hacid, 2012), of particular importance is the influence maximization problem (Kempe et al., 2003; Leskovec et al., 2007; Chen et al., 2009, 2010b; Kimura et al., 2007, 2010) in which the task is to identify a limited number of nodes which together maximize the information spread, and its variants. Variants include the contamination minimization problem in which the task is to identify a limited number of links which together, if blocked, minimizes the information spread (Kimura et al., 2008, 2009b), and the target selection problem in which the task is to identify a limited number of target nodes to send information from outside of a network such that the influence spread is maximized (Saito et al., 2013)[1].

All of these problems need to estimate the influence of a node (a set of nodes) and rank the nodes in accordance with the influence degree. Since the process of information diffusion is modeled as a stochastic process, quantification of influence is meaningful only in the sense of expectation. The influence degree of a node v is defined to be the expected number of the active nodes at the end of information diffusion, *i.e.*, the expected number of nodes where the information started at the node v eventually reaches as the results of information cascade (See 2.1). Kempe et al. (2003) first solved the influence maximization problem by approximating the influence degree by the empirical mean of many runs of simulation and selecting the best set of nodes using a greedy search strategy. Since then, various techniques have been proposed to improve the computational efficiency both in estimating the influence degree and finding the best set of nodes. These include bond percolation (Kimura et al., 2007, 2010), pruning (Kimura et al., 2009a), lazy evaluation (Leskovec et al., 2007; Goyal et al., 2011), burnout (Saito et al., 2009a), shortest path heuristics (Kimura and Saito, 2006; Chen et al., 2009, 2010a,b), belief propagation (Nguyen and Zheng, 2012) and linear system approximation (Yang et al., 2012). Pruning, lazy evaluation and burnout are techniques to reduce the cost of search. The rests are related to techniques to reduce the cost of estimating the influence degree or to reduce the cost of both. A bond percolation is the process in which each link of a network is randomly designated either "occupied " or "unoccupied" according to the diffusion probability associated with each link, and the probabilistic simulation is replaced with the counting of nodes reachable from the starting node (See 2.2). A single bond percolation allows to estimate the number of reachable nodes from all the starting nodes at once, which reduces the computation cost by 2 to 3 orders of magnitude. However, it requires many runs of bond percolation. As the number of runs goes to infinity, the empirical mean converges to the true expectation. The shortest path heuristics, belief propagation and linear system application introduce approximation to the diffusion paths, *e.g.*, assume DAG, and estimate the influence directly, but there is no formal measure how close the estimated influence degree is to the ground truth.

In this paper we take the approach to estimate the expectation by the empirical mean of many runs (weather each run is bond percolation and counting or direct probabilistic simulation or some other does not matter), and propose a framework to evaluate how close the approximation is to the ground truth without knowing the correct answer. We have two targets to estimate: one is the influence degree of individual node and the other is ranking of the nodes. Since we are only interested in the influential nodes, we consider only the top-K nodes with respect to the influence degree. The framework we propose is based on the cross-validation, in particular, we show the one based on leave-N-out in this paper. It returns an approximated difference between the estimated answer and

1. This is similar to the influence maximization problem, but is different in that the target nodes are not guaranteed to spread the passed information, *i.e.*, the cascade may stop there.

the unknown ground truth, *i.e.*, the approximation error of the estimated influence degree for the former target and the precision of the estimated top-K nodes for the latter target. We have tested our method using three real world network structures. The ground truth we used to evaluate our method is obtained by the empirical mean of one million independent runs. Extensive experiments were performed varying the number of runs and repeating them multiple times to evaluate the estimated influence degree and its standard deviation, and top-K nodes precision. The global performance is in line with what the central limit theory indicates, but the details vary depending on the specific network structure and the individual node. We found that estimating top-K nodes is easier than estimating the influence degree itself for all networks. This implies we can identify the influential nodes without knowing the very accurate influence degree. The method we proposed is not specific to social network application. It is very generic and is applicable to any other estimation problems for predictive simulation in which we need a criterion when to stop ensuring the predictive performance.

The paper is organized as follows. Section 2 describes the information diffusion models we used, formally defines influential degree and explains bond percolation. Section 3 is the main part where the problem is mathematically defined as a machine learning problem, and a solution is proposed. Section 4 reports experimental results for influence degree estimation and top-K influential node identification using three real world networks. Section 5 summarizes what has been achieved in this work and addresses the future work.

2. Information Diffusion Models

2.1. Stochastic Cascade Model and Influence Degree

We investigate the spread of information through a social network represented by a directed graph $G = (V, E)$, where V and E ($\subset V \times V$) are the sets of all the nodes and links in the network, respectively. A node is called *active* if it has been influenced with the information, and *inactive* otherwise. We assume that nodes can switch their states from inactive to active, but cannot switch the other way around. When there is a link (u, v) from node u to node v, u is called a *parent node* of v and v is called a *child node* of u. For any node $v \in V$, let $\Gamma(v)$ denote the set of all parent nodes of v in G, that is, $\Gamma(v) = \{u \in V; (u, v) \in E\}$. Here, we focus on stochastic *cascade models* of information diffusion in G such that each node v can be influenced directly only by its parent nodes $\Gamma(v)$.

For simplicity, we only describe basic discrete-time models although it is possible to extend them to more realistic continuous-time models (e.g., asynchronous models with continuous-time delays (Saito et al., 2009b, 2010)). Given an initial set V_0 ($\subset V$) of active nodes at time 0, the diffusion process unfolds in the following way: If a node u first becomes active at time t, it has one chance of activating each inactive child node v. If u succeeds, then v will become active at time $t + 1$. If multiple parent nodes of v become active at time t, their activation attempts are sequenced in an arbitrary order, but performed at time t. Whether or not u succeeds, it cannot make any further attempts to activate v in subsequent rounds. The process terminates if no more activations are possible. Here, the probability that node u succeeds to activate node v is given by some appropriate function $P_v(u, \tilde{\Gamma}^*_t(v))$, where $\tilde{\Gamma}^*_t(v)$ is the set of pairs (w, t_w) such that $w \in \Gamma(v)$ and $t_w < t$; t_w denotes the time at which w first became active. Note that the diffusion (success) probability $P_v(u, \tilde{\Gamma}^*_t(v))$ must be constructed so that the order-independence of multiple activation attempts at any time are satisfied. Let $A(V_0)$ denote the number of active nodes at the end of the diffusion process. Note that $A(V_0)$ is a random variable. Let $\langle A(V_0) \rangle$ denote the expected value of $A(V_0)$. We refer $\langle A(V_0) \rangle$ to

as the *influence degree* of V_0. When the initial active set V_0 consists of a single node v_0, we simply denote $A(\{v_0\})$ and $\langle A(\{v_0\})\rangle$ by $A(v_0)$ and $\langle A(v_0)\rangle$, respectively. In this paper, we explore an efficient estimation of $\langle A(v_0)\rangle$ for any node $v_0 \in V$.

2.2. Independent Cascade Model and Bond Percolation

One of the simplest models in this framework is the *independent cascade (IC) model* (Kempe et al., 2003), where the diffusion probability $P_v(u, \tilde{\Gamma}^*_t(v))$ from node u to node v is a constant $p_{u,v}$ ($0 \leq p_{u,v} \leq 1$), which is independent of the history of the diffusion process. In our experiments, we examine the effectiveness of the proposed estimation method for the IC model although the method can be applied to other general stochastic cascade models. Note that the IC model on G can be identified with the so-called susceptible/infective/recovered (SIR) model (Newman, 2003; Watts and Dodds, 2007) for the spread of a disease on G, where the nodes that first become active at time t in the IC model correspond to the infective nodes at time t in the SIR model. In the SIR model, each node can have three states, "susceptible", "infective", and "recovered", where a susceptible node becomes infective with a certain probability when its parent node is infective, and subsequently recovers. It is known that the SIR model on a network can be exactly mapped onto a *bond percolation model* on the same network (Newman, 2003; Kempe et al., 2003). Thus, the IC model on G is equivalent to a bond percolation model on G, that is, these two models have the same probability distribution for the final set of active/recovered nodes. In our experiments, we exploit this equivalence between the IC and bond percolation models, and efficiently estimate the influence degree of any node $v \in V$ based on the *bond percolation method* (Kimura et al., 2010). Below, we revisit the bond percolation method.

The *bond percolation process with occupation probabilities* $\{p_{u,v} \,|\, (u,v) \in E\}$ on graph G is the random process in which each link $(u,v) \in E$ is independently declared "occupied" with probability $p_{u,v}$. Note that in terms of information diffusion on a network, the occupied links represent the links through which the information propagates, and the unoccupied links represent the links through which the information does not propagate. Here we consider a set S defined by $S = \{1, \cdots, |S|\}$. For a positive integer $|S|$, we perform the bond percolation process $|S|$ times, and sample a set of $|S|$ graphs constructed by the occupied links,

$$\{G^s = (V, E^s) \,|\, s \in S\}.$$

For any $v \in V$, we define $\bar{A}_S(v)$ by

$$\bar{A}_S(v) = \frac{1}{|S|} \sum_{s \in S} \left| F(v; G^s) \right|. \tag{1}$$

Here, $F(v; G^s)$ stands for the set of all the nodes that are *reachable* from node v on graph G^s. We say that node u is reachable from node v on graph G^s if there is a path from v to u along the links on graph G^s. Since for the IC model with diffusion probabilities $\{p_{u,v} \,|\, (u,v) \in E\}$ on graph G the bond percolation proceeds with occupation probabilities $\{p_{u,v} \,|\, (u,v) \in E\}$ on graph G, the influence degree $\langle A(v)\rangle$ of node $v \in V$ for the IC model can well be approximated by $\bar{A}_S(v)$, that is,

$$\langle A(v)\rangle \sim \bar{A}_S(v), \quad (v \in V), \tag{2}$$

if $|S|$ is sufficiently large. We decompose each graph G^s into the strongly connected components (SCCs) as follows:

$$V = \bigcup_{j=1}^{J^s} SCC(u_j^s; G^s),$$

where J^s is the number of the strongly connected components of graph G^s, each u_j^s is an element of V, and $SCC(u_j^s; G^s)$ denotes the SCC of graph G^s that contains node u_j^s. Note that

$$\left|F(v; G^s)\right| = \left|F(u_j^s; G^s)\right|, \quad \text{if } v \in SCC(u_j^s; G^s). \tag{3}$$

Thus, by calculating $\{|F(u_j^s; G^s)| \, j = 1, \cdots, J^s\}$ in advance and using Equation (3), we efficiently calculate $|F(v; G^s)|$ for all $v \in V$ at once. Once we have $\{|F(v; G^s)| \, | \, v \in V, s \in S\}$, we can calculate $\bar{A}_S(v)$ for all $v \in V$ from Equation (1).

Namely, the bond percolation method estimates all the influence degrees $\{\langle A(v)\rangle \, | \, v \in V\}$ on graph G as follows: It first specifies the value of integer $|S|$, calculates $\bar{A}_S(v)$ for all $v \in V$ by performing the above procedure, and estimates $\langle A(v)\rangle$ for all $v \in V$ by using Equation (2).

3. Predictive Simulation Framework

3.1. Influence Degree Estimation

We first consider a problem of estimating influence degree of node $v \in V$. More formally, for a given set of samples $\{A_s(v) = |F(v; G^s)| \, | \, s \in S\}$, where each sample is independently generated according to an identical distribution induced from a bond percolation process, we attempt quantitatively evaluating the following expected approximation error of the estimated influence degree $\bar{A}_S(v)$ of Equation (1) with respect to the influence degree $\langle A(v)\rangle$,

$$D_S(v) = \langle |\langle A(v)\rangle - \bar{A}_S(v)| \rangle. \tag{4}$$

Namely, we formulate our problem as estimating the approximation error between $\langle A(v)\rangle$ and $\bar{A}_S(v)$ only from a limited number of samples $\{A_s(v) \, | \, s \in S\}$ without assuming $\langle A(v)\rangle$ in a typical machine learning problem setting.

To this end, we propose methods based on a leave-N-out cross validation technique in a machine learning approach. For a positive integer $N < |S|$, let $\mathcal{B} \subset 2^S$ be a family of subsets of S whose number of elements is N, that is, $|B| = N$ for $B \in \mathcal{B}$. Then, we can consider the following estimation formula for the approximation error of the influence degree:

$$\begin{aligned}
\bar{D}_S(v; N) &= \sqrt{\langle (\bar{A}_S(v) - \bar{A}_{S\setminus B}(v))^2 \rangle_{B\in\mathcal{B}}} \\
&= \sqrt{\binom{|S|}{N}^{-1} \sum_{B\in\mathcal{B}} \left(\bar{A}_S(v) - \frac{1}{|S| - N} \sum_{s\in S\setminus B} A_s(v) \right)^2} \\
&= C_S(N) \bar{\sigma}_S(v)
\end{aligned} \tag{5}$$

Here $C_S(N)$ is the function calculated by

$$C_S(N) = \sqrt{\frac{N}{(|S| - 1)(|S| - N)}}, \tag{6}$$

and $\bar{\sigma}_S(v)$ denotes the empirical standard deviation of $|S|$ simulation results $\{A_s(v) \mid s \in S\}$,

$$\bar{\sigma}_S(v) = \sqrt{\frac{1}{|S|}\sum_{s\in S}\left(A_s(v) - \bar{A}_S(v)\right)^2}. \tag{7}$$

As for settings to N, we focus on the two special cases which correspond to two methods. In the first method, by setting N to 1, we consider $\bar{D}_S(v;1)$ where $C_S(1) = 1/(|S|-1)$, which coincides with a leave-one-out cross-validation. We refer to this method as the *LOO* method. In the second method, by setting N to $|S|/2$, we consider $\bar{D}_S(v;|S|/2)$ where $C_S(|S|/2) = \sqrt{1/(|S|-1)}$. This coefficient practically coincides with the value calculated from the central limit theorem. Thus, we refer to this method as the *CLT* method. In our experiments, we evaluate the performances of these two proposed methods.

3.2. Top-K Influential Node Identification

Next, we consider a problem of identifying the top-K nodes according to the influence degree $\langle A(v)\rangle$ of each node $v \in V$. For this purpose, we introduce a rank function, denoted by $R(v;A(\cdot))$, which returns the descending order of the node $v \in V$ according to the value of $A(v)$. By using this function, we can respectively express the true top-K nodes $T(K;\langle A(\cdot)\rangle)$ and the empirically estimated top-K nodes $T(K;\bar{A}_S(\cdot))$ obtained from $|S|$ times of simulation results as follows:

$$T(K;\langle A(\cdot)\rangle) = \{v \in V \mid R(v;\langle A(\cdot)\rangle) \le K\}, \tag{8}$$

$$T(K;\bar{A}_S(\cdot)) = \{v \in V \mid R(v;\bar{A}_S(\cdot)) \le K\}. \tag{9}$$

Then, for a given set of samples $\{A_s(v) \mid v \in V, s \in S\}$ generated by bond percolation processes, we attempt quantitatively evaluating the following expected precision of the estimated top-K nodes $T(K;\bar{A}_S(\cdot))$ of Equation (9) with respect to the true top-K nodes $T(K;\langle A(\cdot)\rangle)$ of Equation (8),

$$H_S(K) = \langle \frac{1}{K}|T(K;\langle A(\cdot)\rangle) \cap T(K;\bar{A}_S(\cdot))|\rangle. \tag{10}$$

Namely, we formulate our problem as estimating the top-K nodes identification precision between $T(K;\langle A(\cdot)\rangle)$ and $T(K;\bar{A}_S(\cdot))$ from a limited number of samples $\{A_s(v) \mid v \in V, s \in S\}$ without assuming $\langle A(v)\rangle$ in a typical machine learning setting.

To this end, we also propose methods based on a leave-N-out cross validation technique in a machine learning approach. As described above, by using $\mathcal{B} \subset 2^S$, a family of subsets of S whose number of elements is N, that is, $|B| = N$ for $B \in \mathcal{B}$, we can consider the following formula for estimating the identification precision of the top-K nodes:

$$\bar{H}_S(K;N) = \langle \frac{1}{K}|T(K;\bar{A}_S(\cdot)) \cap T(K;\bar{A}_{S\setminus B}(\cdot))|\rangle_{B\in\mathcal{B}}. \tag{11}$$

However, unlike the problem for estimating the approximation error of influence degree, we cannot derive effective calculation formulae with respect to Equation (11) for an arbitrary N. Thus, we focus on the two special cases which correspond to the following two methods. In the first method, by setting N to $|S|-1$, we consider the following estimation formula:

$$\bar{H}_S(K;|S|-1) = \frac{1}{K\times|S|}\sum_{s\in S}|T(K;\bar{A}_S(\cdot)) \cap T(K;\bar{A}_s(\cdot))|. \tag{12}$$

We refer to the method following this equation as the *SC* method because it simply counts how many times $v \in T(K; \bar{A}_S(\cdot))$ appears as a top-K node in each independent simulation for $s \in S$. On the other hand, in the second method, we consider changing N from 1 to S. Let $\bar{H}_S(K; \langle |S| \rangle)$ be the estimation formula for the second problem, then we can define it as follows:

$$\bar{H}_S(K; \langle |S| \rangle) = \frac{1}{K \times |S|} \sum_{s \in S} |T(K; \bar{A}_S(\cdot)) \cap T(K; \bar{A}_{\{1,\cdots,s\}}(\cdot))|. \tag{13}$$

We refer to the method following this equation as the *CC* method because it cumulatively counts the number of times that each $v \in V$ appears as a reachable node in the first s ($\in S$) simulations to calculate $\bar{A}_{\{1,\cdots,s\}}(\cdot)$ in the equation. In our experiments, we evaluate the performances of these two proposed methods.

4. Experiments

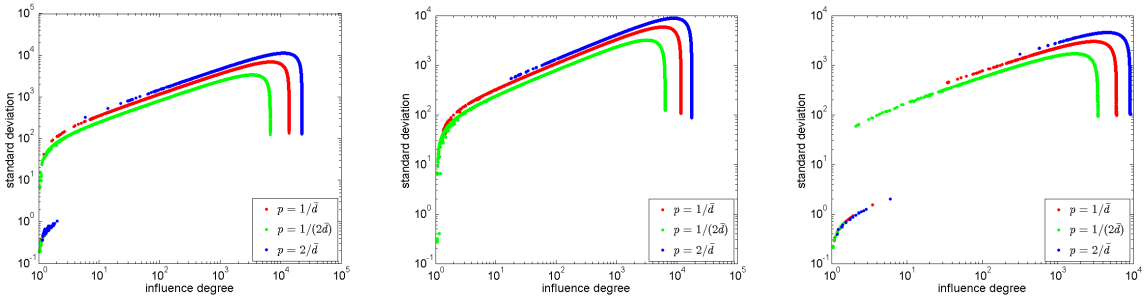
4.1. Datasets

To experimentally evaluate the methods proposed in the previous sections, we employed three datasets of real networks, where all networks are represented as directed graphs. The first one is a reader network extracted from a Japanese blog service site "Ameba"[2], in which each blog can have a list of reader links. A reader link is directional and a link is constructed from blog u to blog v if blog v registers blog u as her favorite one. We crawled the lists of 117,374 blogs of "Ameba" in June 2006, and extracted a large connected network that has 56,604 nodes and 734,737 directed links. We refer to this network as the Ameblo network. The second one is a network extracted from "@cosme"[3], a Japanese word-of-mouth communication site for cosmetics, in which each user page can have *fan links*. A fan link (u, v) means that user v registers user u as her favorite user. We traced up to ten steps in the fan-link network from a randomly chosen user in December 2009, and extracted a large connected network consisting of 45,024 nodes and 351,299 directed links. We refer to this directed network as the Cosme network. The last one is a network derived from the Enron Email Dataset (Klimt and Yang, 2004), in which an email address that appears in the dataset as either a sender or a recipient is regarded as a node and two email addresses u and v are linked by a directional link (u, v) if u sent an email to v. We refer to this directed network as the Enron network, which has 19,603 nodes and 210,950 links.

4.2. Statistical Analysis

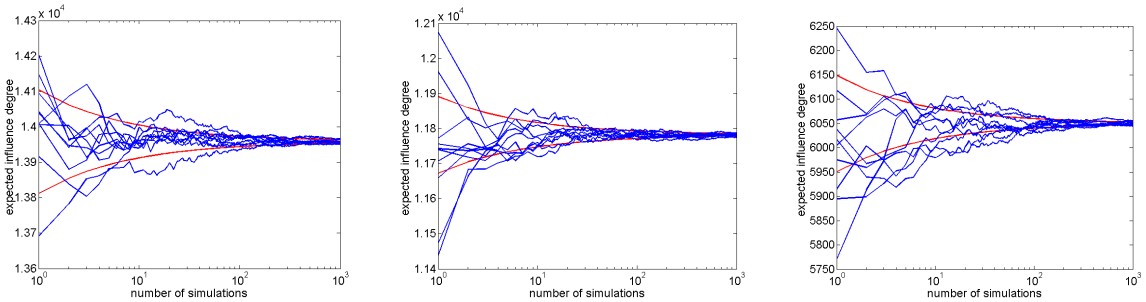
For each of the three real networks, $G = (V, E)$, we estimated the true influence degree $\langle A(v) \rangle$ of each node $v \in V$ by the empirical mean $\bar{A}_{S_0}(v)$ of one million simulations ($|S_0| = 1.0 \times 10^6$), according to Equation (1). We also estimated the true standard deviation $\sigma(v)$ of random variable $A(v)$ as $\bar{\sigma}_{S_0}(v)$, according to Equation (7). As described earlier, in our experiments, we used the bond percolation method instead of the direct simulations because of computation time issues. Figure 1 plots the pair $\left(\bar{A}_{S_0}(v), \bar{\sigma}_{S_0}(v)\right)$ for all $v \in V$ for the Ameblo, Cosme, and Enron networks, where the horizontal and vertical axes indicate influence degree $\bar{A}_{S_0}(v)$ and standard deviation $\bar{\sigma}_{S_0}(v)$, respectively. For the diffusion probability p of the IC model, we investigated the three cases, $p = 1/\bar{d}$, $p = 1/2\bar{d}$,

2. http://www.ameba.jp/
3. http://www.cosme.net/



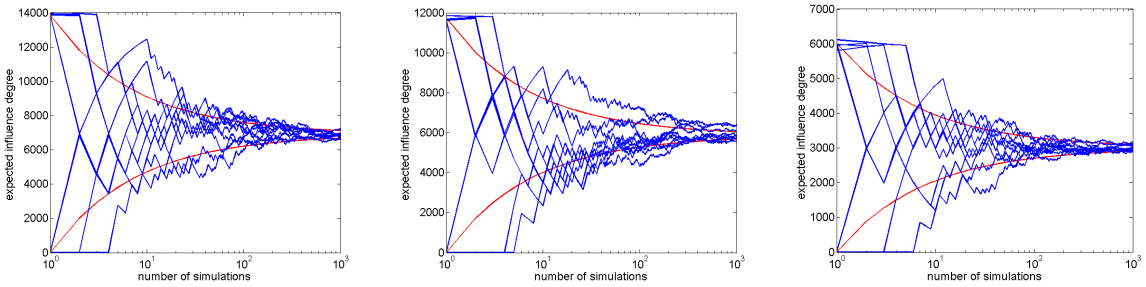
(*a*) Ameblo network (*b*) Cosme network (*c*) Enron network

Figure 1: Results for "influence degree vs. standard deviation" ($p = 1/\bar{d}$).



(*a*) Ameblo network (*b*) Cosme network (*c*) Enron network

Figure 2: Fluctuation of $\bar{A}_S(v_1)$ as a function of S.



(*a*) Ameblo network (*b*) Cosme network (*c*) Enron network

Figure 3: Fluctuation of $\bar{A}_S(v_*)$ as a function of S.

and $p = 2/\bar{d}$, where $\bar{d}$ means the average out-degree of the network. We can observe that all the results are qualitatively similar. Namely, there exists a critical influence degree $\bar{A}_{S_0}(v_*)$ such that the standard deviation $\bar{\sigma}_{S_0}(v)$ for a node v is an increasing function of its influence degree $\bar{A}_{S_0}(v)$ if $\bar{A}_{S_0}(v) \leq \bar{A}_{S_0}(v_*)$, but $\bar{\sigma}_{S_0}(v)$ is a rapidly decreasing function of $\bar{A}_{S_0}(v)$ if $\bar{A}_{S_0}(v) \geq \bar{A}_{S_0}(v_*)$. Note that the node v_* gives the largest standard deviation. It seems natural that the standard deviation for a node tends to become large as its influence degree becomes high. However, the experimental results

show that the standard deviations for the nodes having very high influence degree become relatively small. Let v_1 denote the node of the highest influence degree. Then, the ratio $\bar{\sigma}_{S_0}(v_1)/\bar{A}_{S_0}(v_1)$ is less than 0.01 for all the cases. These results imply that very many simulations are not required for estimating the influence degree of very influential nodes.

Next, for the nodes $v = v_1$ and $v = v_*$ in the case of $p = 1/\bar{d}$, we explored how $A_S(v)$ fluctuates as a function of the number of simulations $|S|$. We limit $|S|$ to $1,000$, and calculated $\bar{A}_{\{1,\cdots,s\}}(v)$ for $s \in S$. We repeated this procedure ten times. Figures 2 and 3 show the influence degree $\bar{A}_S(v)$ as a function of $|S|$ for the Ameblo, Cosme, and Enron networks, respectively. Here, we added the red curves defined by

$$|S| \mapsto A_{S_0}(v) \pm \frac{\bar{A}_{S_0}(v)}{\sqrt{|S|}}$$

as a guide since the central limit theorem states

$$\frac{\sqrt{|S|}\left(\bar{A}_S(v) - \langle A(v)\rangle\right)}{\sigma(v)} \to \mathcal{N}(0,1) \quad \text{in law} \quad \text{as } |S| \to \infty,$$

where $\mathcal{N}(0,1)$ denotes the normal distribution of mean 0 and variance 1. As expected, we can see that although the variance of $A_S(v_*)$ is large, the variance of $A_S(v_1)$ is not large when $|S|$ becomes near $1,000$.

4.3. Evaluation of Influence Degree Estimation

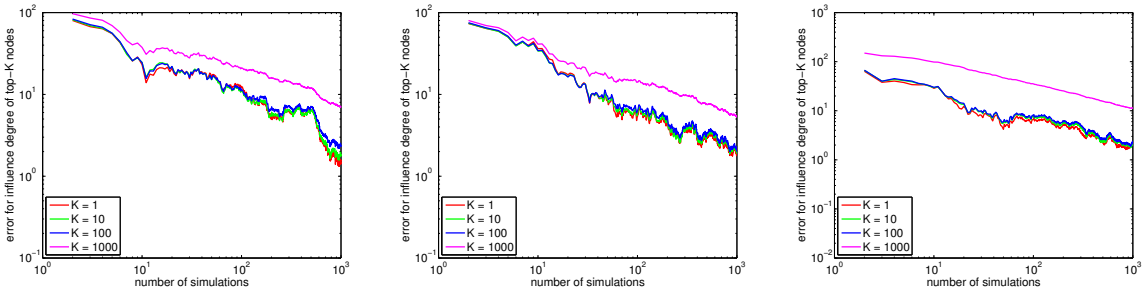
First, we evaluated the two methods proposed in Section 3.1, *i.e.*, the LOO and CLT methods, that estimate the approximation error between the true influence degree of node $v \in V$, $\langle A(v)\rangle$, and its estimation, $\bar{A}_S(v)$, that is the empirical mean of the influence degree of node v over the $|S|$ simulations. We conducted $|S|$ simulations to evaluate how accurately these methods can estimate the approximation error with the limited number of simulations, and compared the error estimated by these methods, $\bar{D}_S(v;N)$, with the true approximation error $D_S(v)$ for each $s \in S$. Since we are interested in the influential nodes, we focused on the top-K nodes in the true influence degree.

In fact, we repeated $|S|$ simulations M times, and investigated the empirical mean of the errors over the M trials. Let $D_{k,m}(s)$ and $\bar{D}_{k,m}(s)$ be the true and estimated approximation error of the k-th node in the first s simulations of the m-th trial of the M repeated $|S|$ simulations, respectively. Namely, in each trial, $D_{k,m}(s)$ is given by $D_{\{1,\cdots,s\}}(v_k)$ for the k-th node v_k, and $\bar{D}_{k,m}(s)$ is given by $\bar{D}_{\{1,\cdots,s\}}(v_k;N)$, where $N = 1$ for the LOO method and $N = s/2$ for the CLT method. It is desirable that we have an upper bound of the approximation error. Thus we seek for the condition that $D_{k,m}(s)/\bar{D}_{k,m}(s) < 1$ holds. Ideally $D_{k,m}(s)/\bar{D}_{k,m}(s)$ should be as close to 1 as possible. Thus, we used the following criterion to evaluate the LOO and CLT methods in this experiment:

$$\delta_{D_K}(s) = \frac{1}{M \times K} \sum_{m=1}^{M} \sum_{k=1}^{K} |1 - D_{k,m}(s)/\bar{D}_{k,m}(s)|. \tag{14}$$

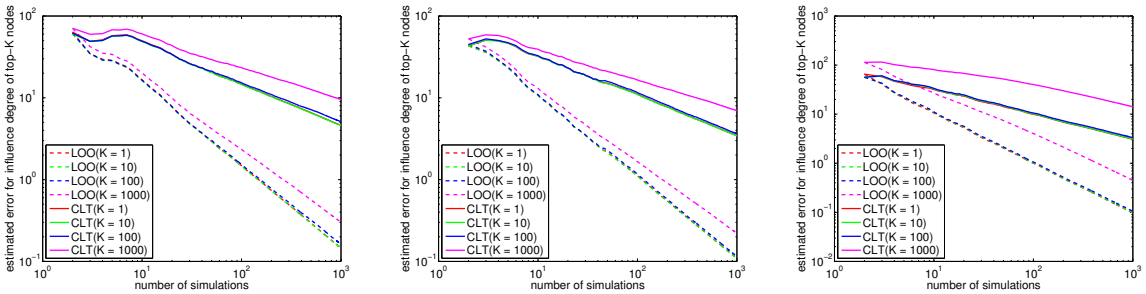
Obviously, the closer to 0 the value of $\delta_{D_K}(s)$ is, the better the performance of the corresponding method is.

Figures 4 and 5 depict the empirical mean of the true approximation error ($M^{-1}K^{-1}\sum_{m=1}^{M}\sum_{k=1}^{K} D_{k,m}(s)$) and its estimation ($M^{-1}K^{-1}\sum_{m=1}^{M}\sum_{k=1}^{K}\bar{D}_{k,m}(s)$) obtained by the LOO and CLT methods for the Ameblo, Cosme, and Enron networks, respectively, and Figure 6 shows the values of $\delta_{D_K}(s)$



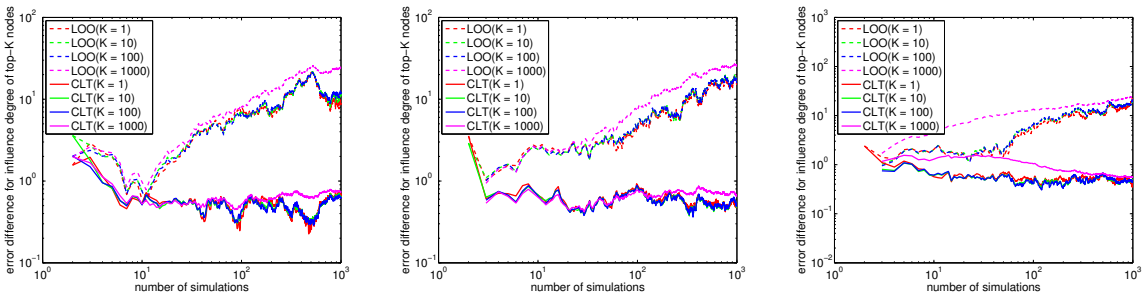
(*a*) Ameblo network (*b*) Cosme network (*c*) Enron network

Figure 4: The true approximation error of the estimated influence degree of the top-K nodes.



(*a*) Ameblo network (*b*) Cosme network (*c*) Enron network

Figure 5: The estimated approximation error of the estimated influence degree of the top-K nodes.



(*a*) Ameblo network (*b*) Cosme network (*c*) Enron network

Figure 6: The relative difference between the true and estimated approximation errors.

for the two methods. Here, we used $\bar{A}_{S_0}(v)$ obtained by one million simulations in Section 4.2 as the true influence degree of node v to calculate $D_{\{1,\cdots,s\}}(\cdot)$, and investigated for different values of $K = 10^0, 10^1, 10^2, 10^3$. As in Section 4.2, we limited $|S|$ to 1,000, set M to 10 and the diffusion probability to $p = 1/\bar{d}$ for every network. It is noted that we cannot estimate the approximation

error based on Equation (5) when $|S| = 1$ because $\bar{\sigma}_S(v)$ defined by Equation (7) becomes 0. Thus, every line in these figures starts from $s = 2$.

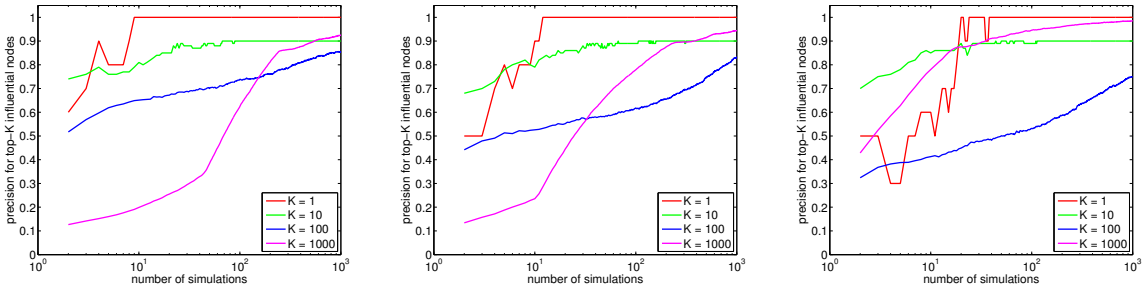
From Figure 4, we can observe common tendencies among all networks that 1) the approximation errors become smaller as s gets larger regardless of the value of K, 2) it is hard to distinguish the lines for $K = 10^0, 10^1, 10^2$ from one another, while the line for $K = 10^3$ is clearly distinguishable from the others and always shows larger errors. These results are consistent with the analytical results in Section 4.2 in which the expected influence degree of a node converges to the true value as s becomes large as shown in Figures 2 and 3, while the influence degree of a lower-ranked node has a larger variance compared to a higher-ranked node as shown in Figure 1, which implies that it becomes harder to estimate the approximation error for a large K. The similar tendencies can be observed in Figure 5.

Besides, by comparing Figures 4 and 5, it is found that the LOO method tends to underestimate the approximation error compared to the CLT method regardless of the value of K for every network. This tendency is quantitatively confirmed in Figure 6, in which the value of $\delta_{D_K}(s)$ for the CLT method becomes less than 1 after the first few simulations except for the case of $K = 10^3$ for the Enron network, while that of the LOO method becomes larger than 1 as s increases in all cases. The reason why the performance of both the methods is relatively worse in the case of $K = 10^3$ for the Enron network compared to the other networks is that the top 1,000 nodes of the Enron network include more nodes that have large variance with respect to the true influence degree compared to the others. Actually, the true influence degree and its standard deviation of the 1,000-th node for the Enron network are $5,750.461$ and $1,209.592$, respectively, while the corresponding values for the Ameblo network and Cosme network are $13,817.63$ and 750.9033 and $11,697.17$ and 624.5759, respectively. This makes it harder to estimate the approximation error for the Enron network.

It is evident that the CLT method outperforms the LOO method in terms of predicting the approximation error of the estimated influence degree. Indeed, for the networks and the parameters K and M we used, the approximation error of the CLT method is the upper bound of the error. This may not necessarily generalize to other networks, but we can say that CLT is a good measure to estimate the approximation error. It is noted that Equation (7) can be calculated independently of the value of N. Thus, the estimation given by Equation (5) is actually determined by the coefficient defined by Equation (6). Our experiment shows that $N = 1$ (LOO) is too small and $N = |S|/2$ (CLT) is the recommended value. In other words, if we set a threshold of the estimated error to terminate the repeated simulations, the LOO method stops earlier than the CLT method, but the actual error of the LOO method is worse than the threshold, while the actual error of the CLT method is expected to be less than the threshold.

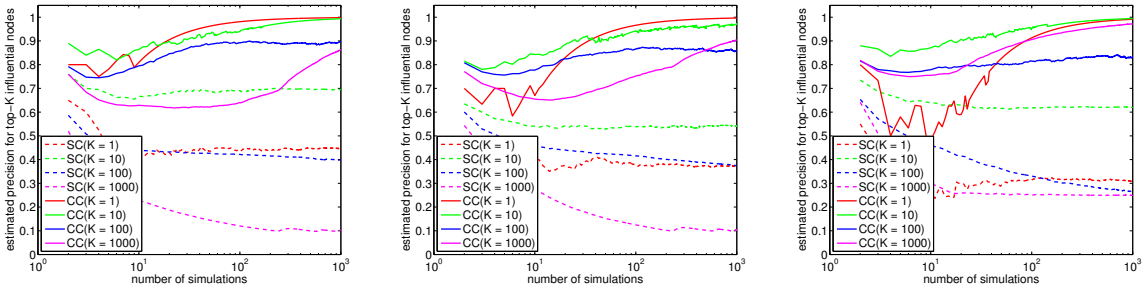
4.4. Evaluation of Top-K Influential Node Identification

Next, we evaluated the two methods proposed in Section 3.2, the SC and CC methods, that estimate the precision of the top-K nodes based on a limited number of simulation results. To this end, we compared the precision estimated by these methods, $\bar{H}_S(K; N)$, with the true precision $H_S(v)$ under the same settings as in Section 4.3, *i.e.*, $M = 10$ trials of $|S| = 1,000$ simulations. Here, let $H_{K,m}(s)$ and $\bar{H}_{K,m}(s)$ be the true and estimated precision of the top-K nodes identified in the first s simulations of the m-th trial, respectively. Namely, in each trial, $H_{K,m}(s)$ is given by $H_{\{1,\cdots,s\}}(K)$, while $\bar{H}_{k,m}(s)$ is given by $\bar{H}_{\{1,\cdots,s\}}(K; N)$, where $N = s - 1$ for the CC method and $N = \langle s \rangle$ for the CC method. We directly evaluated the empirical mean of the difference between the two defined as



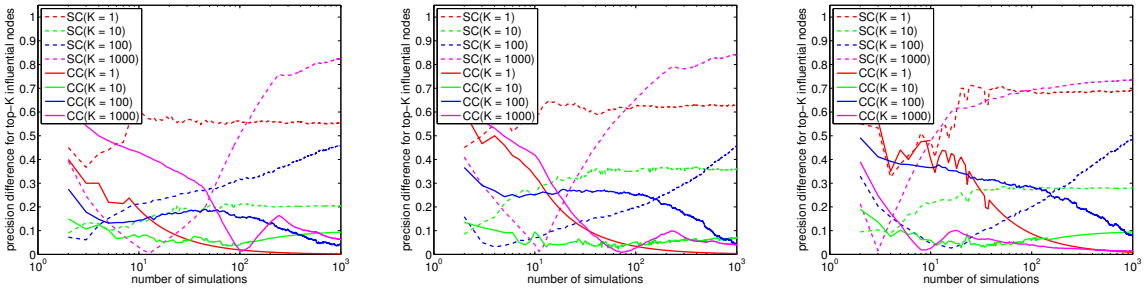
(*a*) Ameblo network (*b*) Cosme network (*c*) Enron network

Figure 7: The true precision for the top-K nodes prediction.



(*a*) Ameblo network (*b*) Cosme network (*c*) Enron network

Figure 8: The estimated precision for the top-K nodes prediction.



(*a*) Ameblo network (*b*) Cosme network (*c*) Enron network

Figure 9: The difference between the true and estimated precisions for the top-K nodes prediction.

follows:

$$\delta_{H_K}(s) = \frac{1}{M}\sum_{m=1}^{M} |H_{K,m}(s) - \bar{H}_{K,m}(s)|. \tag{15}$$

It is obvious that the smaller the value of $\delta_{H_K}(s)$ is, the better the performance of the corresponding method is.

Figures 7, 8 and 9 illustrate the empirical mean of the true precision ($M^{-1}\sum_{m=1}^{M} H_{K,m}(s)$), its estimation ($M^{-1}\sum_{m=1}^{M} \bar{H}_{K,m}(s)$) obtained by the SC and CC methods, and the values of $\delta_{H_K}(s)$ for the two methods for the Ameblo, Cosme, and Enron networks, respectively. Roughly speaking, from Figure 7, we can observe that 1) the true precision approaches asymptotically to 1 as s increases, 2) especially, conducting only about 10 simulations seems to be enough to estimate the best node ($K = 1$) , and 3) identifying the top 100 nodes seems to be somehow more difficult than the other cases. The Enron network shows slightly different tendencies compared to the others as in the previous section. This network needs more simulations than the other networks to correctly identify the best node v_1. This is attributed to the difference shown in Figure 2, in which the expected influence degree of node v_1 for the Enron network fluctuates more compared to the other networks until about the first 10^2 simulations. On the other hand, the precision for the Enron network is better than the other networks for $K = 10^3$. This is because in the Ameblo and Cosme networks the nodes ranked around 1000-th in the true influence degree have influence degree close to each other, and thus they are subject to switch ranks across the boundary $K = 10^3$ in each simulation. On the other hand, in the Enron network, the influence degree of these nodes are not so close to each other, and thus such swaps do not occur so often, which leads to the better performance shown in Figure 7.

Next, from Figure 8, we can observe that the precision estimated by the CC method approaches to 1 as s becomes large for $K = 10^0, 10^1, 10^3$. However, for $K = 10^2$, it falls within the range of 0.8 to 0.9 and these are common to all networks. Somehow, the nodes around the 10^2th rank have similar influence degree for each of the three networks and the opposite phenomenon mentioned above ($K = 10^3$ for the Enron network) is happening. In contrast to these observations, we see that the precision estimated by the SC method is not improved as s becomes larger for any value of K for every network. Indeed, by comparing Figures 7 and 8, we note that the SC method overly underestimates the precision for a large s. This difference is quantitatively revealed in Figure 9, in which $\delta_{H_K}(s)$ for the CC method becomes less than 0.1 at around $s = 10^2$ for $K = 10^0, 10^1, 10^3$ and at around $s = 10^3$ for $K = 10^2$ for every network, while for the SC method it becomes larger as s increases and does not approach to 0. This is because, as shown in Figure 4, for a large $s \in S$, $\bar{A}_{\{1,\cdots,s\}}(v)$ used in the CC method gives a better approximation of the true influence degree of node v compared to $A_s(v)$ used in the SC method that considers the result of a single simulation example in isolation.

Consequently, these results suggest that for a small K, say 10, we could obtain a good approximation of the true top-K nodes by running far fewer simulations using the CC method and an appropriate threshold for the precision. Note that if we set the threshold to 0.9 for $K = 10$, from Figure 8 the repeated simulations would stop before around $s = 20$, and from Figure 6 the value of $\delta_{D_K}(s)$ for the same s is around 0.5 for the CLT method, which means that the error estimated by the CLT method is about twice as large as the true approximation error. This suggests that the top-K nodes can be estimated with less effort compared to their influence degree. In other words, estimating the top-K nodes is easier than estimating the influence degree of a node.

5. Conclusion

In this paper, we addressed a problem of estimating the influence of a node in terms of information diffusion over a social network. It is crucial to efficiently and effectively calculate the influence degree because node influence is an important ingredient to solve many practical problems in social network analysis such as the influence maximization problem. The difficulty involved in estimating

the influence degree comes from the stochastic nature of information diffusion. The influence degree of a node is the expected number of nodes that are influenced by the node as the result of information diffusion process. It is normally approximated by the empirical mean of the many runs of simulation results, which is very time consuming.

We proposed a framework for predictive simulation, which enable us to evaluate how many runs of simulations are required to ensure the accuracy without knowing the true answer. We provided two measures, one for estimating influential degree of individual node and the other for identifying the top-K influential nodes. The method is based on leave-N-out cross validation technique. It approximates the expected difference of the estimated answer from the unknown true answer, *i.e.*, the approximation error of the estimated influence degree of individual node and the precision of the estimated top-K nodes.

We have conducted extensive experiments on three real world networks varying the number of simulations and evaluated the proposed framework. In case of influence degree estimation, the recommended value for N is the half of the number of simulations. Use of this value provides a good measure. In our experiments, the measure provides an upper bound of the error. In case of top-K influential node identification, cumulative handling of N rather than using a single value for N provides a good measure. Estimating the top-K nodes is easier than estimating the influence degree, which implies we can identify the influential nodes without knowing accurately their influence degrees. These results suggest that we can stop running simulations earlier ensuring the predictive performance by providing an appropriate threshold as a stopping criterion for either the estimated approximation error or the estimated top-K precision, or both. It is noted that the framework we proposed is not specific to information diffusion models. Indeed, it is very generic and applicable to any other estimation problems in which such a criterion to avoid unnecessarily running extra simulations is required. However, the method is empirical and does not theoretically guarantee the error upper bound. We have yet to test out the proposed method in a broader setting and also in different domains, too.

Acknowledgments

This work was partly supported by Asian Office of Aerospace Research and Development, Air Force Office of Scientific Research under Grant No. AOARD-13-4042, and JSPS Grant-in-Aid for Young Scientists (B) (No. 23700181).

References

E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts. Everyone's an influencer: Quantifying influence on twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining (WSDM'11)*, pages 65–74, 2011.

W. Chen, Y. Wang, and S. Yang. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09)*, pages 199–208, 2009.

W. Chen, C. Wang, and Y. Wang. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2010)*, pages 1029–1038, 2010a.

W. Chen, Y. Yuan, and L. Zhang. Scalable influence maximization in social networks under the linear threshold model. In *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM 2010)*, pages 88–97, 2010b.

P. Cui, F. Wang, S. Yang, and L. Sun. Item-level social influence prediction with probabilistic hybrid factor matrix factorization. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2011)*, pages 331–336, 2011.

A. Goyal, W. Lu, and L.V. Lakshmanan. Celf++: optimizing the greedy algorithm for influence maximization in social networks. In *Proceedings of the 20th International World Wide Web Conference (WWW2011)*, pages 47–48, 2011.

A. Guille and H. Hacid. A predictive model for the temporal dynamics of information diffusion in online social networks. In *Proceedings of the 21st international conference companion on World Wide Web (WWW'12)*, pages 1145–1152, 2012.

D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2003)*, pages 137–146, 2003.

M. Kimura and K. Saito. Tractable models for information diffusion in social networks. In *Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2006)*, pages 259–271. LNAI 4213, 2006.

M. Kimura, K. Saito, and R. Nakano. Extracting influential nodes for information diffusion on a social network. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence (AAAI-07)*, pages 1371–1376, 2007.

M. Kimura, K. Saito, and H. Motoda. Minimizing the spread of contamination by blocking links in a network. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI-08)*, pages 1175–1180, 2008.

M. Kimura, K. Saito, and H. Motoda. Efficient estimation of influence functions fot sis model on social networks. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI-09)*, 2009a.

M. Kimura, K. Saito, and H. Motoda. Blocking links to minimize contamination spread in a social network. *ACM Transactions on Knowledge Discovery from Data*, 3:9:1–9:23, 2009b.

M. Kimura, K. Saito, R. Nakano, and H. Motoda. Extracting influential nodes on a social network for information diffusion. *Data Min. Knowl. Disc.*, 20:70–97, 2010.

J. Kleinberg. The convergence of social and technological networks. *Communications of ACM*, 51 (11):66–72, 2008.

B. Klimt and Y. Yang. The enron corpus: A new dataset for email classification research. In *Proceedings of the 2004 European Conference on Machine Learning (ECML'04)*, pages 217–226, 2004.

J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2007)*, pages 420–429, 2007.

M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

H. Nguyen and R. Zheng. Influence spread in large-scale social networks - a belief propagation approach. In *Proceedings of the 2012 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2012)*, pages 515–530. LNAI 7524, 2012.

K. Saito, M. Kimura, and H. Motoda. Discovering influential nodes for sis models in social networks. In *Proceedings of the Twelfth International Conference of Discovery Science (DS2009)*, pages 302–316. Springer, LNAI 5808, 2009a.

K. Saito, M. Kimura, K. Ohara, and H. Motoda. Learning continuous-time information diffusion model for social behavioral data analysis. In *Proceedings of the 1st Asian Conference on Machine Learning (ACML2009)*, pages 322–337. LNAI 5828, 2009b.

K. Saito, M. Kimura, K. Ohara, and H. Motoda. Selecting information diffusion models over social networks for behavioral analysis. In *Proceedings of the 2010 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2010)*, pages 180–195. LNAI 6323, 2010.

K. Saito, M. Kimura, K. ohara, and H. Motoda. Which targets to contact first to maximize influence over social network. In *Proceedings of the 6th International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction (SBP2013)*, pages 359–367. LNCS 7812, 2013.

D. J. Watts and P. S. Dodds. Influence, networks, and public opinion formation. *Journal of Consumer Research*, 34:441–458, 2007.

J. Yang and S. Counts. Predicting the speed, scale, and range of information diffusion in twitter. In *Proceedings of the Fourth International Conference on Weblogs and Social Media (ICWSM 2010)*, 2010.

J. Yang and J. Leskovec. Modeling information diffusion in implicit networks. In *Proceedings of the 2010 IEEE International Conference on Data Mining (ICDM'10)*, pages 599–608, 2010.

Y. Yang, E. Chen, Q. Liu, B. Xiang, T. Xu, and S.A. Shad. On approximation of real-world influence spread. In *Proceedings of the 2012 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2012)*, pages 548–564. LNAI 7524a, 2012.