Extracting Influential Nodes for Information Diffusion on a Social Network

Masahiro Kimura
Dept. of Electronics and Informatics
Ryukoku University
Otsu 520-2194, Japan
kimura@rins.ryukoku.ac.jp

Kazumi Saito
NTT Communication Science Labs.
NTT Corporation
Kyoto 619-0237, Japan
saito@cslab.kecl.ntt.co.jp

Ryohei Nakano
Dept. of Computer Science
and Engineering
Nagoya Institute of Technology
Nagoya 466-8555, Japan
nakano@ics.nitech.ac.jp

Abstract

We consider the combinatorial optimization problem of finding the most influential nodes on a large-scale social network for two widely-used fundamental stochastic diffusion models. It was shown that a natural greedy strategy can give a good approximate solution to this optimization problem. However, a conventional method under the greedy algorithm needs a large amount of computation, since it estimates the marginal gains for the expected number of nodes influenced by a set of nodes by simulating the random process of each model many times. In this paper, we propose a method of efficiently estimating all those quantities on the basis of bond percolation and graph theory, and apply it to approximately solving the optimization problem under the greedy algorithm. Using real-world large-scale networks including blog networks, we experimentally demonstrate that the proposed method can outperform the conventional method, and achieve a large reduction in computational cost.

Introduction

A social network is the network of relationships and interactions among social entities such as indivisuals, groups of indivisuals, and organizations. Since the rise of the Internet and the World Wide Web has enabled us to investigate large-scale social networks, there has been growing interest in social network analysis (Newman 2001; McCallum, Corrada-Emmanuel, & Wang 2005; Leskovec, Adamic, & Huberman 2006).

Since a piece of information can propagate from one node to another node through a link on a social network in the form of "word-of-mouth" communication, it is an important research issue to find influencial nodes for the spread of information through a network represented by a directed graph in terms of sociology and "viral marketing". In fact, researchers have recently studied a combinatorial optimization problem called the *influence maximization problem* (Domingos & Richardson 2001; Richardson & Domingos 2002; Kempe, Kleinberg, & Tardos 2003). This 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 for a given integer k. To consider this optimization problem, a model for the process by which a certain information propagates on a social network must be specified. In this paper, we examine the influence maximization problem for two widely-used fundamental information diffusion models, the *independent cascade (IC) model* (Goldenberg, Libai, & Muller 2001; Kempe, Kleinberg, & Tardos 2003; Gruhl *et al.* 2004) and the *linear threshold (LT) model* (Watts 2002; Kempe, Kleinberg, & Tardos 2003).

Kempe, Kleinberg, and Tardos (2003) experimentally showed on large collaboration networks that for the influence maximization problem in the IC and LT models, the greedy hill-climbing algorithm significantly outperforms the high-degree and centrality heuristics that are commonly used in the sociology literature. Moreover, they mathematically proved a performance guarantee of this greedy algorithm for these diffusion models by using an analysis framework based on submodular functions. The greedy algorithm requires computing the vector $\nabla\sigma(A)$ that consists of all the marginal gains for the *influence degree* $\sigma(A)$ given a set A of nodes. Here, the IC and LT models have stochastic nature, and $\sigma(A)$ is defined as the expected number of nodes influenced by the nodes in A. However, it is an open question to compute influence degrees exactly by an efficient method, and so good estimates were obtained by simulating the random process of each model many times. Thus, solving the influence maximization problem under the greedy algorithm needed a large amount of computation.

In this paper, we propose a method of efficiently estimating all the marginal gains $\nabla\sigma(A)$ for influece degree $\sigma(A)$ on the basis of bond percolation and graph theory, and apply it to approximately solving the influence maximization problem under the greedy algorithm. Using real large-scale networks including blog networks, we experimentally evaluate the effectiveness of the proposed method. We finally discuss some related work and set out the conclusion.

Preliminaries

First, we recall some basic notions from graph theory. Next, we define the IC and LT models. Moreover, we define the influence maximization problem, and describe the greedy hill-climbing algorithm for solving the problem.

Graphs

A directed *graph* G is a pair (V, E), where V is a set of nodes (or *vertices*) and $E \subset V \times V$ is a set of directed links

(or *edges*). If there is a directed link (u, v) from node u to node v, node v is called a *child* of node u and node u is called a *parent* of node v. For a subset V' of V, graph $G' = (V', E')$ is called the *induced graph* of G to V' if $E' = E \cap (V' \times V')$.

We call $(u_0, \cdots, u_\ell)$ a *path* from node u_0 to node u_ℓ if we have $(u_{i-1}, u_i) \in E$ $(i = 1, \cdots, \ell)$. We say that node u can *reach* node v or node v is *reachable* from node u if there is a path from node u to node v. For a node v of the graph G, we define $F(v; G)$ to be the set of nodes that are reachable from v, and define $B(v; G)$ to be the set of nodes that can reach v. For $A \subset V$, we set

$$F(A; G) = \bigcup_{v \in A} F(v; G),\ B(A; G) = \bigcup_{v \in A} B(v; G).$$

A *strongly connected component (SCC)* of G is a maximal subset C of V such that for all u, $v \in C$ there is a path from u to v. For a node v of G, we define $SCC(v; G)$ to be the SCC that contains v.

Fundamental Diffusion Models

Throughout this paper, we discuss the spread of a certain information through a social network represented by a directed graph $G = (V, E)$. We call nodes *active* if they have accepted the information. Let N denote the number of nodes in V, and L denote the number of links in E. Let $\Gamma(v)$ denote the set of parent nodes of $v \in V$.

According to the work of Kempe, Kleinberg, and Tardos (2003), we define the IC and LT models on G. In these models, the diffusion processes unfold in discrete time-steps $t \geq 0$, and it is assumed that nodes can switch from being inactive to being active, but cannot switch from being active to being inactive. Given an initial set A of active nodes, we assume that the nodes in A have first become active at step 0, and all the other nodes are inactive at step 0.

Independent Cascade Model First, we define the IC model. In this model, we must specify a real value $p_{u,v} \in [0, 1]$ for each directed link (u, v) in advance. Here, $p_{u,v}$ is referred to as the *propagation probability* through link (u, v). When an initial set A of active nodes is given, the diffusion process proceeds in the following way. When node u first becomes active at step t, it is given a single chance to activate each currently inactive child v, and succeeds with probability $p_{u,v}$. If u succeeds, then v will become active at step $t+1$. If multiple parents of v first become active at step t, then their activation attempts are sequenced in an arbitrary order, but performed at step 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.

For an initial active set A, let $\sigma(A)$ denote the expected number of active nodes at the end of the random process in the IC model. We call $\sigma(A)$ the *influence degree* of target set A.

Linear Thresholod Model Next, we define the LT model. In this model, for any node $v \in V$, we must specify a *weight* $w_{u,v}$ (> 0) from its parent node u such that $\sum_{u \in \Gamma(v)} w_{u,v} \leq 1$. When an initial set A of active nodes is given, and a *threshold* θ_v of each node v is chosen uniformly at random from the interval $[0, 1]$, the diffusion process deterministically proceeds in the following way. A node v that is inactive at step t is influenced by each parent u that is active at step t according to weight $w_{u,v}$. Let $\Gamma_t(v)$ denote the set of parent nodes of v that are active at step t. If the total weight from active parents is at least threshold θ_v, that is, $\sum_{u \in \Gamma_t(v)} w_{u,v} \geq \theta_v$, then v will become active at step $t+1$. The process terminates if no more activations are possible.

Note that the threshold θ_v models the tendency of node v to accept the information when its parents do. Since it is generally difficult to specify such thresholds for a real-world network in advance, we choose them randomly. When we estimate the influence of a target set, we average over possible threshold values for all the nodes. Therefore, we regard the LT model as a stochastic model associated with the uniform distribution on $[0, 1]^N$.

Suppose that A is an initial set of active nodes. Let $\sigma(A)$ denote the expected number of final active nodes for the random process from A under the LT model. We call $\sigma(A)$ the *influence degree* of target set A.

Influence Maximization Problem

We investigate the influence maximization problem for the IC and LT models. The problem is defined as follows: Given a positive integer k, find a set A_k^* of k nodes to target for initial activation such that $\sigma(A_k^*) \geq \sigma(B)$ for any set B of k nodes. To approximately solve this optimization problem, we consider the following greedy hill-climbing algorithm:

1. Set $A \leftarrow \emptyset$.
2. **for** $i = 1$ to k **do**
3. Choose a node $v_i \in V$ maximizing $\sigma(A \cup \{v\})$.
4. Set $A \leftarrow A \cup \{v_i\}$.
5. **end for**

Let A_k denote the set of k nodes obtained by this algorithm. Then, it is known that $\sigma(A_k) \geq (1 - 1/e)\ \sigma(A_k^*)$, that is, a performance guarantee of the approximate solution A_k is obtained (Kempe, Kleinberg, & Tardos 2003).

To implement this greedy algorithm, we need a method of evaluating the N-dimensional vector $\nabla\sigma(A)$,

$$\nabla\sigma(A) = (\sigma(A \cup \{v\}))_{v \in V} \in \mathbf{R}^N,$$

that consists of all the marginal gains for influence degree $\sigma(A)$. Since it is not clear how to evaluate $\nabla\sigma(A)$ exactly by an efficient method, a good estimate was conventionally obtained by simulating the random process of each model many times (Kempe, Kleinberg, & Tardos 2003). However, as shown in the experiments, the greedy algorithm based on this estimation method of $\nabla\sigma(A)$ needs a large amount of computation for solving the influence maximization problem on a large-scale network.

Proposed Method

We propose a method of efficiently estimating $\nabla\sigma(A)$ for $A \subset V$ on the basis of bond percolation and graph theory, and practically solve the influence maximization problem on $G = (V, E)$ under the greedy hill-climbing algorithm.

Bond Percolation

A *bond percolation* process on G is the process in which each link of G is randomly designated either "occupied " or "unoccupied" according to some probability distribution. Let us consider the following set of L-dimensional vectors,

$$R_G = \left\{ r = (r_{u,v})_{(u,v)\in E} \in \{0,1\}^L \right\}.$$

A bond percolation process on G is determined by a probability distribution q on R_G. Namely, for a random vector $r \in R_G$ drawn from q, each link $(u,v) \in E$ is designated "occupied" if $r_{u,v} = 1$, and it is designated "unoccupied" if $r_{u,v} = 0$. Let E_r denote the set of occupied links for $r \in R_G$, and let G_r denote the graph (V, E_r). For each $r \in R_G$, we can consider the deterministic diffusion model $\mathcal{M}_r$ on G_r such that $F(A; G_r)$ becomes the final set of active nodes when A is an initial set of active nodes. By associating the diffusion model $\mathcal{M}_r$ on G_r with a probability distribution q on R_G, we define a stochastic diffusion model on G. We call this diffusion model the *bond percolation model* on G, and refer the probability distribution q on R_G to as the *occupation probability distribution* of the model.

We easily see that the IC model on G can be identified with the so-called *susceptible/infective/recovered (SIR) model* (Newman 2003) for the spread of a disease on G, where the nodes that have just become active at step t in the IC model correspond to the infective nodes at step t in the SIR model. It is known that the SIR model on a network can be exactly mapped onto a bond percolation model on the same network (Newman 2002; 2003). Hence, we see that the IC model on G is equivalent to some bond percolation model on G, that is, these two models have the same probability distribution for the final set of active nodes given a target set. Here, for the IC model on G, the occupation probability distribution q of the corresponding bond percolation model is given by

$$q(r) = \prod_{(u,v)\in E} \left\{ (p_{u,v})^{r_{u,v}} (1 - p_{u,v})^{1-r_{u,v}} \right\} \quad (r \in R_G),$$

that is, each link (u,v) of G is independently declared to be "occupied" with probability $p_{u,v}$. Here, $p_{u,v}$ is the propagation probability through link (u,v) in the IC model.

On the other hand, to derive the result that the influece degree function σ is submodular in the LT model, Kempe, Kleinberg, and Tardos (2003) essentially proved that the LT model on G can also be equivalent to some bond percolation model on G. Here, for the LT model on G, the corresponding occupation probability distribution q is generated by declaring "occupied" and "unoccupied" links in the following way: For any $v \in V$, we pick at most one of the incoming links to v by selecting link (u,v) with probability $w_{u,v}$ and selecting no link with probability $1 - \sum_{u\in\Gamma(v)} w_{u,v}$. After this process, the picked links are declared to be "occupied" and other links are declared to be "unoccupied". Here, $w_{u,v}$ is the weight of link (u,v) in the LT model.

Estimation Method

Now, we present a method of efficiently estimating all the marginal gains $\nabla\sigma(A)$ for the influence degree $\sigma(A)$ of target set $A \subset V$ under the IC and LT models. As shown in the preceding section, the IC and LT models on G can be identified with bond percolation models on G. Therefore, we have

$$\sigma(A \cup \{v\}) = \sum_{r\in R_G} q(r) \, |F(A \cup \{v\}; G_r)|$$

for any $v \in V$, where q is the corresponding occupation probability distribution, and $|S|$ denotes the number of elements in a subset S of V. For a sufficiently large positive integer M, let $\{r_1, \cdots, r_M\}$ be a set of sample vectors drawn independently from the probability distribution q on R_G. Then, we can approximate the influence degree $\sigma(A \cup \{v\})$ for $v \in V$ by

$$\sigma(A \cup \{v\}) \simeq \frac{1}{M} \sum_{m=1}^{M} |F(A \cup \{v\}; G_{r_m})|. \qquad (1)$$

Basically, we consider estimating $\nabla\sigma(A)$ by using Equation (1). To estimate it more efficiently, we propose an algorithm of evaluating simultaneously all the influence sizes $\{|F(A \cup \{v\}; G_{r_m})|;\ v \in V\}$ for each graph G_{r_m}. One of the key ideas is to apply a symbolic algorithm for SCC decomposition (Xie & Beerel 2000).

To evaluate $\{|F(A \cup \{v\}; G_r)|;\ v \in V\}$ for an arbitrary $r \in R_G$, we use the following algorithm:

1. Compute the subset $F(A; G_r)$ of V.
2. Set $|F(A \cup \{v\}; G_r)| \leftarrow |F(A; G_r)|$ for an arbitrary $v \in F(A; G_r)$.
3. Set $U \leftarrow \emptyset$.
4. Compute the subset $V_r^A = V \setminus F(A; G_r)$ of V, and the induced graph G_r^A of G_r to V_r^A.
5. **while** $V_r^A \setminus U \neq \emptyset$ **do**
6. Pick a node $u \in V_r^A \setminus U$.
7. Compute the subset $F(u; G_r^A)$ of V_r^A.
8. Compute the subset $C(u; G_r^A) = B(F(u; G_r^A); G_r^A) \cap F(u; G_r^A)$ of $F(u; G_r^A)$.
9. Set $|F(A \cup \{v\}; G_r)| \leftarrow |F(u; G_r^A)| + |F(A; G_r)|$ for an arbitrary $v \in C(u; G_r^A)$.
10. Set $U \leftarrow U \cup C(u; G_r^A)$.
11. **end while**

In this algorithm, we attempt to achieve a reduction in computational cost by exploiting the following facts. First, in Step 2, we use the fact that if $v \in F(A; G_r)$, the set $F(A \cup \{v\}; G_r)$ that is reachable from $A \cup \{v\}$ is equal to the set $F(A; G_r)$. Next, from Step 4 to Step 11, we use the fact that if $v \notin F(A; G_r)$, the influence size $|F(A \cup \{v\}; G_r)|$ is obtained by the sum of $|F(A; G_r)|$ and $|F(v; G_r^A)|$. This fact enables us to reduce the graph in question from G_r to G_r^A. In Step 8, we note that the set $C(u; G_r^A)$ is equal to the SCC $SCC(u; G_r^A)$ that contains u. Moreover, in Step 9, we use the fact if node v belongs to the same SCC $C(u; G_r^A)$ as node u, the influnce size of v on graph G_r^A is equal to that of u, that is, $|F(v; G_r^A)| = |F(u; G_r^A)|$.

Experimental Evaluation

Using real large-scale networks, we experimentally evaluated the performance of the proposed method for solving the influence maximization problem in the IC and LT models under the greedy hill-climbing algorithm.

Network Dataset

In the evaluation experiments, we should desirably use large-scale networks that exhibit many of the key features of real social networks. Here, we report on the experimental results for two different datasets of such real networks.

First, we employed a trackback network of blogs, since a piece of information can propagate from one blog author to another blog author through a trackback. By tracing ten steps ahead the trackbacks from the blog of the theme "JR Fukuchiyama Line Derailment Collision" in the site "goo" (`http://blog.goo.ne.jp/usertheme/`), we collected a large connected trackback network in May, 2005. This network was a directed graph of $12,047$ nodes and $53,315$ links, and showed the so-called "power-law" distributions for the out-degree and in-degree that most real large networks exhibit. Here, the out-degree and in-degree distributions are the distributions of the number of outgoing and incoming links for every node, respectively. We call this network data the blog dataset.

Next, we employed a network of people that was derived from the "list of people" within Japanese Wikipedia. Specifically, we extracted the maximal connected component of the undirected graph obtained by linking two people in the "list of people" if they co-occur in six or more Wikipedia pages, and constructed a directed graph by regarding those undirected links as bidirectional ones. We call this network data as the Wikipedia dataset. Here, the total numbers of nodes and directed links were $9,481$ and $245,044$, respectively.

Newman and Park (2003) observed that social networks represented as undirected graphs generally have the following two statistical properties unlike non-social networks. First, they show positive correlations between the degrees of adjacent nodes. Second, they have much higer values of the *clustering coefficient* than the corresponding *configuration models* (i.e., random network models). Here, the clustering coefficient C for an undirected graph is defined by

$$C = \frac{3 \times \text{number of triangles on the graph}}{\text{number of connected triples of nodes}},$$

where a "triangle" means a set of three nodes each of which is connected to each of the others, and a "connected triple" means a node connected directly to an unordered pair of others. Note that in terms of sociology, C measures the probability that two of your friends will also be friends of one another. Given a degree distribution, the corresponding configuration model of random network is defined as the ensemble of all possible graphs that possess the degree distribution, with each having equal weight. The value of C for the configuration model can be exactly calculated (Newman 2003). For the undirected graph of the Wikipedia dataset, the value of C of the corresponding configuration model was 0.046, while the actual measured value of C was 0.39. Moreover, the degrees of adjacent nodes were positively correlated for this undirected graph. Therefore, we consider that the Wikipedia dataset can be used as the network data to evaluate the performance of the proposed method for solving the influence maximization problem on a social network.

Experimental Setting

For solving the influence maximization problem under the greedy algorithm, we compared the proposed method with a conventional method.

Given a subset A of V, $\nabla\sigma(A)$ is conventionally computed by independently estimating $\sigma(A\cup\{v\})$ for all $v \in V$. Moreover, each $\sigma(A \cup \{v\})$ is estimated in the following way: We run the random process of each model from the initial active set $A \cup \{v\}$, and count the number of final active nodes. The empirical mean obtained by many such simulations is used as the estimate of $\sigma(A \cup \{v\})$. From the problem of computational time, we mainly used 100 simulations and 1000 simulations to estimate $\nabla\sigma(A)$ in the experiments. We refer the methods using 100 simulations and 1000 simulations for the IC model to as the *IC100* and the *IC1000*, respectively. In the same way, we define the *LT100*, *LT1000* and *LT10000* for the LT model.

For the proposed method based on bond percolation, we need to specify the number M of sample vectors in Equation (1). We refer the methods using $M = 100$, $M = 1000$ and $M = 10000$ for the IC model to as the *ICBP100*, *ICBP1000* and *ICBP10000*, respectively. We define the *LTBP100*, *LTBP1000* and *LTBP10000* for the LT model in the same way.

On the other hand, the IC and LT models have parameters to be specified in advance. In the IC model, we assigned a uniform probability p to the propagation probability $p_{u,v}$ for any directed link (u, v) of the network, that is, $p_{u,v} = p$. In the LT model, we uniformly set weights as follows: For any node v of the network, the weight $w_{u,v}$ from a parent node $u \in \Gamma(v)$ is given by $w_{u,v} = 1/|\Gamma(v)|$.

Experimental Results

We compared the proposed method with the conventional method in terms of both the performance of the approximate solution A_k obtained for a target set size k and the processing time. The performance of A_k is measured by the influece degree $\sigma(A_k)$. We estimated $\sigma(A_k)$ by using $300,000$ simulations accoriding to the work of Kempe, Kleinberg, and Tardos (2003). All our experimentation was undertaken on a single Dell PC with an Intel 3.4Ghz Xeon processor, with 2GB of memory, running Linux.

Tables 1 and 2 show the performance of the approximate solution A_k of size k by each method for the IC model with $p = 10\%$ and the LT model on the blog dataset, respectively, where the values are rounded to the first decimal place. As predicted, we observe that the solutions by the IC1000, ICBP1000, LT1000 and LTBP1000 outperfom those by the IC100, ICBP100, LT100 and LTBP100, respectively. Moreover, we observe that the solutions by the ICBP1000 and LTBP1000 outperfom those by the IC1000 and LT1000, respectively. Here, we investigate the reason for the results.

Table 1: Performance of approximate solutions for the influence maximization problem under the IC model with $p = 10\%$ in the blog dataset.

k	IC100	IC1000	ICBP100	ICBP1000
1	173.9	173.9	173.9	173.9
10	661.0	693.4	693.1	701.8
20	743.1	858.1	869.0	874.3
30	831.7	959.1	983.8	990.7

Table 2: Performance of approximate solutions for the influence maximization problem under the LT model in the blog dataset.

k	LT100	LT1000	LTBP100	LTBP1000
1	275.6	285.6	285.6	285.6
10	1543.8	1592.4	1590.5	1603.5
20	2126.2	2412.0	2428.0	2436.5
30	2649.9	3023.5	3049.6	3065.3

Let us consider estimating $\nabla\sigma(A_k)$, and choosing a node v_{k+1} that maximizes $\sigma(A_k \cup \{v\})$ $(v \in V)$. Then, we note that the effect of A_k is not equally evaluated for all $v \in V$ in the conventional method, since $\sigma(A_k \cup \{v\})$ is independently estimated for every v by simulations. We also note that the number of final active nodes for a given target set greatly varied every simulation in the IC and LT models. These facts imply that for the conventional method without performing many simulations, the selection of v_{k+1} can completely depend on how the effect of A_k is evaluated by chance for each v. On the other hand, the effect of A_k is equally evaluated for all $v \in V$ in the proposed method. In fact, when $\sigma(A_k \cup \{v\})$ is estimated using Equation (1), each $|F(A_k \cup \{v\}; G_{r_m})|$ is basically computed by

$$|F(A_k \cup \{v\}; G_{r_m})| = \left|F(v; G_{r_m}^{A_k})\right| + |F(A_k; G_{r_m})|\,.$$

Therefore, we consider that the proposed method can outperform the conventional method.

Table 3 shows the processing time to obtain A_k for the IC1000, ICBP1000, LT1000 and LTBP1000 on the blog dataset, where the values are rounded to three significant figures. As predicted, the IC1000, ICBP1000, LT1000 and LTBP1000 needed about ten times as much processing time as the IC100, ICBP100, LT100 and LTBP100, respectively. We observe from Table 3 that the ICBP1000 and LTBP1000 can be much more efficient than the IC1000 and LT1000, respectively. For example, to obtain the approximate solution A_{30} for $k = 30$, both the IC1000 and LT1000 needed about 2.5 days, while the ICBP1000 and LTBP1000 needed about 2.5 and 1.5 minutes, respectively. We also examined the LT10000 on the blog dataset. Although the ICBP10000 and LTBP10000 outperformed the ICBP1000 and LTBP1000 just a little, resepectively, the LT10000 still improved the LT1000 in performance of approximate solutions. For example, for $k = 30$, the performance values of the solutions by the LT10000, LTBP10000 and ICBP10000 were 3059.0, 3066.3 and 991.6, respectively. Moreover, to obtain approximate solution A_{30}, the LT10000 needed about 27 days, while the LTBP10000 needed only about 14 minutes. These results indicate that the proposed method can be much more efficient than the conventional method, and achieve a large reduction in computational cost.

Table 3: Processing time (sec.) in the blog dataset.

k	IC1000	ICBP1000	LT1000	LTBP1000
1	3.70×10^2	7.07	6.57×10^2	3.19
10	4.69×10^4	5.68×10^1	4.24×10^4	2.96×10^1
20	1.24×10^5	1.09×10^2	1.25×10^5	5.64×10^1
30	2.13×10^5	1.60×10^2	2.32×10^5	8.20×10^1

Tables 4, 5 and 6 show the experimental results in the Wikipedia dataset. We can see that the results were qualitatively very similar to the ones for the blog dataset. We also conducted experiments on some real large networks including a blogroll network of blogs, and confirmed the effectiveness of the proposed method.

Table 4: Performance of approximate solutions for the influence maximization problem under the IC model with $p = 1\%$ in the Wikipedia dataset.

k	IC100	IC1000	ICBP100	ICBP1000
1	122.0	138.6	137.1	138.6
10	371.1	390.6	396.6	405.3
20	410.8	455.7	469.3	475.1
30	449.5	497.0	509.8	516.0

Table 5: Performance of approximate solutions for the influence maximization problem under the LT model in the Wikipedia dataset.

k	LT100	LT1000	LTBP100	LTBP1000
1	340.8	340.8	293.4	340.8
10	1237.2	1715.5	1669.3	1718.0
20	1991.8	2554.8	2496.3	2581.6
30	2214.4	3117.2	3054.8	3181.0

Table 6: Processing time (sec.) in the Wikipedia dataset.

k	IC1000	ICBP1000	LT1000	LTBP1000
1	6.63×10^2	1.91×10^1	5.41×10^2	5.17
10	1.94×10^5	1.74×10^2	9.60×10^4	4.64×10^1
20	4.82×10^5	3.42×10^2	3.03×10^5	8.57×10^1
30	8.03×10^5	5.10×10^2	5.69×10^5	1.21×10^2

Related Work

First, we describe some work related to the computation of influence degrees in the IC model. Let us recall that the SIR model for the spread of a disease on a network is equivalent to a bond percolation model on the same network, and the size of a disease outbreak from a node corresponds to the size of the cluster that can be reached from the node by traversing only the "occupied" links. Using this correspondence, researchers presented a method of theoretically calculating the probability distribution for the size of a disease outbreak that starts with a randomly chosen node in the configuration model (i.e., a random network model) with a given degree distribution (Newman 2002; 2003). Moreover, they theoretically derived a condition for the disease outbreak from a randomly chosen node to give an *epidemic outbreak* that affects a non-zero fraction on the network in the limit of large network size. Mathematically more rigorous treatments of similar results can be found in the work of Molloy and Reed (1998).

Next, we describe some work related to the computation of influence degrees in the LT model. Watts (2002) investigated the LT model on a network to explain large but rare cascade phenomena triggerd by small initial shocks. Using the concept of *site percolation*, he theoretically derived a condition for the cascade from a randomly chosen seed node to give a *global cascade* that affects a non-zero fraction on the network in the limit of large network size for the configuration model (i.e., a random network model) with a given degree distribution.

The above mentioned studies focused on global properties averaged over a random network in the limit of large network size, while our primary concern is to practically answer which nodes are most influential for information diffusion on a given real-world network of finite size. We also note that those studies dealt with undirected graphs, while our work investigates information diffusion on networks represented by directed graphs. Moreover, the theories developed in those studies assumed that the loop structure on a network of interest can be essentially ignored in the limit of large network size. However, this property is not true of many large-scale social networks, and it is an open question whether or not those theories are effective for such networks (Newman 2003). In fact, although the clustering coefficient C quantifies the loop structure in a network, it was observed that many social networks have much higher values of C than the corresponding configuration models (i.e., random network models) (Newman & Park 2003).

Conclusion

We have considered the influence maximization problem on a large-scale social network represented as a direced graph for the IC and LT models. For approximately solving the problem, the conventional method under the greedy algorithm needed a large amount of computation. Thus, we have proposed a method of efficiently estimating all the marginal gains $\nabla\sigma(A)$ for the influence degree $\sigma(A)$ of a given target set A, and applied it to approximately solving the problem under the greedy algorithm. Using real-world large-scale networks including blog networks, we have experimentally demonstrated that the proposed method can be much more effective than the conventional method.

Acknowledgements

This work was partly supported by JSPS Grant-in-Aid for Scientific Research (C) (No. 18500113).

References

Domingos, P., and Richardson, M. 2001. Mining the network value of customers. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 57–66.

Goldenberg, J.; Libai, B.; and Muller, E. 2001. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing Letters* 12:211–223.

Gruhl, D.; Guha, R.; Liben-Nowell, D.; and Tomkins, A. 2004. Information diffusion through blogspace. In *Proceedings of the 7th International World Wide Web Conference*, 107–117.

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

Leskovec, J.; Adamic, L. A.; and Huberman, B. A. 2006. The dynamics of viral marketing. In *Proceedings of the 7th ACM Conference on Electronic Commerce*, 228–237.

McCallum, A.; Corrada-Emmanuel, A.; and Wang, X. 2005. Topic and role discovery in social networks. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, 786–791.

Molloy, M., and Reed, B. 1998. The size of the giant component of a random graph with a given degree sequence. *Combinatorics, Probability and Computing* 7:295–305.

Newman, M. E. J., and Park, J. 2003. Why social networks are different from other types of networks. *Physical Review E* 68:036122.

Newman, M. E. J. 2001. The structure of scientific collaboration networks. *Proceedings of National Academy of Science, USA* 98:404–409.

Newman, M. E. J. 2002. Spread of epidemic disease on networks. *Physical Review E* 66:016128.

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

Richardson, M., and Domingos, P. 2002. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 61–70.

Watts, D. J. 2002. A simple model of global cascades on random networks. *Proceedings of National Academy of Science, USA* 99:5766–5771.

Xie, A., and Beerel, P. A. 2000. Implicit enumeration of strongly connected components and an application to formal verification. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 19:1225–1230.