2013 IEEE 33rd International Conference on Distributed Computing Systems

Least Cost Rumor Blocking in Social Networks

Lidan Fan*, Zaixin Lu*, Weili Wu*, Bhavani Thuraisingham*, Huan Ma†, and Yuanjun Bi*
*Department of Computer Science
The University of Texas at Dallas
Dallas, USA
Email: {lidan.fan, zaixin.lu, weiliwu, bhavani.thuraisingham, yuanjun.bi}@utdallas.edu
†School of Information
Renmin University of China
Beijing,China
Email: mahuan@ruc.edu.cn

***Abstract*—In many real-world scenarios, social network serves as a platform for information diffusion, alongside with positive information (truth) dissemination, negative information (rumor) also spread among the public. To make the social network as a reliable medium, it is necessary to have strategies to control rumor diffusion. In this article, we address the *Least Cost Rumor Blocking* (LCRB) problem where rumors originate from a community C_r in the network and a notion of *protectors* are used to limit the bad influence of rumors. The problem can be summarized as identifying a minimal subset of individuals as initial protectors to minimize the number of people infected in neighbor communities of C_r at the end of both diffusion processes. Observing the community structure property, we pay attention to a kind of vertex set, called *bridge end set*, in which each node has at least one direct in-neighbor in C_r and is reachable from rumors. Under the OOAO model, we study LCRB-P problem, in which α $(0 < \alpha < 1)$ fraction of bridge ends are required to be protected. We prove that the objective function of this problem is submodular and a greedy algorithm is adopted to derive a $(1-1/e)-$approximation. Furthermore, we study LCRB-D problem over the DOAA model, in which all the bridge ends are required to be protected, we prove that there is no polynomial time $o(\ln n)-$approximation for the LCRB-D problem unless $P = NP$, and propose a Set Cover Based Greedy (SCBG) algorithm which achieves a $O(\ln n)-$approximation ratio. Finally, to evaluate the efficiency and effectiveness of our algorithm, we conduct extensive comparison simulations in three real-world datasets, and the results show that our algorithm outperforms other heuristics.**

***Keywords*-least cost rumor blocking; opportunistic One-Activate-One model; deterministic One-Activate-Many model; approximation algorithm; social networks;**

I. Introduction

With the increasing popularity of online social networks, such as twitter, facebook, renren and so forth, rumors can spread farther, quicker, and even with more terrible effect. In real-world situations, rumors exist in almost every domain of society. For example, a rumor [1] generated in Twitter said that the president of Syria is dead, which hit the twitter greatly and was circulated fast among the population, finally leading to a sharp, quick increase in the price of oil. Another event occurred in August, 2012, thousands of people in Ghazni province left their houses in the middle of the night in panic after the rumor of an earthquake, which said that a major earthquake would hit the area until 5:00 am [2]. Believing in it, many people from Ghazni city and other districts of the province left their houses and spent the entire night outside.

Against this backdrop, in this work, we consider a strategy that initiates protectors to fight against rumors on social networks. In this study, we assume that rumors and protectors follow the same diffusion mechanism, and each node can be in one of the three statuses: *protected*, *infected*, or *inactive*. The key difference between the two cascades, namely *cascade P* for protector and *cascade R* for rumor, is that when cascade P and cascade R arrive a node at the same time, cascade P has priority to activate the node. Some schemes on limiting rumor diffusion have been proposed in earlier works. Kimura *et al.* in [8] suggested to block a certain number of links in a network to reduce the terrible results caused by rumors. In [13], Chen *et al.* presented an efficient algorithm to maximize the positive influence when negative opinion appears. The works of [14], [16], [26] considered to block the influence of rumors as much as possible through initiating a protective campaign.

Based on the knowledge that a social network is composed of a set of disjoint communities [25], we assume that rumors originate from one community, which can be validated by instance [2]. Furthermore, the community structure property in a social network shows that individual relations within a community is denser than that across communities. Accordingly, influence spread faster within a community, while slower across different communities. This thereby provides us the observation that it is practical for us to prevent R cascade from spreading out its own community. One strategy is to protect the individuals that have direct neighbor(s) in the community that rumor originators are generated and can be reached by rumor originators. For simplification, the community that contains rumor originators is called *rumor community* and the neighbor communities of rumor community are called *R-neighbor communities* .

1063-6927/13 $26.00 © 2013 IEEE
DOI 10.1109/ICDCS.2013.34

In this work, we study the problem of minimizing the number of protectors used to protect bridge ends, and call it *Least Cost Rumor Blocking* (LCRB) problem. Those bridge ends are the boundary individuals in R-neighbor communities and can be infected by the cascade coming from rumor originators. We search them in the first stage of our algorithm by *Breadth First Search* (BFS) method through constructing *Rumor Forward Search Tree(s)* (RFST).

We investigate the LCRB problem under two influence diffusion models: the *Opportunistic One-Activate-One* (OPOAO) model and the *Deterministic One-Activate-Many* (DOAM) model. Under the OPOAO model, we study the LCRB-P (the LCRB problem under the OPOAO model) problem, in which α $(0 < \alpha < 1)$ fraction of bridge ends are required to be protected, and prove that the objective function of this problem is submodular, and thus the greedy algorithm yields a spread that is within $1 - 1/e$ of optimal [5]. In the DOAM model, we consider the LCRB-D (the LCRB problem under the DOAM model) problem, in which all the bridge ends are required to be protected. We propose the *Set Cover Based Greedy* (SCBG) algorithm through converting the LCRB-D problem to set cover problem. Then we demonstrate that there is no polynomial time $o(\ln n)-$approximation for the LCRB-D problem unless $P = NP$, and get a $O(\ln n)-$approximation ratio solution. Finally, we conduct experiments on real-world networks and prove that the Greedy and the SCBG algorithm outperform other heuristics both from efficiency and accuracy, respectively.

Roadmap: The remainder of this paper is organized as follows: In section II, we survey related works and in section III, we specify two influence diffusion models, namely, the OPOAO model and the DOAM model. In section IV, we formulate the problems under the two models and introduce several relative definitions. In section V, for the OPOAO model, we show the submodularity of the objective function for LCRB-P problem and present the greedy algorithm. For the DOAM model, we prove that there is no polynomial time $o(\ln n)-$approximation for the LCRB-D problem unless $P = NP$, and provide a detail description of the SCBG algorithm. In section VI, we analyze experimental implementation results and finally, we conclude this paper in section VII.

II. Related Work

The influence diffusion problem was first captured by Richardson and Domingos in [3], [4], where they explored the *Influence Maximization* (IM) problem as an algorithmic problem. Later, Kempe *et al.* in [5], [6] further formulated the IM problem into an optimization problem and studied it on two classical models: the *Independent Cascade (IC)* model and the *Linear Threshold* (LT) model, which is the milestone for subsequent attempts on IM problem. To learn more, the readers are referred to [10], [11], [7], [12], [21], [24], [31], [33]. However, these definitions of influential individuals ignore certain aspects of real social networks such as the existence of multiple cascades, which have applications in political election, product promotion, rumor dissemination and so forth. Hence, recently, some researchers made efforts on multiple-cascade diffusion problem and its variants in [15], [17], [19], [20], [23], [13].

Bharathi *et al.* [15] studied competitive influence diffusion under the extension of the IC model. They proposed a $(1 - 1/e)-$approximation algorithm for computing the best response to an opponent's strategy, and gave a FPTAS for the problem of maximizing the spread of the influence of a single player under a tree. Kostka *et al.* [20] regarded the rumor diffusion as a game theoretical problem under a much more restricted model than the IC and LT models, and pointed out that the first player does not always have benefit for the earlier start. In [23], Trpevski *et al.* proposed a competitive rumor spreading model based on the SIS (susceptible-infected-susceptible) model, but they did not address the influence maximization problem or the rumor blocking problem. *Borodin et al.* [19] studied competitive influence diffusion in several different models generated from the LT model. Chen *et al.* [13] investigated the positive influence maximization problem under an extension of the IC model which incorporates the negative opinions of the customers to the product quality.

The influence blocking maximization (IBM) problem was introduced in [14], [16]. Unlike previous works that aim at maximizing the spread of the influence for each cascade, [14], [16] focused on the problem that one entity tries to block the influence diffusion of its competing entity as much as possible by selecting a given number of initial seed set. In [14], Budak *et al.* studied the Eventual Influence Limitation (EIL) problem under the extension of the IC model. Unfortunately, they only proposed the greedy algorithm and several simple heuristics. In [16], He *et al.* proposed the competitive linear threshold (CLT) model to address the IBM problem. They proved that the objective function for this problem is submodular and obtained a $(1 - 1/e)-$approximation ratio. To reduce time consumption, they adopted the CLDAG algorithm, which is similar to the LDAG algorithm in [11].

In [26], Nguyen *et al.* exploited the β_T^I-Node Protector diffusion problems, which aim at minimizing the set of highly influential nodes used to limit the spread of misinformation originated from I to a desired rate $(1-\beta)$ $(\beta \in [0, 1])$ in T steps. The authors proposed a Greedy Viral Stopper (GVS) algorithm that greedily adds nodes with the best influence gain for $\beta-$Node Protectors to the current solution. They also applied GVS to the network restricted to $T-$hop neighbors of the initial set I and obtained a slightly better bound for β_T^I-Node Protector problems. Finally, a community-based algorithm was employed to reach a good selection of protectors to decontaminate the rumors in a timely manner.

III. Two Influence Diffusion Models

In this section, we introduce the OPOAO and DOAM models, both of which model the diffusion of two cascades evolving simultaneously in a network. Let R (for "rumor") and P (for "protector") denote the two cascades respectively. For the sake of consistency, we will speak of a node as being *infected (protected)* (influenced by rumors (protectors) either initially or sequentially from one of its neighbors), or *inactive* otherwise. The initial set of infected (protected) nodes for cascade R (P), namely rumor originators (protector originators), is denoted as S_R (S_P). We assume that individuals in both of the two models have high trust level in their neighbors as in [2]. It means that individuals will believe in what their neighbors tell them.

In the two models, a social network is considered as a directed graph $G = (N, E)$, consisting of nodes N and edges E. N can be viewed as the individuals of the social network, and E can be regarded as social relationships among individuals.

Right now, we introduce three common properties of the two models: 1) Cascade R and P start at the same time; 2) when R and P reach a node u at the same time, P has the priority to activate u; 3) R or P diffuses *progressively*, that is, nodes can switch from being inactive to being infected or protected, but do not switch in the other direction, and once an inactive individual becomes infected or protected, it will keep the status forever. The first assumption makes sense since it is essentially the same as the case where R starts earlier than P, under which we incorporate the newly infected nodes before P into the initial rumors, and view the new set as the initial rumors. For assumption 2), it is reasonable since people are likely to believe the truth. The third assumption comes from [5].

A. Opportunistic One-Activate-One (OPOAO) Model

At step 0, there are two disjoint initial (seed) sets, the set of rumor originators S_R and the set of protector originators S_P. When a node u first becomes infected or protected at step t, it has a single chance to choose one of its neighbors as an activation target, which will be activated successfully at $t + 1$. Each neighbor of u can be selected as the activation target of u with the same probability $1/d_{out}(u)$, where $d_{out}(u)$ is the out-degree of u. At $t + 1$, u and the newly activated node choose their own activation targets, respectively. This procedure unfolds in discrete time and continues until no inactive nodes can be protected or infected.

The influence diffusion procedure is similar to person-to-person contact mechanism in mobile social network, in which each person can only communicate with one person at the same time. Obviously, each node has no memory about his/her interactions with others, and the speed of influence spread is slow under this model for the existence of repeat selection.

B. Deterministic One-Activate-Many (DOAM) Model

At step 0, there are two disjoint initial (seed) sets, the set of rumor originators S_R and the set of protector originators S_P. When a node u first becomes infected or protected at step t, all of its currently inactive neighbors will be activated successfully at step $t + 1$, and each protector or rumor only has single chance to influence its neighbors. The cascade spreads in discrete time and does not stop until no inactive nodes can be protected or infected.

This influence propagation mechanism is similar to information broadcast procedure. Obviously, the influence spread fairly fast compared with that in the OPOAO model since at each step, the number of newly activated nodes increases significantly.

IV. Problem Formulation

With the discussed OPOAO and DOAM models, in this section, we introduce our problems and several definitions concerning with them.

It is known that social networks possess a common phenomenon: the property of containing community structure, that is, they divide into groups of vertices with dense connections within each group while sparse connections crossing groups, where the vertices and connections stand for network users and their social relations, respectively. In general, people join a same community with sharing common interests or other attributes, which means they tend to interact more frequently with other members in the same community than with people outside. In other words, edges crossing between communities are of usually few, thus a node from a community often has little chance to spread out rumor to a node in a different community.

Taking into account of this advantage of community structure, to efficiently decontaminate the wide spread of rumors in a network by least number of protectors, we pay attention to the members in the R-neighbor communities, i.e., confine the rumor diffusion to its own community. To realize it, intuitively, it is reasonable (connections across different communities are sparse) to protect the bridge ends, which have relations with the members in rumor community, and can be reached earlier than other members in their own communities.

Definition 1: A social network is a directed graph $G(V, E, C)$, where each node $v_i \in V$ represents an individual in the network, and a directed edge $(v_i, v_j) \in E$ denotes the event that individual v_i has impact on individual v_j, here $C = \{C_1, C_2, \cdots, C_k\}$ is a set of disjoint communities that form the network satisfying $\bigcup_{r=1}^{k} V(C_r) = V$, where $V(C_r)$ represents the individuals in community C_r.

Definition 2: Least Cost Rumor Blocking (LCRB) problem: Given a community C_k in $G(V, E, C)$, a set of rumor originators $S_R \subseteq V(C_k)$ ($C_k \in C$ is predetermined) and bridge ends B, our goal is to find a least number of nodes as protector originators such that they can protect at least

α ($0 \leq \alpha \leq 1$) fraction of the bridge ends at the end of influence diffusion.

Considering the overhead used for controlling the spread of rumors, we introduce two variants of the LCRB problem: the LCRB-P for the OPOAO model and the LCRB-D for the DOAM model, respectively.

Definition 3: Least Cost Rumor Blocking problem under Deterministic (Opportunistic) model: The LCRB problem is called LCRB-D problem when $\alpha = 1$, indicating that the selected protectors must guarantee that all bridge ends are protected. The LCRB-P problem is to protect at least α ($0 < \alpha < 1$) fraction of the bridge ends.

Obviously, the requirement of protection level for the LCRB-D problem is higher than that for the LCRB-P problem. It is because that under the DOAM model, rumors propagate rapidly among individuals in the same community. That is, under the DOAM model, once a rumor appears in a community, a large amount of individuals in the community will be infected within a short time.

Definition 4: Set Cover (SC) Problem: Given a set of elements $U = \{v_1, v_2, \cdots, v_n\}$ and a set of m subsets of U, called $S = \{S_1, S_2, \cdots, S_m\}$, find a "least cost" (minimum size) collection $\mathcal{C}$ of sets from S such that $\mathcal{C}$ covers all the elements in U. That is, $\bigcup_{S_i \in \mathcal{C}} S_i = U$.

V. Methodology

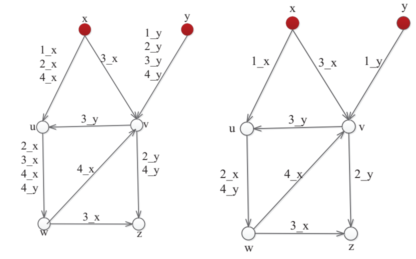
In this section, we propose the greedy algorithm for the LCRB-P problem and the SCBG algorithm for the LCRB-D problem. The LCRB-D and LCRB-P problems include two stages, one is to determine the bridge ends and the other is to select the protectors. We use the BFS method to find out the bridge ends.

A. For the OPOAO Model

For the LCRB-P problem under the OPOAO model, we prove that the expected influence function of protectors is submodular and the greedy algorithm guarantees a $(1-1/e)$ approximation ratio.

1) Submodularity of the Expected Protector Influence Function under the OPOAO Model:

Given a set of bridge ends, $PB(A)$ is defined as a *protector blocking set* on the bridge ends, in which the individuals will be infected if the set of protector originators is empty, but is not infected if the set of protector originators is A. Denote by $\sigma(A)$ the expected size of $PB(A)$. Due to the NP-hardness of the LCRB-P problem, we need to find approximation algorithms for it. The submodularity of $\sigma(A)$ provides us a good way to obtain an approximation algorithm for our problem. To prove the submodularity property, we need to verify that $\sigma(A)$ satisfies the diminishing return condition. That is, for any two sets X and Y with $X \subseteq Y$, the marginal gain of adding a node v into X is not less than that of adding the same node into Y: $\sigma(X \bigcup \{v\}) - \sigma(X) \geq \sigma(Y \bigcup \{v\}) - \sigma(Y)$.



(a) (b)

Figure 1. (a) An illustration for timestamp assignment on edges; (b) The simplification of (a).

Since it is hard to compute the exact value of $\sigma(A)$, we use $PB(A)$ to estimate its value and prove the submodularity through modifying the approach used by He *et al.* in [16]. Nevertheless, our proof is more intricate since the behaviors of repeat activation during influence diffusion in the OPOAO model. To deal with this issue, we propose a timestamp assignment method, which follows the rule: at each step, when an active node determines its activation target, a timestamp will be assigned to the corresponding edge between the active node and its target. Obviously, an edge may have many timestamps since it can be chosen as an activation target for many times. Due to the fact that once a node is infected or protected, it will never change its status. Therefore, for simplification, on each edge, we only preserve the smallest timestamp corresponding to each rumor seed, which reduce the number of timestamps on each edge greatly. To clarify the timestamp assignment procedure, we take the diffusion process of rumors for an example.

For example, in Fig. $1(a)$, x and y are the rumor originators. At step 1, node x chooses u as its activation target and node y selects v as its activation target, then timestamps 1_x and 1_y are assigned on edges (x, u) and (y, v), respectively. At step 2, u chooses w and v chooses z, and meanwhile, x chooses u and y chooses v again, then timestamps 2_x and 2_y are assigned on (x, u) and (y, v) accordingly. For the simplification version Fig. $1(b)$ of Fig. $1(a)$, only two timestamps $2\_x, 4\_y$ are preserved on edge (u, w), which means that the cascade from x reaches node w at the 2nd step while that from y arrives w at the 4th step.

Given an initial graph $G = (V, E, C)$, the set of rumor originators S_R and the set of protector originators S_P, we construct two random graphs G_R and G_P for cascade R and P respectively. At the first step, for each $u \in S_R$, we randomly pick up one out-edge (u, v) with probability $1/d_u$, where d_u is the out-degree of u, and then assign timestamp 1_u on this edge. Let X denote the newly infected vertices at the end of the step. Then, for each node belonging to $S_R \cup X$, repeat the random selection process. As for the timestamp in cascade P, on edge (u, v), denote by t'_w the event that the influence of protector w arrives node v at step

t', where t' is the timestamp value and w is a protector. The expected *influence function of protectors* is defined as $\sigma(A) = E_{(G_R,G_P)}(|PB(A)|)$, which is the expected number of bridge ends that can be protected by vertex set A.

Using the following lemmas we prove the submodularity of $PB(A)$ through timestamp assignment method.

Lemma 1: In G_R, for any node v in the set B of bridge ends, there exists at least a path of cascade R from a rumor scource u to v if there is a timestamp k_u on one in-edge of v, and for any u in G_P, there exists at least a path of cascade P from a protector seed w to u if there is a timestamp k'_w on one in-edge of u.

Then we use the next two lemmas to show the necessary conditions for $v \in PB(S)$ and $v \in PB(T \cup \{u\}) \setminus PB(T)$. Let $G_P(S)$ be the graph with vertices that can be reached by influence from S and edges that have been used to choose activation targets.

Lemma 2: The sufficient conditions for node $v \in PB(S)$ are:

(1) Node v belongs to G_R and $G_P(S)$;

(2) There exists at least one timestamp k'_u on one in-edge of v in $G_P(S)$ and is no larger than the smallest timestamp on in-edge(s) of v in G_R.

Lemma 3: The sufficient conditions for node $v \in PB(T \cup \{u\}) \setminus PB(T)$ are:

(1) Node v belongs to G_R and $G_P(T \cup \{u\})$;

(2) There exists at least one timestamp k'_u on one in-edge of v in $G_P(T \cup \{u\})$ and is no larger than the smallest timestamps on in-edge(s) of v in G_R,

(3) For all $x \in T$, the smallest timestamp among all $t'\_x's$ on in-edges of v in $G_P(T)$ is larger than that on in-edges of v in G_R.

Lemma 4: The cardinality set function $|PB(S)|$ for a pair of random graphs G_R and G_P is monotone and submodular.

Theorem 1: For the OPOAO model, $\sigma(A)$ is monotone and submodular.

2) Greedy Algorithm: We introduce the greedy algorithm, which always selects the node that contributes the largest marginal gain in protecting the bridge ends.

Algorithm 1 Greedy Algorithm for the OPOAO model

```
INPUT: A directed graph $G = (V, E, C)$, a community $C_m$ and a rumor originators $S_R = \{r_1, r_2, \cdots, r_M\} \subseteq V(C_m)$, a protection level $\alpha$;
OUTPUT: Protector seed set $S_P \subseteq V$;
For each $r \in S_R$, construct the Rumor Forward Search Tree (RFST) by the BFS method to find all bridge ends in $G$, and denote them by a set $B$;
Initialize $S_P = \phi$;
while $\sigma(S_P) < \alpha|B|$ do
    Select $u = argmax_{v \in V \setminus S_P \cup S_R} \sigma(S_P \cup \{v\}) - \sigma(S_P)$;
    $S_P = S_P \cup \{u\}$;
end while
return $S_P$.
```

B. For the DOAM Model

In the following, we provide a $O(\ln n)-$approximation ratio solution based on the fact that the LCRB-D problem is equivalent to the Set Cover (SC) problem.

1) Performance for the LCRB-D Problem:

Theorem 2: [18] There is a polynomial time $O(\ln n)-$approximation algorithm for the LCRB-D problem, where n is the number of vertices in the set B of bridge ends.

Proof: Assume that we have an input of LCRB-D instance $\mathcal{A}$. For each vertex v_i of B, use BFS method to find all vertices that can reach v_i before v_i is infected, and this can be done in polynomial time. Assume we have a candidate root set S, for each vertex r_j in S, use BFS method to find all vertices of B that are reachable from r_j before they are infected. Obviously, each root can protect a subset of vertices of B, then the problem becomes a SC problem, i.e. use the least number of roots to cover all vertices of B. Therefore, it has a polynomial time $O(\ln n)-$factor approximation, where n is the number of nodes in B. ■

Theorem 3: If the LCRB-D problem has an approximation algorithm with ratio $k(n)$ if and only if the SC problem has an approximation algorithm with ratio $k(n)$.

Proof: Assume S_1, $\cdots$, S_m is the list of sets for the SC problem and $S_1 \bigcup S_2 \bigcup \cdots \bigcup S_m = \{a_1, \cdots, a_n\}$, we construct a social network as follows.

(1) For each set S_i, create a vertex u_i. For each a_j, create a vertex v_j, add directed edges from u_i to v_j if $a_j \in S_i$. An edge from u_i to v_j means v_j can be protected by u_i.

(2) Create a social network with a constant number of individuals and an infected node r, add directed edges from r to $v_1, v_2, \cdots, v_n$.

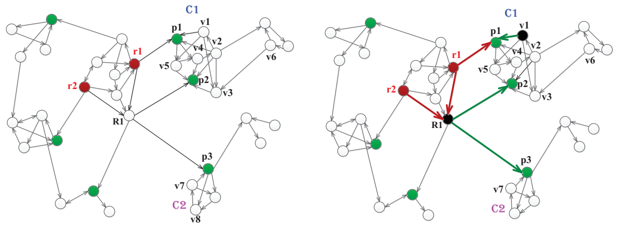
(3) Let B be the set of bridge ends including vertices $v_1, v_2, \cdots, v_n$ that need to be protected.

(4) The set cover problem is converted into the LCRB-D problem. Thus, it is reasonable to point out that the LCRB-D problem has a $k(n)$-approximation if and only if the SC problem has a $k(n)$-approximation. ■

Corollary 1: There is no polynomial time $o(\ln n)-$approximation for the LCRB-D problem unless $P = NP$.

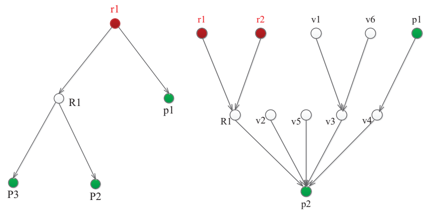
Proof: It follows from Theorem 3 and the well-known inapproximability result for the SC problem [18]. ■

2) The SCBG Algorithm: Now we introduce the SCBG algorithm described in **Algorithm** 3. The main idea is as follows: given the set S_R of rumor originators and the set B of bridge ends, for any $v \in B$, use BFS method to construct v's Bridge End Backward Search Tree (BBST) Q_v with v as the root. (The search depth is determined by the shortest length of the path between v and the rumor(s) found first, each node in this tree except the rumor(s) can protect node v). Denote by $Q_1, Q_2, \cdots, Q_{|B|}$ the corresponding BBSTs



(a) (b)

Figure 2. (a) Communities containing rumors and bridge ends, red nodes are rumors and green vertices are bridge ends; (b) A set of protector originators {v1, R1} for bridge ends in communities C1 and C2.



(a) (b)

Figure 3. (a) Forward search tree for rumor $r1$ with respect to Fig. 2(a), and bridge ends are $p1, p2, p3$; (b) Backward search tree for bridge end $p2$ with respect to Fig. 2(a), all nodes in this tree except $r1, r2$ can protect $p2$.

for all bridge ends. Then for each $u \in Q_i$, $1 \leq i \leq |B|$, check all other BBSTs i.e. Q_j $(j \neq i, and\ 1 \leq j \leq |B|)$ to find the ones that contain u, and record them, then connect their roots to node u. Thus a 1-hop tree is formed and u is labeled as the root of the newly created 1-hop tree. Let SW_u denote all leaves in this tree. Then we apply **Algorithm** 2 to select the least number of sets from $SW_u's$ to cover all the nodes in B. Finally, the $u's$ form the set S_P of protector originators.

Algorithm 2 Greedy Algorithm in SCBG Algorithm

```
INPUT $B$, $Q_i$ and $SW_j$, where $i = 1, \cdots, |B|$, $j = 1, \cdots, |\bigcup_{k=1}^{|B|} Q_k \setminus S_R|$;
OUTPUT $W$;
Initialize $L = \phi$ and $W = \phi$;
while $|L| < |B|$ do
    Select $u = arg \quad max_{v \in \bigcup_{k=1}^{|B|} Q_k \setminus S_R} |SW_v \setminus L|$;
    $W = W \cup \{u\}$ and $L = L \cup SW_u$;
end while
return $W$.
```

To clarify the formation process of the RFST and BBST, we give the following examples. In Fig. $2(a)$, start from the rumor originators $\{r_1, r_2\}$, go along with paths that reach some nodes in R-neighbor communities, we obtain the set of bridge ends by BFS method, and mark them green. In (b), for simplification, we only illustrate an optimal protector originators for communities C_1 and C_2 respectively, which is the black vertex set $\{R_1, v_1\}$. As seen from Fig. $2(b)$, the green edges form the paths generated by cascade P while the red ones form the paths generated by cascade R.

Algorithm 3 SCBG Algorithm-Select Protector Seed Set

```
INPUT: A directed graph $G = (V, E, C)$, a given community $C_m$ and rumor originators $S_R = \{r_1, r_2, \cdots, r_M\} \subseteq V(C_m)$;
OUTPUT: Protector seed set $W \subseteq V$;
For each $r \in S_R$, construct Rumor Forward Search Tree (RFST) by BFS method to find all bridge ends in $G$, which are the leaves of the RFST, and denote them by a set $B$;
For each node $v \in B$, construct Bridge end Backward Search Tree (BBST) by BFS method to find and record all the in-neighbors $w \in N^i(v)$ of $v$, where $i$ is determined by the value of the shortest paths between $v$ and any node $w \in S_R$. Assume $N^0(v) = v$. Denote the vertex set of this tree as $Q_v$;
List all $Q_v$'s as $Q_1, \cdots, Q_{|B|}$. For $u \in Q_i \setminus S_R \bigcup_{k=1}^{i-1} Q_k$, add a directed edge from $u$ to $i$. Search $Q_{i+1}, \cdots, Q_{|B|}$ to determine whether $u$ belongs to them, if $u \in Q_w$, add a direct edge from $u$ to $w$, then a 1-hop tree is constructed with root $u$, and leaves are $w$'s, denote by $SW_u$ the leaves of those 1-hop trees.
Apply Algorithm 2 on $SW_u$'s to cover $B$;
Return OUTPUT of Algorithm 2.
```

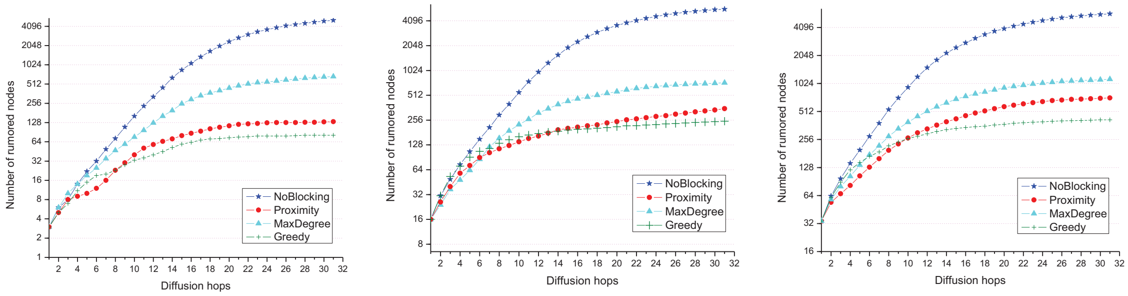
VI. Experiment Setting

We conduct experiments on our algorithms as well as two heuristics on several real-world networks. Our experiments aim at testing our algorithms from the following aspects: (a) effectiveness at different network density, here it means the average node degree; (b) effectiveness at different community size; (c) effectiveness at different rumor originators.

A. Data Collection

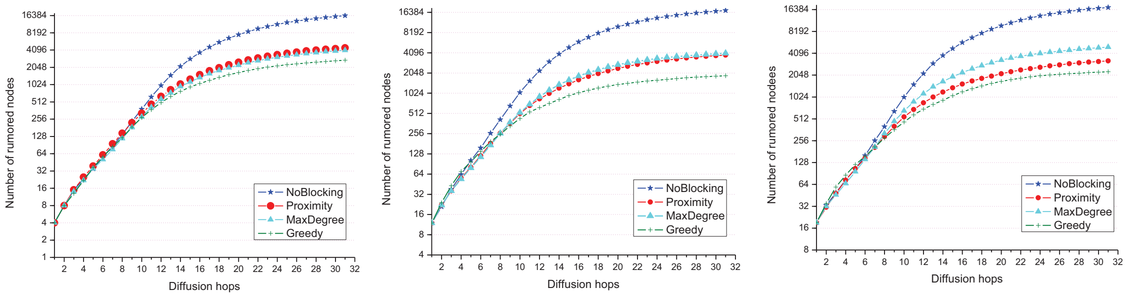
We use two real-world networks. The first one, namely Enroll Email communication network, is the same as used in [28], [29]. The second is a collaboration network which is used in the experimental study in [27], and it has been shown to capture many of the key features of social networks in [30].

1) Enron Email Communication Network: This network covers all the email communications within a dataset of around half million emails. Nodes of the graph represent email addresses and a directed edge from i to j means i sends at least one email to j. This dataset contains 36692 nodes connected by 367662 edges with an average node degree of 10.0.



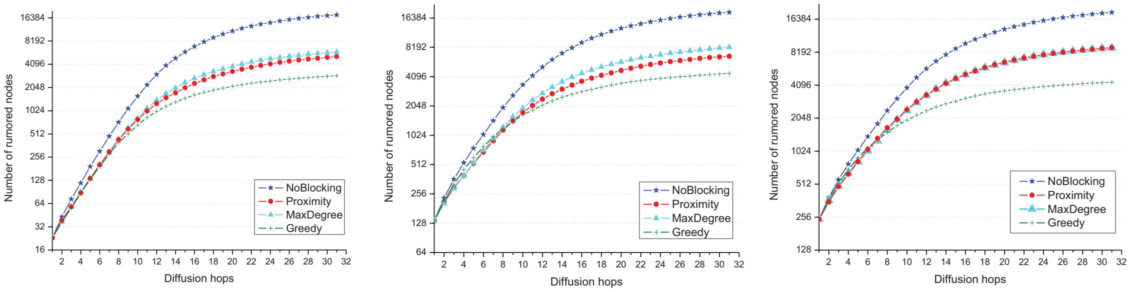
(a) $|R| = 1\%|C|, |P| = 68$ (b) $|R| = 5\%|C|, |P| = 90$ (c) $|R| = 10\%|C|, |P| = 112$

Figure 4. Infected nodes under the OPOAO model on Hep collaboration network with $|N| = 15233, |C| = 308, |B| = 387$.



(a) $|R| = 5\%|C|, |P| = 10$ (b) $|R| = 10\%|C|, |P| = 21$ (c) $|R| = 20\%|C|, |P| = 27$

Figure 5. Infected nodes under the OPOAO model on Enron Email network with $|N| = 36692, |C| = 80, |B| = 135$.



(a) $|R| = 1\%|C|, |P| = 28$ (b) $|R| = 5\%|C|, |P| = 74$ (c) $|R| = 10\%|C|, |P| = 98$

Figure 6. Infected nodes under the OPOAO model on Enron Email network with $|N| = 36692, |C| = 2631, |B| = 2250$.

2) Collaboration Network: Hep collaboration network is extracted from the e-print arXiv, and covers scientific collaborations between authors with papers submitted to High Energy Physics. In this network, nodes stand for authors and an undirected directed edge between i and j implies that i co-authors a paper with j. Since our problems are based on directed graph, we represent each undirected edge (i, j) by two directed edges (i, j) and (j, i). This dataset contains 15233 nodes connected by 58891 edges with an average node degree of 7.73.

B. Comparison Results

As introduced in our problems, the community structure of networks plays an important role for our methods. An accurate community structure that reflects the network topology will help us to validate the effectiveness of our algorithms. Since the partition of network communities is not the main point in our work, we use a community detection approach proposed by Blondel *et al.* [25], whose performance has been verified in [32]. After obtaining the community structure of a network, we choose different sizes of rumor communities and compute the number of the bridge ends. From the Enroll Email network, we select two communities, one with 2631 nodes and 2250 bridge ends, and the other with 80 nodes and 135 bridge ends. From the collaboration network, we select a community with 308 nodes and 387 bridge ends. Next, we evaluate the performance of our algorithms in comparison with two heuristics: MaxDegree and Proximity.

1) Comparison Algorithms:

MaxDegree: a basic algorithm, which simply chooses the nodes according to the decreasing order of node degree as the protectors.

Proximity: a simple heuristic algorithm, in which the direct out-neighbors of rumors are chosen as the protectors.

We exclude random algorithm due to its poor performance. Instead, a NoBlocking line is included to reflect the performance of these algorithms.

2) Experimental Results: To simplify our presentation, we denote by $|R|$ the number of rumor originators, $|P|$ the number of protector originators, $|C|$ community size, $|B|$ the number of the bridge ends, $|N|$ the number of nodes in a network. Since the number of infected nodes is large, we adopt the $\log$-time chart to report the simulation results.

Number of infected nodes under the OPOAO Model: Since it is time consuming for us to obtain the solution (the number of protector originators) for the LCRB-P problem, we evaluate the effectiveness of the three algorithms from another aspect: for the same number of protector and rumor originators, how many nodes will be infected in a network? The fewer number of nodes is infected, the better is the algorithm. We run the three competitive algorithms with 31 hops, where 1 hop means 1 step with respect to influence diffusion. Fig. 4 to Fig. 6 show the average results obtained by repeated Monte Carlo simulation.

As depicted in Fig. 4 to Fig. 6, the greedy algorithm performs best among all the three algorithms from around 9 hops, while in the early stages, both the Proximity and MaxDegree perform better. This is reasonable, since our algorithm chooses protector originators by considering the total effectiveness in protecting the bridge ends, indicating that some protectors may be far away from rumor originators. However, Proximity and MaxDegree merely base on the network topology, especially for Proximity. In the early steps, all the protectors selected by Proximity (selected randomly from the direct neighbors of rumor originators) can immunize a large number of nodes within local area. On the other hand, nodes with large degree scatter through networks, thus in the early steps, MaxDegree can choose nodes near rumor originators with high probability.

Moreover, we also observe that Proximity outperforms MaxDegree as shown in Fig. 4 to Fig. 6. Particularly, in Fig. 4, the difference of the performance of Proximity and MaxDegree is large, this is because in the Hep network, node degree is lower, indicating the number of neighbors of each node is small, which in turn means that smaller number of protectors are required to block rumor spread. However, in Fig. 5 and Fig. 6, both Proximity and MaxDegree achieve almost the same performance due to the high network density, under which Proximity needs large number of protectors to control rumor spread.

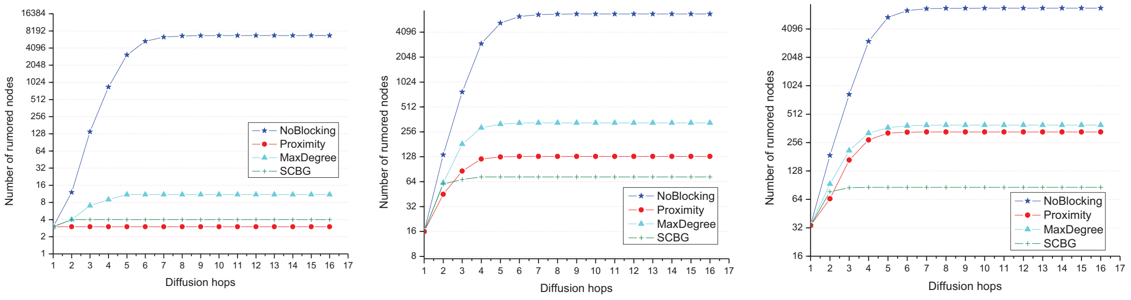
Table I
COMPARISON RESULTS FOR THE DOAM MODEL

Dataset/$\|N\|$/$\|C\|$	$\|R\|$	SCBG	Proximity	MaxDegree
Hep/15233/308	1%	32.9	25.3	140.6
	5%	42.1	74.3	147.8
	10%	48.9	133.8	152.6
Email/36692/80	5%	6.2	43.7	72.7
	10%	8.2	46.9	79.3
	20%	13.8	62.9	91.1
Email/36692/2631	1%	20.4	289.3	1208.8
	5%	50.9	1067.6	1350.2
	10%	68.4	1422.6	1683.8

As for the relative increase speed of the number of infected nodes (the fraction between newly infected nodes and early existing infected nodes), Fig. 4 to Fig. 6 report that in all the three algorithms, it does not increase, i.e., decrease or remain unchanged. This is reasonable, since along with the increase of the number of infected nodes, the chance of currently infected nodes being chosen as activation targets increases, which implies that the number of newly infected nodes may become smaller. Furthermore, these figures exhibit a common phenomenon that after 32 hops, the size of newly infected nodes is quite small for these three methods, and even the Noblocking line shows similar property.

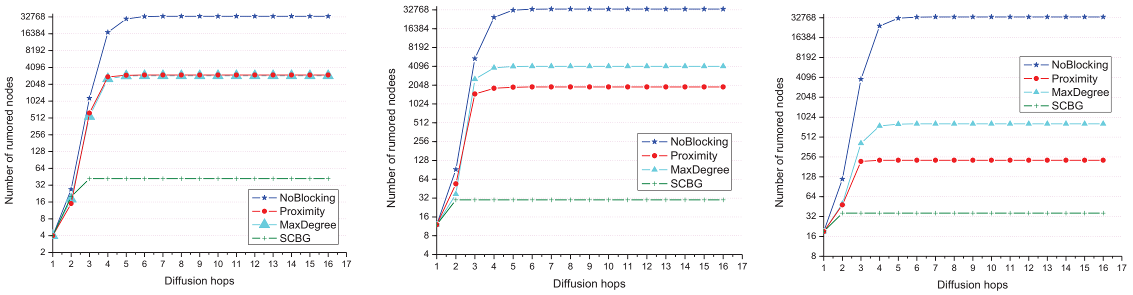
Number of Selected Protectors under the DOAM Model: Take a look at table I, in which the decimals represent the average number of protectors selected by each algorithm, our SCBG algorithm performs best in all test cases except the first one, which merely has 3 rumor originators. Our interpretation is that Hep network has low average node degree, when the size of rumor originators is small, only a few protectors are needed to block the spread of these rumors. Obviously, Proximity always performs better than MaxDegree in that Proximity can control rumor spread before large number of nodes are infected, while MaxDegree do not have this attribute.

Note that among the three communities, along with an increase in rumor originators, the number of protectors selected by our algorithms increases slowly compared with the other heuristics. Especially in the third community, whose size is $|C| = 2631$. When the number of rumor originators is increased from 27 ($1\%|C|$) to 132, the number of protector originators selected by our algorithm is increased from 20.4 to 50.9 (average value), that is, the change in the number of protector originators is about 30.5. However, the change in the number of of protector originators is about 778.3 and 141.4 in Proximity and MaxDegree, respectively. The results in this community clearly shows that the SCBG algorithm significantly outperforms both Proximity and MaxDegree in



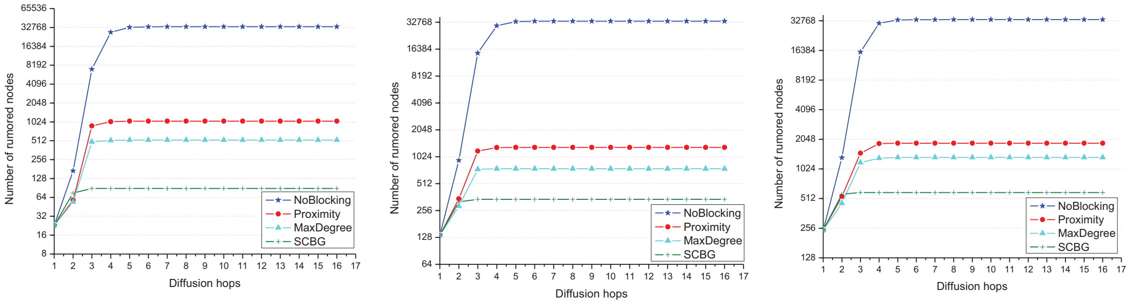
(a) $|R| = 1\%|C|, |P| = 34$ (b) $|R| = 5\%|C|, |P| = 44$ (c) $|R| = 10\%|C|, |P| = 55$

Figure 7. Infected nodes under the DOAM model on Hep collaboration network with $|N| = 15233, |C| = 308, |B| = 387$.



(a) $|R| = 5\%|C|, |P| = 8$ (b) $|R| = 10\%|C|, |P| = 11$ (c) $|R| = 20\%|C|, |P| = 14$

Figure 8. Infected nodes under the DOAM model on Enron Email network with $|N| = 36692, |C| = 80, |B| = 135$.



(a) $|R| = 1\%|C|, |P| = 21$ (b) $|R| = 5\%|C|, |P| = 52$ (c) $|R| = 10\%|C|, |P| = 69$

Figure 9. Infected nodes under the DOAM model on Enron Email network with $|N| = 36692, |C| = 2631, |B| = 2250$.

networks with large size and high density.

Number of infected nodes under the DOAM Model: In this part, we aim to test the effectiveness of the algorithms in protecting nodes in networks. For each community, we predetermine the size of protectors according to the SCBG algorithm. For the two heuristics, we compute their solutions first, then randomly choose the protectors with the predetermined size as the ones used in comparison implementation. From Fig. 7 to Fig. 9, we observe that rumors spread quite fast within the first 4 steps while after the 4th step almost no new nodes can be infected over all cases.

Except Fig. 7(a), (The Proximity protects 1 more node than the SCBG algorithm due to small rumor size and low network degree.), the SCBG algorithm always protects the most number of nodes in comparison with the two heuristics. Therefore, we believe that our algorithm can be applied to problems, in which the goal is, on one hand, to protect target nodes with least number of protectors, on the other hand, to reduce the number of nodes infected in networks at the end of cascade diffusion. We also notice that Proximity outperforms MaxDegree in Fig. 7 and Fig. 8 for different rumor sizes. However, in Fig. 9, MaxDegree performs better than Proximity. The reason is that this network has higher average node degree.

VII. Conclusion

In this paper, we study two variants of the least cost rumor blocking (LCRB) problem under the OPOAO (LCRB-P problem) and the DOAM models (LCRB-D problem), respectively. As for the OPOAO model, we show that the objective function for the LCRB-P problem is submodular, therefore, the classical greedy algorithm is employed to produce a $(1-1/e)-$approximation ratio. With regard to the DOAM model, we transfer the LCRB-D problem to the set cover problem, then propose the SCBG algorithm to achieve a $O(\ln n)-$factor solution for it. Finally, the simulation reports demonstrate that both the greedy algorithm and the SCBG algorithm outperform two heuristics: MaxDegree and Proximity. However, in the OPOAO model, the greedy algorithm is time consuming, therefore, finding efficient algorithms to overcome this drawback is a possible research direction. Furthermore, one could also study the LCRB problem under other influence diffusion models, especially models without submodularity property. Another direction is looking into the problem of locating rumor originators since in many real world situations, it is hard to quickly detect rumors in the first place.

Acknowledgment

This work is partially supported by the U.S. National Science Foundation under Grant CNS-0831579, CNS-1016320 and CCF-0829993, and by The Air Force Office of Scientific Research under Award No. FA-9550-09-1-0468. We thank Dr. Robert Herklotz for his support.

References

[1] http://news.yahoo.com/blogs/technology-blog/twitter-rumor-/ /leads-sharp-increase-price-oil-173027289.html

[2] http://www.iirme.com/Global/IIRME/Training/BC4560/ resources/How_Firms_Should_Fight_Rumors.pdf

[3] M. Richardson and P. Domingos, Mining knowledge-sharing sites for viral marketing. *In KDD*, 2002, pp. 61-70.

[4] P. Domingos and M. Richardson, Mining the network value of customers. *In KDD*, 2001, pp. 57-66.

[5] D. Kempe, J. M. Kleinberg and É. Tardos, Maximizing the spread of influence through a social network. *In KDD*, 2003.

[6] D. Kempe, J. M. Kleinberg and É. Tardos, Influential nodes in a diffusion model for social networks. *In ICALP*, 2005, pp. 1127–1138.

[7] M. Kimura, K. Saito and R. Nakano, Extracting influential nodes for information diffusion on a social network. *In AAAI*, 2007, pp. 1371-1376.

[8] 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*, 2008.

[9] M. Kimura, K. Saito and H. Motoda, Blocking links to minimize contamination spread in a social network. *ACM Trans. Knowl. Discov. Data*, 2009.

[10] W. Chen, Y. Wang and S. Yang, Efficient influence maximization in social networks. *In KDD*, 2009.

[11] W. Chen, Y. Yuan and L. Zhang, Scalable influence maximization in social networks under the linear threshold model. *In ICDM*, 2010, pp. 88-97.

[12] W. Chen, C. Wang and Y. Wang, Scalable influence maximization for prevalent viral marketing in large-scale social networks. *In KDD*, 2010, pp. 1029-1038.

[13] W. Chen, A. Collins, R. Cummings, T. Ke, Z. Liu, D. Rincn, X. Sun, Y. Wang, W. Wei and Y. Yuan, Influence maximization in social networks when negative opinions may emerge and propagate. *In SDM* , 2011, pp. 379-390.

[14] C. Budak, D. Agrawal and A. E. Abbadi, Limiting the spread of misinformation in social networks. *In WWW*, 2011, pp. 665-674.

[15] S. Bharathi, D. Kempe and M. Salek, Competitive influence maximization in social networks. *In WINE*, 2007, pp. 306-311.

[16] X. He, G. Song, W. Chen and Q. Jiang, Influence blocking maximization in social networks under the competitive linear threshold model. *SDM* , 2012.

[17] T. Carnes, C. Nagarajan, S. M. Wild and A. van Zuylen, Maximizing influence in a competitive social network, a followers perspective. *In ICEC*, 2007, pp. 351-360.

[18] U. Feige, A threshold of $\ln n$ for approximating set cover. *In ACM*, 1998, pp. 634-652.

[19] A. Borodin, Y. Filmus and J. Oren, Threshold models for competitive influence in social networks. *In WINE*, 2010, pp. 539-550.

[20] J. Kostka, Y. A. Oswald and R.Wattenhofer, Word of mouth: Rumor dissemination in social networks. *In SIROCCO*, 2008, pp. 185-196.

[21] R. Narayanam and Y. Narahari, Determining the top-k nodes in social networks using the shapley value. *In AAMAS*, 2008, pp. 1509-1512.

[22] C. Wang, J. C. Knight and M. C. Elder, On computer viral infection and the effect of immunization. *In ACSAC*, 2000, pp. 246-256.

[23] D. Trpevski, W. K. S. Tang and L. Kocarev, Model for rumor spreading over networks. *Physics Review E*, 2010.

[24] Y. Wang, G. Cong, G. Song and K. Xie, Community-based greedy algorithm for mining top-k influential nodes in mobile social networks. *In KDD*, 2010, pp. 1039-1048.

[25] V. D. Blondel, J. Guillaume, R. Lambiotte and E. Lefebvre, Fast unfolding of communities in large networks. *J. Stat. Mech.: Theory and Experiment*, 2008.

[26] N. P. Nguyen, G. Yan, M. T. Thai and S. Eidenbenz, Containment of Misinformation Spread in Online Social Networks. available online.

[27] J. Leskovec, J. Kleinberg and C. Faloutsos, Graph Evolution: Densification and Shrinking Diameters. *ACM Transactions on Knowledge Discovery from Data (ACM TKDD)*, 1(1), 2007.

[28] J. Leskovec, K. Lang, A. Dasgupta and M. Mahoney, Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters. *Internet Mathematics*, 6(1) pp. 29–123, 2009.

[29] B. Klimmt and Y. Yang, Introducing the Enron corpus. *CEAS conference*, 2004.

[30] M. Newman, The structure of scientific collaboration networks. *Proc. Natl. Acad. Sci*, 98, 2001.

[31] Z. Lu, W. Zhang, W. Wu, B. Fu and D. Du, Approximation and Inapproximation for The Influence Maximization Problem in Social Networks under Deterministic Linear Threshold Model. *In Proceedings of the 31st IEEE International Conference on Distributed Computing Systems Workshops*, pp. 160–165, 2011.

[32] A. Lancichinetti and S. Fortunato, Community detection algorithms: A comparative analysis. Physical review. E. 80, 2009.

[33] L. Fan, Z. Lu, W. Wu, Y. Bi and A. Wangz, A new model for product adoption over social networks. *In 19th International Computing and Combinatorics Conference*, CSoNet'13, 2013.