

# Multiple Source Detection without Knowing the Underlying Propagation Model

Zheng Wang, Chaokun Wang,\* Jisheng Pei, Xiaojun Ye

School of Software, Tsinghua University, Beijing 100084, P.R. China  
Tsinghua National Laboratory for Information Science and Technology (TNList)  
{zheng-wang13, pjs07}@mails.tsinghua.edu.cn; {chaokun, yexj}@tsinghua.edu.cn

## Abstract

Information source detection, which is the reverse problem of information diffusion, has attracted considerable research effort recently. Most existing approaches assume that the underlying propagation model is fixed and given as input, which may limit their application range. In this paper, we study the multiple source detection problem when the underlying propagation model is unknown. Our basic idea is *source prominence*, namely the nodes surrounded by larger proportions of infected nodes are more likely to be infection sources. As such, we propose a multiple source detection method called *Label Propagation based Source Identification (LPSI)*. Our method lets infection status iteratively propagate in the network as labels, and finally uses local peaks of the label propagation result as source nodes. In addition, both the convergent and iterative versions of LPSI are given. Extensive experiments are conducted on several real-world datasets to demonstrate the effectiveness of the proposed method.

## Introduction

Information diffusion is one of the most important topics in social network research (Centola 2010; Wang et al. 2016). Recently, people are more interested in its reverse problem: Given a snapshot of a partially infected network, can we identify the infection sources? The answer to this problem has vast applications in mitigating the damage of epidemics caused by infectious diseases, rumor spreading in social media, and so forth.

The abovementioned *multiple source detection problem* has attracted many researchers (Prakash, Vreeken, and Faloutsos 2012; Luo, Tay, and Leng 2013; Zang et al. 2015; Chen, Zhu, and Ying 2016). These studies deal with this problem differently according to the types of the used propagation models such as the Susceptible-Infected (SI) model (Anderson, May, and Anderson 1992) and Susceptible-Infected-Recovered (SIR) model (Allen 1994), i.e., they all assume that the underlying propagation model is fixed and known.

However, in practice, identifying the correct propagation model always needs prior knowledge, which limits the application range of source detection methods. For instance, it is

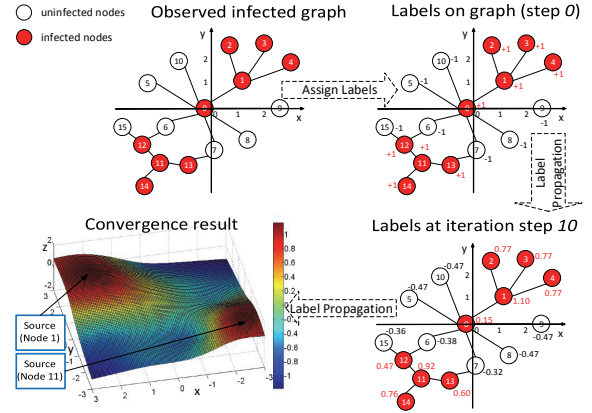


Figure 1: The framework of our LPSI approach to the multiple source detection problem.

hard to choose an appropriate propagation model for a new infectious disease or online rumor. Moreover, it is difficult to acquire the true values of parameters in the pre-selected underlying propagation model. Therefore, it is necessary but challenging to detect infection sources without knowing the underlying propagation model.

In this paper, we address the multiple source detection problem based on the idea of *source prominence*, namely the nodes surrounded by larger proportions of infected nodes are more likely to be source ones. To better understand the basic idea, one can imagine that a part of the network has been infected by an infection source. On the one hand, at the margin of the infected region, nodes tend to have less infected neighbors. On the other hand, in the center of the infected region, nodes tend to have more infected neighbors. This intuition is reasonable in most existing propagation models, such as the SI and SIR models.

Inspired by the primary idea of source prominence, we propose a multiple source detection method called *Label Propagation based Source Identification (LPSI)*. Our approach tries to automatically identify actual source nodes without knowing the underlying propagation model. The general process is as follows. We first assign positive labels to the infected nodes, and negative labels to the uninfected nodes in the network. Then we iteratively propagate label in-

\*Corresponding author: Chaokun Wang.

formation among nodes based on a probability matrix which is generated according to the network structure. Finally, we get the convergence result, where “source” nodes are shown as local peaks with the highest “infected” label values.

**Example:** As an example, Fig. 1 depicts a partially infected network, in which a sub-network has been infected by a stochastic process starting from two sources (nodes 1 and 11). The red nodes are infected nodes and the white ones are uninfected. At first, we assign positive labels (+1) to infected ones, and negative labels (-1) to uninfected ones. After that, their label values are propagated and updated iteratively, and the propagation result at iteration step 10 is shown at the bottom right of Fig. 1. Finally, we get the convergence result, where nodes 1 and 11 are two local maximum points. As a result, we consider these two nodes as infection sources.

**Contributions:** The major contributions of this paper are summarized as follows:

- We present and formalize the multiple source detection problem without knowing the underlying propagation model, which has rarely been mentioned in the literature.
- We propose the Label Propagation based Source Identification (LPSI) method for the above problem. In addition, both the convergent and iterative versions of LPSI are brought forward.
- Extensive experiments conducted on real-world datasets demonstrate the effectiveness and efficiency of our methods in identifying the actual source nodes.

## Preliminaries

In this section, we briefly review some propagation models proposed so far, and formulate our problem.

### Propagation Models

According to (Easley and Kleinberg 2010), existing propagation models could be categorized as either infection models or influence models, with respect to their intended applications.

**Infection Models** To describe the transmission of communicable disease through individuals, various infection (epidemic) models are proposed, such as Susceptible-Infected (SI) model (Anderson, May, and Anderson 1992) and Susceptible-Infected-Recovered (SIR) model (Allen 1994).

In the SI model, each node is in one of two states: susceptible ( $S$ ) or infected ( $I$ ). Once a node is infected, it stays infected forever. In each discrete time step, each infected node tries to infect its susceptible (uninfected) neighbors independently with probability  $p$ , which reflects the strength of the disease spread. While in the SIR model, each node has three possible states: susceptible ( $S$ ), infected ( $I$ ) and recovered ( $R$ ). The infection process is similar except that infected nodes can recover with probability  $q$ . In addition, recovered nodes cannot be infected any more.

**Influence Models** In order to model how users influence each other in a social network, researchers have proposed many influence models, such as Independent-Cascade

(IC) model (Goldenberg, Libai, and Muller 2001) and Linear Threshold (LT) Model (Kempe, Kleinberg, and Tardos 2003).

In both IC and LT models, every node is represented as a binary variable with either active (infected) or inactive (susceptible) status. The major difference between these two models is the way how an active node influences its neighbors. In the IC model, when a node first becomes active at a time step, it has exactly one chance to independently influence (infect) its susceptible neighbors, and cannot activate neighbors in subsequent rounds. While in the LT model, the sum of incoming edge influence degrees on any node is assumed to be at most 1 and every node has an activation threshold uniformly at random from  $[0, 1]$ . At each time step, a node is activated by their activated neighbors if the sum of influence degrees exceeds its threshold. In both models, the influence propagates until no more nodes can become active.

## Multiple Source Detection Problem

The multiple source detection problem studied in this paper can be formulated as follows: Given a social network  $G = (V, E)$ , an infection node vector  $Y = (Y_1, \dots, Y_{|V|})$  where  $Y_i = 1$  indicates node  $i$  is infected and  $Y_i = -1$  otherwise, the goal is to find the original infection source set  $\mathcal{S} \subset V$ .

Note that in the above definition, we do not assume the underlying propagation model is known. As such, our method is propagation model independent, which has a broader application range compared to state-of-the-art methods.

## The LPSI Method

We formally present our LPSI method (in Alg. 1) in this section. The aim of LPSI is to identify the original infection source number as well as source nodes of a partially infected network, which can be achieved by the following three steps.

### Step 1: Assign labels to the partially infected network

Label vector  $\mathcal{G}^t$ , initiated with the infection node vector  $Y$ , is used to assign labels to the nodes at time  $t$  in network  $G$ . In other words, at the beginning, we assign positive labels (+1) and negative labels (-1) to infected and uninfected nodes in the network, respectively.

### Step 2: Label propagation on the network

Before starting label propagation, we should build a weight matrix which decides the label propagation probability among nodes. We build the weight matrix  $W$  on edge set  $E$ , where  $W_{ij}=1$  represents there is an edge between node  $i$  and node  $j$ . This matrix is further symmetrically normalized as  $S$  (Line 2 in Alg. 1), in which  $S_{ij}$  represents the label propagation probability from node  $j$  to node  $i$ .

Based on the matrix  $S$ , we propagate labels on the network iteratively. In each iteration, each node gets a fraction of label information from its neighborhood, and retains some label information of its initial state. Therefore the label value of node  $i$  at time  $t+1$  becomes:

$$\mathcal{G}_i^{t+1} = \alpha \sum_{j: j \in \mathcal{N}(i)} S_{ij} \mathcal{G}_j^t + (1 - \alpha) Y_i \quad (1)$$

where  $0 < \alpha < 1$  is the fraction of label information that node  $i$  gets from its neighbors, and  $\mathcal{N}(i)$  represents the

---

**Algorithm 1** Label Propagation based Source Identification (LPSI)

---

**Input:** The infected network  $G=(V, E)$ , parameter  $\alpha$  ;  
The initial infection node vector  $Y$ .

**Output:** The source set  $\mathcal{S}$ .

```
1: Form the weight matrix  $W$  defined by  $W_{ij} = 1$  if there
   exists an edge connecting nodes  $i$  and  $j$ ;
2: Construct the matrix  $S = D^{-1/2}WD^{-1/2}$ , where  $D$  is
   a diagonal matrix with its  $(i,i)$ -element equal to the sum
   of the  $i$ -th row of  $W$ ;
3:  $\mathcal{G}^{t=0} \leftarrow Y$ ;
4: while  $\mathcal{G}^t$  does not reach the convergence  $\mathcal{G}^*$  do
5:   for each node  $i$  do
6:      $\mathcal{G}_i^{t+1} = \alpha \sum_{j:j \in \mathcal{N}(i)} S_{ij} \mathcal{G}_j^t + (1 - \alpha) Y_i$ ;
7:   end for
8:    $t = t + 1$ ;
9: end while
10:  $\mathcal{S} = \{\}$ ;
11: for each original infected node  $i$  do
12:   if  $\mathcal{G}_i^* >$  all  $i$ 's neighbors'  $\mathcal{G}^*$  value then
13:      $\mathcal{S} = \mathcal{S} \cup \{i\}$ ;
14:   end if
15: end for
16: return  $\mathcal{S}$ ;
```

---

neighborhood of node  $i$ . We can stop this iteration when convergence is reached. Here, “convergence” means the label values of nodes will not change in several successive iterations of label propagation (the convergence analysis can be found in the next section).

**Step 3: Sources Identification** Suppose the label vector  $\mathcal{G}^t$  finally converges to  $\mathcal{G}^*$  at the end of the above label propagation process. One node  $i$  is identified as a source node if it satisfies the following two conditions: 1) node  $i$  is an infected node initially, i.e.,  $Y_i = 1$ ; and 2) its final label value  $\mathcal{G}_i^*$  is larger than those of its neighbors.

The first condition reflects the fact that infected nodes are more likely to be sources than uninfected ones. Although this selection may miss a few recovered source nodes under some propagation models such as the SIR model, it avoids the confusion of the nodes which have never been infected. The second condition ensures the detected sources should be local maximum points in the label propagation result, which keeps consistent with the primary idea of source prominence. As such, these local maxima are determined by both node infection status (labels) and network structure. More explanations about the local maxima could be found in the next section.

### Algorithm Analysis

In this section, we analyze the properties of convergence and local maxima in the label propagation process of LPSI. In addition, we discuss the relationship between the idea of source prominence and propagation models.

### Convergence Analysis

In our LPSI method, the iteration equation of the label propagation (Eq. 1) can be rewritten as  $\mathcal{G}^{t+1} = \alpha S \mathcal{G}^t + (1 - \alpha) Y$ . By the initial condition that  $\mathcal{G}^0 = Y$ , we have:

$$\mathcal{G}^t = (\alpha S)^t Y + (1 - \alpha) \sum_{i=0}^{t-1} (\alpha S)^i Y. \quad (2)$$

As proved in (Zhou et al. 2004; Wang and Zhang 2008), the parameter  $0 < \alpha < 1$  and normalized matrix  $S$  will lead:  $\lim_{t \rightarrow \infty} (\alpha S)^t = 0$ , and  $\lim_{t \rightarrow \infty} \sum_{i=0}^{t-1} (\alpha S)^i = (I - \alpha S)^{-1}$ , where  $I$  is an  $n \times n$  identity matrix. Consequently, the iteration will converge to:

$$\mathcal{G}^* = (1 - \alpha)(I - \alpha S)^{-1} Y. \quad (3)$$

Therefore, the label propagation iteration in Algorithm 1 will finally converge. In addition, Eq. 3 shows that we can obtain the convergence result directly without any iterations.

### Local Maxima in Label Propagation

In our LPSI method, the convergence label vector  $\mathcal{G}^*$  minimizes the following cost function (Zhou et al. 2004):

$$\mathcal{Q}(\mathcal{G}) = \frac{1}{2} \left( \sum_{i,j=1}^n W_{ij} \left\| \frac{\mathcal{G}_i}{\sqrt{D_{ii}}} - \frac{\mathcal{G}_j}{\sqrt{D_{jj}}} \right\|^2 + \mu \sum_{i=1}^n \|\mathcal{G}_i - Y_i\|^2 \right). \quad (4)$$

The first term of the right-hand side in the cost function is the smoothness constraint, which means that label values (i.e., infection status) should not change too much between connected nodes. In this constraint, the difference between two nodes is further normalized by their degrees, which keeps consistent with the basic idea of source prominence, i.e., the nodes surrounded by larger proportions of infected nodes tend to have higher infected label values. Similarly, the nodes surrounded by larger proportions of uninfected nodes tend to have lower infected (i.e., higher uninfected) label values. Thus, the infected label values of nodes increase as they get closer to source nodes. Intuitively, a “source” is likely to be a local maximum point surrounded by a group of neighboring nodes, whose infected label values decrease with respect to their distance from the source.

The second term of the right-hand side in Eq. 4 is the fitting constraint, which means that the final label propagation result  $\mathcal{G}^*$  should not change too much from the original label assignment (i.e., initial infection status). This trade-off between these two constraints is captured by the parameter  $\mu$  which has a linear relationship with the parameter  $\alpha$  in Eq. 3 (Zhou et al. 2004).

### Source Prominence vs. Propagation models

In this section we revisit the aforementioned two types of propagation models, i.e., infection models and influence models. Regardless of different propagation models, nodes close to source nodes would have a higher probability to get infected (activated) than the nodes far away from source nodes. It can be explained by the fact that the infection initially starts from source nodes, and is further propagated to the rest of the network. Clearly, this phenomenon exists

in the general propagation progress. Therefore, the source prominence effect should hold in most existing propagation models, which is further verified in our later experiments. In addition, the same intuition has been adopted in several existing studies, such as (Prakash, Vreeken, and Faloutsos 2012), (Shah and Zaman 2011) and (Zang et al. 2015).

## Two Versions of Our Method

Based on the above convergence analysis, in this section, we first present the convergent version of our method. Then, to balance accuracy and efficiency, we further give the iterative version of our method.

### The Convergent Version

LPSI<sub>con</sub> is the convergent version of LPSI method. The “convergent” means we get the convergence result of the label propagation (Lines 3 to 9 in Alg. 1), which can be obtained either by the iteration procedure or by Eq. 3.

**Time Complexity** Algorithm 1 first builds two matrices (Lines 1 and 2), and the running time is  $O(|E|)$ , where  $E$  is the edge set. Then it gets the convergence result of label propagation (Lines 3 to 9), and the running time is  $O(N^3)$ <sup>1</sup>, where  $N$  is the node number. Finally, it finds local maxima (Lines 11 to 15), and the running time is  $O(L * N)$ , where  $L$  is the average number of neighbors per node. Consequently, the overall complexity of LPSI<sub>con</sub> is  $O(N^3)$ .

### The Iterative Version

LPSI<sub>iter</sub> is the iterative version of LPSI method. The “iterative” means the propagation result is obtained in a few iterations (Lines 3 to 9 in Alg. 1), i.e., the iteration terminates without considering whether convergence is guaranteed.

As shown above, getting the convergence result of label propagation has high complexity ( $O(N^3)$ ), which may be prohibitive for some practical applications. However, to capture the intuition of source prominence, the convergence result may not be necessary. Another quantity of interest is: how many iteration steps are needed to capture this intuition? In our later experiments, we show that a small iteration number (5 in our evaluation) is enough.

**Time Complexity** The time complexity of the iterative version of label propagation (Lines 3 to 9 in Alg. 1) is  $O(t * L * N)$ , where  $L$  is the average number of neighbors per node and  $t$  is the number of iterations. In addition, the time complexity of the other steps in Alg. 1 is  $O(|E| + L * N)$ . Consequently, the overall time complexity of LPSI<sub>iter</sub> is  $O(t * L * N)$ . Note that,  $L * N$  can be described by the edge number  $|E|$ . Therefore, LPSI<sub>iter</sub> has a linear complexity with respect to the number of edges.

## Experiments

### Experimental Setup

**Datasets** As stated in Table 1, we use the following three real-world datasets:

<sup>1</sup>Here we actually calculate the convergence by Eq. 3, and adopt the fact that time complexity of matrix inversion is close to  $O(N^3)$  (Zhu, Lafferty, and Rosenfeld 2005).

Table 1: Datasets

Dataset	#Nodes	#Edges	#avg(degree)
KARATE	34	78	4.6
Jazz	198	2,742	27.7
Ego-Facebook	4,039	88,234	43.7

1. KARATE (Zachary 1977) is a social network of friendships between 34 members of a karate club at a US university in the 1970s.
2. Jazz (Gleiser and Danon 2003) is a network of Jazz bands performing from 1912 to 1940.
3. Ego-Facebook (Leskovec and Mcauley 2012) is a Facebook graph dataset obtained from survey participants.

**Propagation models** As mentioned above, existing propagation models can be categorized into infection models and influence models. To evaluate our method extensively, in each of these categories, we consider two representative propagation models. We test two different infection models: SI model and SIR model. As the same in (Zhu and Ying 2016; Luo 2015; Zhu and Ying 2014), the infection probability  $p$  is chosen uniformly from  $(0, 1)$  for the SI model, and an extra recovery probability  $q$  is chosen uniformly from  $(0, p)$  for the SIR model.

In addition, we evaluate two different influence models: IC model and LT model. In the IC model, the infection probability  $p$  is chosen uniformly from  $(0, 1)$ . In the LT model, as in (Kempe, Kleinberg, and Tardos 2003), we treat the infection weights among nodes as follows. If nodes  $u, v$  have degrees  $d_u$  and  $d_v$ , then the infection weight of edge  $(u, v)$  is  $1/d_v$ , and edge  $(v, u)$  has weight  $1/d_u$ . Furthermore, the threshold of each node is uniformly chosen from a small interval  $[0, 0.5]$ , so as to infect a large part of the network<sup>2</sup>.

**Comparing Methods** We test the convergent version (LPSI<sub>con</sub>) and the iterative version (LPSI<sub>iter</sub>) of our LPSI method under both SI and SIR models. Under the SI model, we compare these two versions with **Net-Sleuth** (Prakash, Vreeken, and Faloutsos 2012). Under the SIR model, we compare these two versions with **Zang’s** method (Zang et al. 2015). In addition, a variant of Zang’s method (denoted as **Zang\_si**<sup>3</sup>) is tested under the SI model. To date, these comparing methods are the latest and most well-known solutions for the multiple source detection problem.

Since there are few (comparable) works under the IC and LT models, we only test LPSI<sub>con</sub> and LPSI<sub>iter</sub> here. An overview of this comparison can be found in Table 2.

In both LPSI<sub>con</sub> and LPSI<sub>iter</sub>, we set the parameter  $\alpha=0.5$ . In addition, to show the effectiveness of LPSI<sub>iter</sub>, its iteration number is set to a small one (5 in this study). All

<sup>2</sup> (Chen, Wang, and Yang 2009) and (Kempe, Kleinberg, and Tardos 2003) have shown that if the threshold is chosen from  $[0, 1]$ , it is hard for a small set of sources to infect a large part of network.

<sup>3</sup>The original Zang’s method is just designed for the SIR model, in which recovered nodes should be identified first. We can ignore this recovery step and use the remaining steps to detect sources under the SI model.

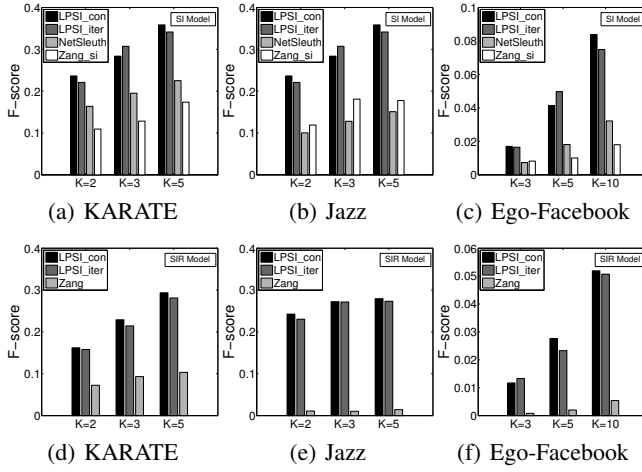


Figure 2: Source detection accuracies under infection models, i.e., SI model (row 1) and SIR model (row 2).

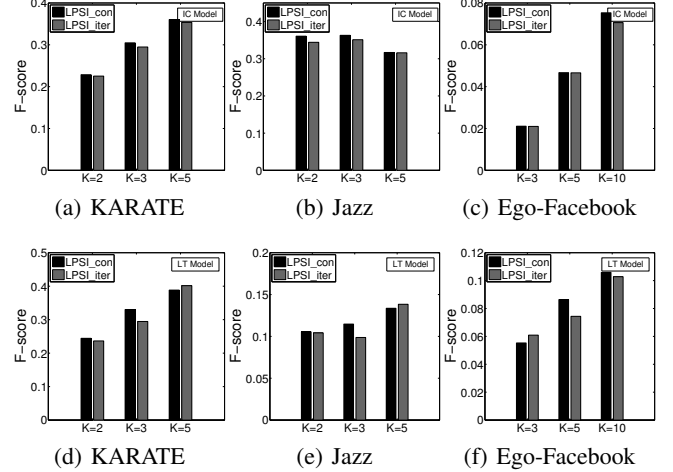


Figure 3: Source detection accuracies under influence models, i.e., IC model (row 1) and the LT model (row 2).

Table 2: Comparing Methods under Infection Models

Method	SI model	SIR model	IC model	LT model
LPSI.con	✓	✓	✓	✓
LPSI.iter	✓	✓	✓	✓
NetSleuth	✓			
Zang		✓		
Zang_si	✓			

parameters in other methods are adjusted to achieve the best performance.

All algorithms are implemented in Matlab. The program runs on a server with Intel(R) Core(TM) i7-2600 3.40GHz CPU and 32 GB memory.

**Experimental Settings** For an extensive comparison, we compare these methods on all datasets with different source numbers. For the small datasets KARATE and Jazz, we vary the source number  $K=2, 3, 5$ . For the other larger dataset Ego-Facebook, we vary the source number  $K=3, 5, 10$ , which is known to be the largest source number that has been evaluated (Prakash, Vreeken, and Faloutsos 2012; Zang et al. 2015).

All reported results are averaged over 500 independent runs. In each run, we first randomly generate a set of sources in the dataset. As the same in (Prakash, Vreeken, and Faloutsos 2012), we then simulate an infection till at least 30%<sup>4</sup> of the network is infected, and give the resulting footprint as input. Finally, we use different methods to detect the source set so as to evaluate their performance.

### Evaluation of Source Detection

We compare the source detection accuracy of different methods. The standard *recall*, *precision* and *F-score* metrics are used to validate this accuracy by comparing the detected

<sup>4</sup>Since KARATE is a small dataset compared to the tested source numbers, we set the max infect rate to 50% for this dataset.

source set with the actual source set. Figures 2 and 3 show the experimental results under infection models and influence models, respectively. Due to space constraints, we only show the F-score of these methods, and the complete listing of results is available on the author’s homepage.

**Evaluation under Infection Models** Figure 2 shows the experimental results under the SI model and SIR model. The first observation is that even without knowing the underlying infection model, LPSI.con and LPSI.iter still significantly outperform NetSleuth and Zang’s methods. This superiority becomes more remarkable as the network size increases. On the small datasets (KARATE and Jazz), in terms of F-score, LPSI.con and LPSI.iter outperform the other two methods by 50%~200% relatively. On the other larger dataset (Ego-Facebook), the outperformance is more pronounced (around 100%~500% relatively). The reason may be that our LPSI method is designed based on the idea of source prominence which holds under commonly used infection models. In contrast, the other two methods fail to identify the correct sources in most cases. As stated in (Prakash, Vreeken, and Faloutsos 2012), NetSleuth tends to return the most likely “sources” which could re-produce the given infected network, rather than real sources. On the other hand, experimental results in (Zang et al. 2015) also show that Zang’s method can hardly locate the real sources, but could only locate “approximate sources” near the real sources even when the source number is given.

The second observation is that LPSI.con and LPSI.iter could handle the multiple source detection problem under both SI and SIR models. As shown in Fig. 2, when the epidemic model switches from the SI to SIR model, the results of F-score only suffer a slight decline. This indicates that the source prominence exists under both infection models, although it seems a little less significant under the SIR model.

The third observation is that even under a small iteration number setting (5 in our experiments), LPSI.iter is still competitive with LPSI.con. This means that the source promi-

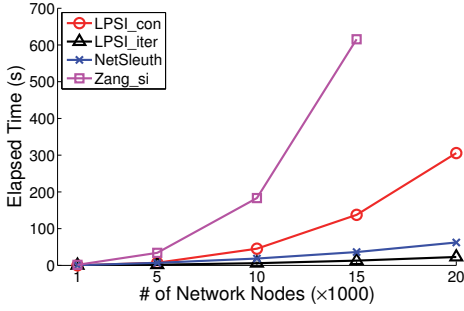


Figure 4: Scalability on synthetic data.

nence could be easily captured by LPSI.iter in a few iteration steps. It also indicates this iterative version has a good time performance, which is verified in the following experiments.

**Evaluation under Influence Models** Figure 3 shows the performance of LPSI.con and LPSI.iter under the IC model and LT model. We can see that our proposed methods still reach the similar performance under these two influence models, which indicates that the source prominence could also be easily captured by LPSI.iter under influence models.

The other observation is that the performance of LPSI.con and LPSI.iter does not vary a lot between infection models and influence models. For instance, in KARATE and Ego-Facebook datasets, the F-score values of these two methods are very similar. Although the performance of these two methods declines in the Jazz dataset under the LT model, we also find that the detection accuracy would be similar as that under the IC model when the activation threshold in LT model is uniformly chosen from other values such as  $[0, 0.2]$  or  $[0, 0.4]$ . These results indicate that the source prominence effect exists under the general propagation models.

### Scalability

Scalability analysis is performed on the synthetic scale-free networks (Barabási and Albert 1999) under the SI model. The performance under other models is similar, so we omit it here. We vary the number of nodes in the network and test the elapsed time. Figure 4 shows the results. Zang’s method is the most computationally intensive algorithm, which contains the leading eigenvector based community detection (Zang et al. 2015; Wang et al. 2015) and betweenness centrality calculation. As expected, the convergence method LPSI.con is also time-costly. In contrast, the time costs of both LPSI.iter and Netsleuth increase similarly and slowly with the increase of the network size. This is because they both have linear complexity with respect to the number of edges of the network. In addition, LPSI.iter can always keep significantly lower time cost than NetSleuth.

### Impact of Parameter $\alpha$

The parameter  $\alpha$  in our method (Eq. 1) is introduced to control the effects from neighbors during the label (infection status) propagation process. Therefore, we investigate the impact of  $\alpha$  via analyzing how its changes would affect the

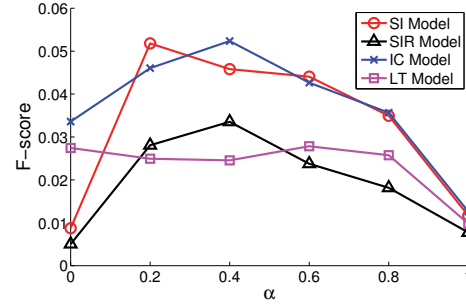


Figure 5: Parameter  $\alpha$  in LPSI.con on Ego-Facebook with  $K=5$ .

performance of LPSI.con in terms of F-score. (The performance of LPSI.iter is similar, so we omit it here for the limitation of space.) Figure 5 shows the results on Ego-Facebook ( $\#source=5$ ) under all four mentioned propagation models.

We observe that no matter under which propagation model setting, the F-score values decrease when  $\alpha$  approaches 0 or 1. On the other hand, the performances with  $\alpha \in [0.2, 0.6]$  are always stable and preferred. These observations confirm with the intuition that we should consider both the initial infection status and the effects from neighbors for source detection.

### Related Work

The information source detection problem has been extensively studied recently. In general, existing work can be divided into two categories. The first category focuses on the single source detection problem. (Shah and Zaman 2010; 2011) introduced and formalized the problem of identifying the single source of an epidemic under the SI model. (Zhu and Ying 2013) studied the single source detection problem under the SIR model. (Zhu and Ying 2014) further studied this problem with sparse observations, and (Shen et al. 2016) considered the infection time information.

The second category focuses on the multiple source detection problem. (Lappas et al. 2010) studied the problem of identifying  $K$  effectors under the IC model, in which the source number  $K$  should be specified manually. (Luo, Tay, and Leng 2013) considered the multiple source detection problem under the SI model, when the number of infection sources is bounded. The work most related to ours is (Prakash, Vreeken, and Faloutsos 2012) and (Zang et al. 2015), which could automatically identify the source number as well as the actual source nodes under the SI model and SIR model, respectively.

Contrary to assuming that the underlying propagation model is fixed and given as input, we consider the multiple source detection problem when the propagation model is unknown in this work.

### Conclusion

In this paper, we study the multiple source detection problem when the underlying propagation model is unknown. Based on the idea of source prominence, we introduce a multiple

source detection method LPSI. In addition, both the convergent and iterative versions of LPSI are given. Extensive experimental results show that even without knowing the underlying propagation model, these two versions still attain high accuracy in detecting the source nodes. In particular, the iterative version of LPSI achieves high scalability as well as superior performance. These inspiring results indicate that this work expands the application range of multiple source detection methods.

## Acknowledgements

Special thanks to Dr. B. Aditya Prakash who provided the source code of NetSleuth. This work is supported in part by the National Natural Science Foundation of China (No. 61373023, No. 61170064).

## References

- Allen, L. J. 1994. Some discrete-time si, sir, and sis epidemic models. *Mathematical biosciences* 124(1):83–105.
- Anderson, R. M.; May, R. M.; and Anderson, B. 1992. *Infectious diseases of humans: dynamics and control*, volume 28. Wiley Online Library.
- Barabási, A.-L., and Albert, R. 1999. Emergence of scaling in random networks. *Science* 286(5439):509–512.
- Centola, D. 2010. The spread of behavior in an online social network experiment. *science* 329(5996):1194–1197.
- Chen, W.; Wang, Y.; and Yang, S. 2009. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 199–208. ACM.
- Chen, Z.; Zhu, K.; and Ying, L. 2016. Detecting multiple information sources in networks under the sir model. *IEEE Transactions on Network Science and Engineering* 3(1):17–31.
- Easley, D., and Kleinberg, J. 2010. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press.
- Gleiser, P. M., and Danon, L. 2003. Community structure in jazz. *Advances in complex systems* 6(04):565–573.
- 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(3):211–223.
- Kempe, D.; Kleinberg, J.; and Tardos, É. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 137–146. ACM.
- Lappas, T.; Terzi, E.; Gunopulos, D.; and Mannila, H. 2010. Finding effectors in social networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1059–1068. ACM.
- Leskovec, J., and McAuley, J. J. 2012. Learning to discover social circles in ego networks. In *Advances in neural information processing systems*, 539–547.
- Luo, W.; Tay, W. P.; and Leng, M. 2013. Identifying infection sources and regions in large networks. *IEEE Transactions on Signal Processing* 61(11):2850–2865.
- Luo, W. 2015. *Identifying infection sources in a network*. Ph.D. Dissertation, Nanyang Technological University.
- Prakash, B. A.; Vreeken, J.; and Faloutsos, C. 2012. Spotting culprits in epidemics: How many and which ones? In *IEEE 12th International Conference on Data Mining (ICDM)*, 11–20. IEEE.
- Shah, D., and Zaman, T. 2010. Detecting sources of computer viruses in networks: theory and experiment. In *ACM SIGMETRICS Performance Evaluation Review*, volume 38, 203–214. ACM.
- Shah, D., and Zaman, T. 2011. Rumors in a network: Who’s the culprit? *IEEE Transactions on Information Theory* 57(8):5163–5181.
- Shen, Z.; Cao, S.; Wang, W.-X.; Di, Z.; and Stanley, H. E. 2016. Locating the source of diffusion in complex networks by time-reversal backward spreading. *Physical Review E* 93(3):032301.
- Wang, F., and Zhang, C. 2008. Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering* 20(1):55–67.
- Wang, M.; Wang, C.; Yu, J. X.; and Zhang, J. 2015. Community detection in social networks: an in-depth benchmarking study with a procedure-oriented framework. *Proceedings of the VLDB Endowment* 8(10):998–1009.
- Wang, Z.; Wang, C.; Pei, J.; Ye, X.; and Yu, P. S. 2016. Causality based propagation history ranking in social networks. In *IJCAI*.
- Zachary, W. W. 1977. An information flow model for conflict and fission in small groups. *Journal of anthropological research* 452–473.
- Zang, W.; Zhang, P.; Zhou, C.; and Guo, L. 2015. Locating multiple sources in social networks under the sir model: A divide-and-conquer approach. *Journal of Computational Science*.
- Zhou, D.; Bousquet, O.; Lal, T. N.; Weston, J.; and Schölkopf, B. 2004. Learning with local and global consistency. *Advances in neural information processing systems* 16(16):321–328.
- Zhu, K., and Ying, L. 2013. Information source detection in the sir model: A sample path based approach. In *Information Theory and Applications Workshop (ITA), 2013*, 1–9. IEEE.
- Zhu, K., and Ying, L. 2014. A robust information source estimator with sparse observations. *Computational Social Networks* 1(1):1–21.
- Zhu, K., and Ying, L. 2016. Information source detection in the sir model: a sample-path-based approach. *IEEE/ACM Transactions on Networking* 24(1):408–421.
- Zhu, X.; Lafferty, J.; and Rosenfeld, R. 2005. *Semi-supervised learning with graphs*. Carnegie Mellon University, Language Technologies Institute, School of Computer Science.