Exploring Influence Maximization in Location-Based Social Networks

Shasha Li, Kai Han(✉), and Jiahao Zhang

School of Computer Science and Technology/Suzhou Institute for Advanced Study, University of Science and Technology of China, Hefei, People's Republic of China
{lisa1990,jhcheung}@mail.ustc.edu.cn, hankai@ustc.edu.cn

Abstract. In the last two decades, the issue of Influence Maximization (IM) in traditional online social networks has been extensively studied since it was proposed. It is to find a seed set which has maximum influence spread under a specific network transmission model. However, in real life, the information can be spread not only through online social networks, but also between neighbors who are close to each other in the physical world. Location-Based Social Network (LBSN) is a new type of social network which is emerging increasingly nowadays. In a LBSN, users can not only make friends, but also share the events they participate in at different locations by checking in. In this paper, we aim to study the IM in LBSNs, where we consider both the influence of online and offline interactions. A two-layer network model and an information propagation model are proposed. Also, we formalize the IM problem in LBSNs and present an algorithm obtaining an approximation factor of $(1 - 1/e - \epsilon)$ in near-linear expected time. The experimental results show that the algorithm is efficient meanwhile offering strong theoretical guarantees.

Keywords: Location-based social networks · Influence maximization · Two-layer network model

1 Introduction

Social network is a network system formed by social relations among individual members. Social network analysis is based on informatics, mathematics, sociology, management, psychology and other muti-disciplinary fusion theory to study the mechanism of the formation of various social relations, analyze human behavior characteristics and understand the rule of information dissemination.

As online social networks such as Blogs, Facebook, Twitter have been widely used, they have become important platforms for people to make friends, share ideas and issue advertisements. Therefore, the analysis and research on online social networks have developed vigorously. Among them, one of the most popular topics is the issue of IM which asks for a set of k seed nodes in a network to trigger the largest cascade on a propagation model. A great deal of methods on

H. Gao et al. (Eds.): CollaborateCom 2018, LNICST 268, pp. 92–111, 2019.
https://doi.org/10.1007/978-3-030-12981-1_6

the IM have been extensively studied, because it provides a good way to improve marketing, branding and product adoption.

However, they tend to only consider the influence transmission in online social networks and ignore it in the physical life. In order to break this limitation, we study the issue of IM in LBSNs which considers the influence transmission in both online social networks and the physical world. We conduct our research on the datasets of Gowalla and Brightkite where users' online social relationships and location check-ins are collected. Based on the characteristics of these data, a two-layer network model is proposed. Every user has dual identities: the online and offline nodes. The relationship structure of online nodes is stable while the location of offline nodes changes over time. In this model, users can spread information in two ways: sharing through online social networks or talking to people they meet in the physical world. However, the information does not simply travel through online social networks or the physical world separately. It may propagate from online social networks to the physical world or from the physical world to online social networks, which is called cross propagation.

It is very difficult to study the IM problem in this model for two reasons. First, users' locations changing over time makes it seem impossible to study the offline influence propagation. Second, the cross propagation makes the process of influence spread more complicated. For the first problem, we get the offline influence between any two users by analyzing their historical location records, so as to obtain a stable offline relationship structure. For the second problem, we can use the graph theory to combine online and offline relationship graphs into a stable network structure. Thus a complex two-layer network model becomes a traditional network model. It becomes easy to study the issue of IM in LBSNs with the theory about influence propagation in traditional online social networks. Also, it is clear that the IM in LBSNs is an NP-hard problem, since Kempe et al. [2] have proved the NP-hard nature of IM problem for the traditional network model.

Almost all the studies about IM in LBSNs are based on empirical heuristics without any approximation guarantees. In this paper, we present a fast algorithm for the IM problem, obtaining an approximation factor of $(1 - 1/e - \epsilon)$ in near-linear expected time. The experiments on real-world datasets show that in addition to provable guarantees, our algorithm significantly outperforms node selection heuristics. Therefore, our algorithm is efficient meanwhile offering strong theoretical guarantees.

In summary, the main contributions of this paper are as follows:

- Based on the characteristics of two actual datasets named Brightkite and Gowalla, we propose a two-layer network model. This model is a good illustration of people's online and offline interactions. What's more, it connects two interaction modes very well.
- Through analyzing the network model we build and the real-world datasets, we present an influence propagation model describing how information is transmitted in both online social networks and the physical world. Also we propose several methods to convert a complex two-layer propagation model

to a traditional propagation model. Thus it becomes easy to study the issue of IM in LBSNs with the theory about influence propagation in traditional online social networks.
– We present an IM algorithm in LBSNs that runs in near-linear expected time and returns $(1 - 1/e - \epsilon)$-approximate factor under the propagation model we describe.
– The experiments we conduct on the real-world LBSN datasets confirm the effectiveness and efficiency of our proposed algorithm.

The remainder of the paper is organized as follows. The related works are presented in Sect. 2. A two-layer network model and the formulation of the IM in LBSNs are described in Sect. 3. An influence propagation model is proposed and the method converting the two-layer propagation model to a traditional propagation model is introduced in Sect. 4. Section 5 provides the corresponding solution to the IM problem. Section 6 discusses the experiment results, and we conclude the paper in Sect. 7.

2 Related Wroks

The IM problem was first proposed by Domingos and Richardson [1]. They concluded the problem as an algorithm problem and introduced it into the field of social network, which caused many scholars to study. Kempe et al. [2] were the first to formulate influence maximization as a discrete optimization problem and proved that it is an NP-hard problem. What's more, they also proposed two pioneering diffusion models, namely, Independent Cascade (IC) model and Linear Threshold (LT) model, and designed a greedy algorithm with provable approximation guarantee. However, the algorithm has a serious drawback of high time complexity. This celebrated work has motivated a lot of work to improve the greedy algorithm. Leskovec et al. [3] proposed the Cost-Effective Lazy Forward selection (CELF) algorithm which used the submodularity property of the influence maximization objective to greatly reduce the calculation of approximation. Goyal et al. [4] presented the CELF++ algorithm which further improved the calculation speed of the greedy algorithm. Chen et al. [5] also provided a further improvement to the greedy algorithm that still had a guaranteed approximation. At last, Borgs et al. [6] created a theoretical breakthrough in time complexity by using a novel Reverse Influence Sampling (RIS) technique. Their algorithm still returns a $(1-1/e-\epsilon)$-approximate solution with a high probability. Whereafter, many works try to further reduce the time complexity based on the framework Borgs et al. [6] provided. Tang et al. [7,8] and Nguyen et al. [9] used highly sophisticated estimating methods to reduce the number of RIS samples, thus the time complexity was reduced. The Stop and Stare (SSA) and the Dynamic Stop and Stare (D-SSA) devised by Nguyen et al. [10] are optimal algorithms for the IM problem at present. Another direction of research is heuristic algorithms which have a huge advantage of running time but without any approximation ratio guarantee. Basic heuristic algorithms are max degree algorithm, distance

centrality algorithm etc. To summarize, it can be seen from the above research status that the research about IM in traditional online social networks is quite mature.

In recent years, location-based social networks have received wide popularity. A great deal of existing works have been done to study LBSNs from multiple aspects. Cranshaw et al. [10] devised a model for predicting friendship between two users by analyzing their location trails. Pham et al. [11] and Zhang et al. [12] utilized people's movement in the real world to derive the influence among nodes. Li et al. [13] investigated the problem of Geo-Social Influence Spanning Maximization: selecting a certain number of users whose influence can maximally cover a targeted geographical region. Zhou et al. [14] carried out the research about IM in O2O model which means to conduct online promotion and bring maximum number of people to consume in the offline shops. Cai et al. [15] studied an event activation position selection problem in LBSN. Yang et al. [16] explored the IM in online and offline double-layer propagation scheme. But they introduced an empirical heuristic method without any approximation guarantees. Thus, the classical influence maximization in LBSNs is still a domain remaining to be researched.

3 Network Model and Problem Definition

3.1 Network Model

User and location are two main subjects that are closely related to LBSNs. Users visit certain locations in the physical world, leaving a corresponding location history. If we connect these locations over time, we can get the trajectory of each user. Based on these trajectories and online social networks users participate in, a network model in the LBSN can be constructed, as illustrated in Fig. 1.

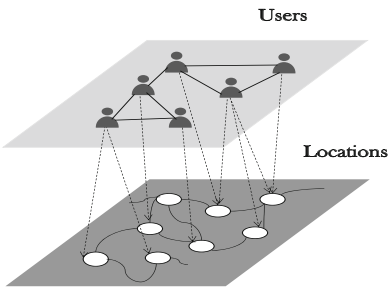


Fig. 1. The network model based on LBSNs

There are two types of nodes: user and location. The edge from the user to the location indicates that the user has accessed the location. The upper layer of the model is a user-user graph which represents the relationships among all

users in an online social network. The lower layer of the model is a location-location graph where a point is a location and an edge indicates that a user has visited both nodes successively. In this paper, we assume that information may be transmitted at a certain probability from user u to v, if u and v are friends in the online social network, or u and v meet in the physical world.

In the datasets of Gowalla and Brightkite, users' online social relationships are clear. It is easy to describe the process of information transmission in online social networks. However, only the data of location check-ins is collected in the datasets and the time attribute values for these data are different, so it is difficult to describe the process of information transmission in the physical world. Even if two users are together, it is quite possible that there are certain time and position differences in their check-ins. For example, user u went on a trip with v. When they arrived at a scenic spot A, u posted a check-in. But twenty minutes later, v released another check-in when they arrived at scenic spot B. Considering this situation, within a specific time interval τ, if the distance between two users' check-in locations is less than a certain value r, we assume that they met each other once. we use $u.x$ and $u.y$ to denote the x-coordinate and y-coordinate for the user u. For two users u, v, the Euclidean distance is donated as

$$d(u, v) = \sqrt{(u.x - v.x)^2 + (u.y - v.y)^2}. \tag{1}$$

3.2 Problem Definition

In the aforementioned model we built, the information can be transmitted at a certain probability from user u to v, if u and v are friends in the online social network, or u and v meet in the physical world. In this situation, the problem can be described as follows:

Definition 1 *(Influence Maximization (IM) in LBSNs). Given a graph $G(V, E)$ which represents the structure of online social network, a dataset C which contains check-ins for all users over a given period of time and integer $k \geq 1$, the influence maximization problem is to find a set of nodes S_k containing at most k nodes that maximizes its influence spread, i.e.,*

$$S_k = \arg \max_{S:|S|=k} E[I(S)] \tag{2}$$

4 Influence Propagation Model

In this part, we first describe the online and offline propagation models respectively. In the online propagation model, we can directly adopt the two most basic and widely-studied diffusion models: Linear Threshold(LT) and Independent Cascade(IC) Models proposed by Kempe et al. [2]. In the offline propagation model, the influence rate between any two nodes can be obtained by utilizing their check-ins history. Therefore, we can construct a stable offline social graph. Then we can further adopt the classical propagation models: LT and IC models. Finally, we combine the online and offline propagation models into a traditional propagation model (Table 1, Fig. 2).

Table 1. Main notations used in the paper

Notation	Description
$G(V, E, B)$	A graph G with a node set V and a directed edge set E. Each directed edge has a weight and the set of weights is B
$b^{on}_{u,v}, b^{off}_{u,v}, b_{u,v}$	The weights between nodes u and v in online network, offline network and synthetic network, respectively
$d(u, v)$	The Euclidean distance between nodes u and v
θ_v	The minimum sum of weights of v's active neighbors in order to activate v
d_u	The degree of node u
S_k	A set of seed nodes containing at most k nodes
$I(S)$	The number of nodes S can activate
r	The maximum distance error when two people meet
τ	The maximum time error when two people meet
$m(u, v, d)$	Whether nodes u and v have met on a date d
C_u	A set of all check-ins of node u in a data set
c_u	A member of C_u, namely, $c_u \in C_u$
$m(c_u, C_v)$	Whether the check-in c_u of node u coincides with at least one check-in from the set C_v of node v
C	A set of check-ins

4.1 Online Social Network Propagation Model

We assume that the information travels via the edges of a graph. Each edge (u, v) has a weight which represents the probability that node u affects node v.

The IC model abstracts the independent interaction among people in social networks. Many simple entities are in line with the characteristics of independent transmission, such as the spread of new information in the online network or the spread of new virus among people. In the IC model, we first activate seed nodes to make them infectious, and then the process unfolds in rounds. When node v first becomes infected in step t, it has a single chance to infect each currently uninfected neighbors in step $t+1$. Until no more nodes are infected, the process finishes. The idea of the LT model is derived from such an assumption: for the unaffected node v, more and more neighbors of v are affected as time goes on. At some point, v may also be affected. In the LT model, we assign a threshold θ_v uniformly at random to each node v from the interval [0, 1]. The threshold represents the minimum sum of weights of v's active neighbors to activate v. In addition, it requires each node v to meet a condition: $\sum_{w:neighbor\ of\ v} b_{w,v} \leq 1$. v is activated when the sum of weights of v's active neighbors is greater than θ_v : $\sum_{w:neighbor\ of\ v} b_{w,v} \geq \theta_v$. The process is over until no more nodes can be infected.

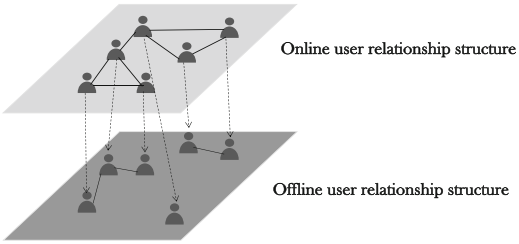


Fig. 2. The influence propagation model based on LBSNs

If we adopt the IC model, we can assign a uniform probability of b to each edge in the graph, such as 0.01 and 0.1. If we adopt the LT model, we can specify a weight of $1/d_u$ for each edge (u, v), where d_u represents the degree of node u. At the same time, we assign a threshold θ_v uniformly at random to each node v from the interval $[0, 1]$. It is worth noting that it is an open question about what model to be adopted and how to allocate weights between nodes. For example, Cai et al. [15] considered the user's interests and the times they received information from neighbors to measure the online influence between nodes.

4.2 Offline Social Network Propagation Model

From the datasets and the network model, we can only get a series of user movements over time at the lower layer of the model. Therefore, it becomes difficult to measure the influence between nodes through these trajectories.

First of all, we might think that the more often two people appear together, the more influence they have. Even when two people are together, they are less likely to submit check-ins at the same time. Also, if they are in a moving state and they submit check-ins one after another at a small time interval, their check-in locations will be somewhat different. Our solution to this problem is as follows: for two users u and v, given a time interval τ and a maximum distance value r, if $d(u, v) < r$, we think that they meet each other.

Next, we introduce several methods to calculate the influence rate between any two nodes.

We measure by days to gather statistic data. Suppose d as a date with a minimum unit of day, we define a function $m(u, v, d)$ which indicates whether node u and v have met on d. If they met on d, $m(u, v, d) = 1$. Conversely, $m(u, v, d) = 0$. It means that in all check-ins of u and v on d, as long as there's a pair of check-ins showing that they met, then $m = 1$, otherwise $m = 0$. Clearly, $m(u, v, d)$ and $m(v, u, d)$ are equal. We represent all the dates in the dataset as a set D and the sum of the dates as $|D|$. $\sum_{d \in D} m(u, v, d)$ is the total number of days on which u and v meet for the dateset D. Thus, $b_{u,v}$(the influence rate between u and v) is $\sum_{d \in D} m(u, v, d)/|D|$. In this way, we can construct a stable undirected graph. Then the IC model can be easily applied above.

We measure by check-ins to gather statistic data. C_u denotes all check-ins of node u in the dataset C. c_u is a member of C_u, namely $c_u \in C_u$. The function $m(c_u, C_v)$is used to indicate whether the check-in c_u of node u coincides with at least one check-in from the set C_v of node v. $\sum_{c_u \in C_u} m(c_u, C_v)$ is the total number of check-ins in which u meets v for the dataset C_u. We denote the sum of all check-ins in C_u as $|C_u|$. Thus, $b_{u,v}$ (the influence rate of u to v) is $\sum_{c_u \in C_u} m(c_u, C_v)/|C_u|$.Clearly, $b_{u,v}$ is usually not the same value as $b_{v,u}$. In this way, we can construct a stable directed graph. Also, the IC model can be easily applied. Alternatively, we can define $b_{v,u}$ as $\sum_{c_u \in C_u} m(c_u, C_v)/|C_u|$. In this case, node u meets this condition : $\sum_{wneighborofu} b_{w,u} \leq 1$ which suggests that we can adopt the LT model. Again, we assign a threshold uniformly at random to each node v from the interval [0, 1].

Finally, it's also an open question about how to calculate weights between nodes and what model to be adopted.

4.3 Single Layer Propagation Model

In the two-layer propagation model, people can spread the information in two ways: sharing information through online social networks or talking to people they meet in the physical world. However, the information does not simply travel in online social networks or the physical world separately. Cross propagation makes the process of influence propagation more complicated (Fig. 3).

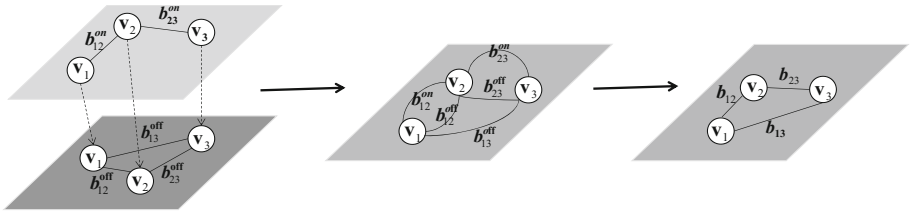


Fig. 3. An example of converting two-layer graph to single-layer graph

Inspired by Yang et al. [16], the two-layer propagation model can be compressed into a single-layer propagation model. For nodes u and v, they can share information by communicating online or talking offline. Therefore, the information can not be spread between u and v, if and only if they don't communicate online, meanwhile, they don't talk offline. Hence, in the resulting single layer model, the weight of u to v is as follows:

$$b_{u,v} = 1 - (1 - b_{u,v}^{on})(1 - b_{u,v}^{off}) \tag{3}$$

We can apply IC model directly in the resulting single layer propagation model. However, we must standardize the incident edges of each node, so that the sum of weights of incident edges is less than or equal to 1, if we want to apply LT model.

5 Our Solution

A single-layer graph has been constructed from the double-layer graph in Sect. 4. Hence, it becomes easy to study the issue of IM in LBSNs with the theory about influence propagation in traditional online social networks. In this section, we formally state the IM problem in traditional online social networks, and present an overview of the RIS framework which is a theoretical breakthrough way to solve the IM problem. Subsequently, our solution to the IM problem will be introduced followed by a summary of approximation and complexity.

5.1 IM Definition in Traditional Online Social Network

Let G be a graph with a node set V and a directed edge set E. Each directed edge has a propagation probability $b(e) \in [0, 1]$. We refer to S as a seed set, and $I(S)$ as the number of nodes that are infected in the end, namely the spread of S. Given the propagation models constructed previously, we formally define the IM problem as follows:

Definition 2 *(Influence Maximization in Traditional Online Social Network).* *Given a graph $G = (V, E, B)$, a constant $k \geq 1$ and a propagation model, the influence maximization problem is to find a seed set S_k of k nodes at most that maximizes its influence spread, i.e.,*

$$S_k = \arg \max_{S:|S|=k} E[I(S)]$$

5.2 Summary of the RIS Framework

The inefficiency of traditional greedy methods has long been a drawback for IM problem. By using the RIS technique, which is the foundation of the state-of-the art method, the time complexity of algorithms is greatly reduced. The RIS is based on the concept of Random Reverse Reachable set (Random RR set), which is defined below.

Definition 3 *(Random Reverse Reachable set (Random RR set)).* *Given a graph $G = (V, E, B)$ and a random node $v \in V$, g is a random graph obtained by removing each edge e from G with $1 - b(e)$ probability. The Random RR set is the set of nodes in g that can reach v.*

Borgs et al. [6] proved the following Theorem 1 through mathematical analysis. Theorem 1 provides the theoretical basis for solving the IM problem by the RIS method.

Theorem 1. *Given $G = (V, E, B)$ and a random RR set R from G. For a seed set $S \subset V$,*

$$I(S) = nPr[ScoversR]. \tag{4}$$

Theorem 1 implies that we can estimate $E(I(S))$ by estimating the probability of the event S covers R. An example is used to illustrate the process of this method in the Fig. 4. Four random RR sets are generated with sources v_1, v_2, v_3 and v_4, respectively. v_3 appears most frequently among these sets. From the intuitive observation, v_3 is also the most influential node in practice. Based on the above theorem, Borgs et al. [6] proposed a two-step method for IM problem:

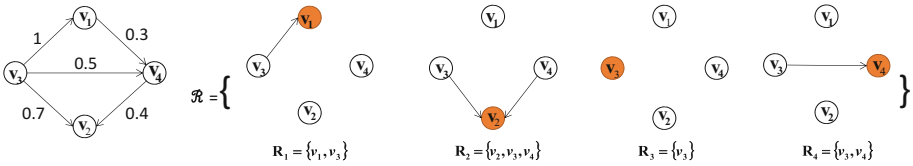


Fig. 4. An example of generating random RR sets for a simple graph.

1. Generate a sufficiently large collection R of random RR sets from G.
2. Use the greedy algorithm of maximum coverage problem to find the seed set with maximum coverage to R.

5.3 SSA and D-SSA Algorithm

So far, the optimal algorithm with approximation guarantee for IM problem is SSA and D-SSA [10], both of which exploit the zero-one estimation theorem [17] to gauge the required number of RR sets. SSA keeps generating RR sets until the generated seed set S is a good approximate solution. The general process of SSA is described as follows:

1. Generate an initial set R of random RR sets with a certain number.
2. Use the greedy algorithm of maximum coverage problem to find a size-k seed set S from R.
3. Generate another set of random RR sets to test whether S is a better approximate solution.
4. If S is a better approximation, terminate the procedure and return S; Otherwise, double the number of random RR sets, and goto Step 2.

In the process described above, the SSA generates two independent sets of RR sets: One is to find the seed set and the other is for estimating the influence of the seed set. It has three parameters ϵ_1, ϵ_2 and ϵ_3. These parameters decide the approximation errors allowed in step 3 and the number of random RR sets generated in each iteration of SSA. They are fixed to $\epsilon_1 = 1/6\delta$, $\epsilon_1 = 1/2\delta$, $\epsilon_3 = \delta/(4(1 - 1/e))$. However, Nguyen et al. [10] propose to vary them in different iterations to reduce the total number of random RR sets generated. Thus they propose the DSSA algorithm. The general process of DSSA is described as follows:

1. Generate an initial set R of random RR sets with a certain number.
2. Use the greedy algorithm for maximum coverage on R to find a size-k seed set S, along with the number of RR sets in R that overlap S.
3. Generate another set of random RR sets to derive an estimation of S' expected spread and to determine the value of ϵ.
4. Evaluate whether S is a good approximation solution based on the number of RR sets in R that overlap S, the estimation of S' expected spread and ϵ.
5. If S is a better approximation, terminate the procedure and return S; Otherwise, double the number of random RR sets, and goto Step 2.

Nguyen et al. [10] show that SSA and DSSA return a $(1 - 1/e - \epsilon)$-approximate solution with at least $(1 - \delta)$ probability. The details of those algorithms and the analysis of approximation complexity are given in [10].

6 Experiments

6.1 Experimental Settings

All the experiments are run on a PC machine with Intel Core i7 4.00 GHz processor, 16.0G RAM and 64 bit Linux operating system. We carry experiments under IC model on the following datasets and algorithms.

Datasets. We adopt two widely-used LBSNs, Brightkite and Gowalla's datasets in our research. All two datasets are collected from Stanford Network Analysis Project (SNAP) by Stanford University [18,19]. Brightkite was once a location-based social networking service provider where users shared their locations by checking-in. The friendship network was collected using their public API, and consists of 58228 nodes and 214078 undirected edges. we treat it as a directed graph by converting an undirected edge into two opposite directed edges. A total of 4491143 checkins of these users over the period of Apr. 2008 - Oct. 2010 has been collected. Gowalla is a location-based social networking website where users share their locations by checking-in. The friendship network is undirected and was collected using their public API, and consists of 196591 nodes and 950327 edges. we also treat it as a directed graph by converting an undirected edge into two opposite directed edges. A total of 6442890 check-ins of these users over the period of Feb. 2009 - Oct. 2010 has been collected.

Parameter Settings. For simplicity, we assign the same weight b to each edge in the online network model. In the offline network model, we give a distance r and a time interval τ to represent the maximum distance error and the maximum time error respectively when two users meet. The location is measured in degrees so that we use degrees for r. k is the number of seed nodes, varying from 20 to 100 in our experiment. The specific choices of the four parameters b, r, τ and k are shown in Table 2.

Table 2. Experiment parameters

Parameter	Values				
Online influence rate b	0.1	0.01	0.001	0.0001	
Distance error r	0.001	0.0001	0.00001		
Time error τ	0.5 h	1 h	12 h	24 h	
Number of seed nodes k	20	40	60	80	100

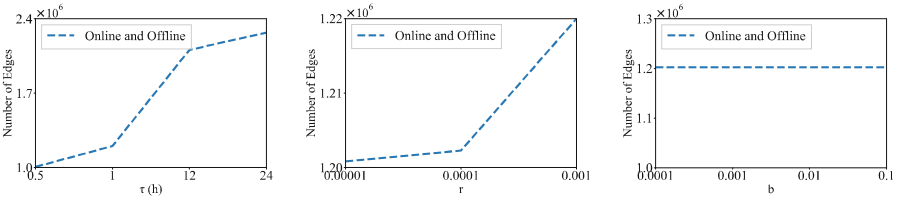
Algorithm Compared. On IM experiments, we compare DSSA with three algorithms (TIM [7], IMM [8] and SSA [9]), which are RIS-based algorithms that provide $(1 - 1/e - \delta)$-approximation guarantee.

6.2 Modeling Process

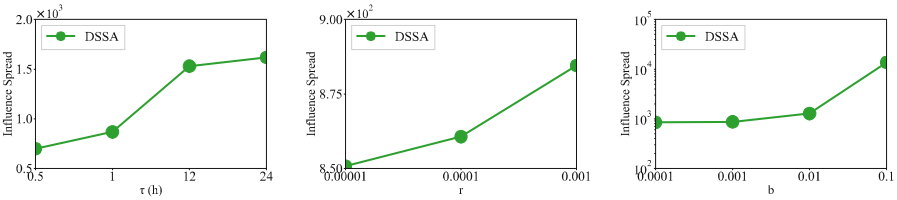
For simplicity, we assign the same weight b to each edge in the online network model. In the offline network model, two methods are introduced to calculate the influence rate between nodes. One is to measure by days for judging whether two users met. The influence rate of two users is obtained by dividing the number of days that they met by the number of days in the dataset. We can get an undirected graph in this way. The other is to measure by check-ins for judging whether two users met. The influence rate of one user u to another v is obtained by dividing the number of check-ins that u met v by the number of u' check-ins in the data set. We can get a directed graph in this way. Finally, we compress the two-layer propagation model into a single-layer propagation model by using Eq. (3). The detailed process is shown in Algorithms 1 and 2. In both algorithms, we represent the graph in the form of an adjacency list.

In the process of modeling, there are three key parameters that affect the comprehensive influence rate of two nodes: uniform influence rate b between nodes in online model, the maximum distance error r and the maximum time error τ when two users meet. In order to compare the effects of different parameters on modeling, we compare the number of edges on the different resulting single-layer graphs and use the DSSA to measure the influence spread by fixing the number $k = 20$ of seed nodes. Figures 5 and 6 show the experiment results of measuring by days on Brightkite and Gowalla during the modeling process. Figures 7 and 8 show the results of measuring by check-ins on Brightkite and Gowalla. In Figs. 5(a), 6(a), 7(a) and 8(a), we change the maximum time error τ from 0.5 h to 24 h. It is obvious that the larger τ is, the more edges it has in the resulting graphs. Also, the larger τ is, the more influence it can achieve in the resulting graphs, which can be learned from Figs. 5(d), 6(d), 7(d) and 8(d). By Eq. (3), the edges in the resulting graph are the union of edges in the online graph and the offline graph. The larger τ is, the more edges we can get in the offline graph. Furthermore, the number of edges in the resulting graph increases.

In Figs. 5(b), 6(b), 7(b) and 8(b), the maximum distance error r is the parameter we want to evaluate, which ranges from 0.00001 to 0.001. A degree of longi-

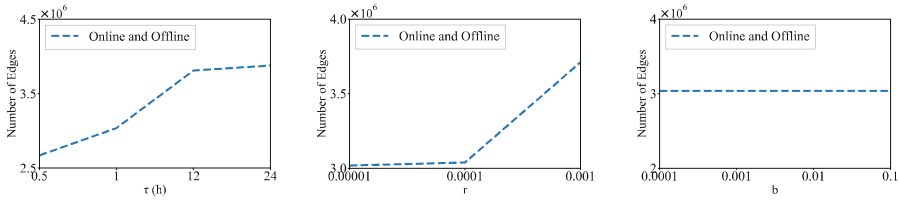


(a) $b = 0.001, r = 0.0001$ (b) $b = 0.001, \tau = 1h$ (c) $r = 0.0001, \tau = 1h$

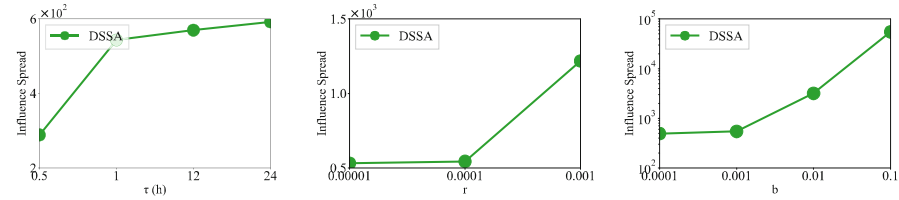


(d) $b = 0.001, r = 0.0001$ (e) $b = 0.001, \tau = 1h$ (f) $r = 0.0001, \tau = 1h$

Fig. 5. The results of measuring by days on Brightkite



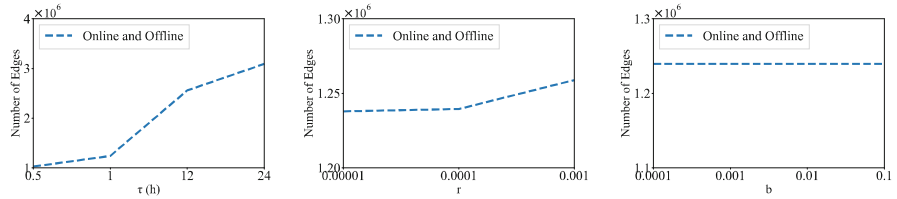
(a) $b = 0.001, r = 0.0001$ (b) $b = 0.001, \tau = 1h$ (c) $r = 0.0001, \tau = 1h$



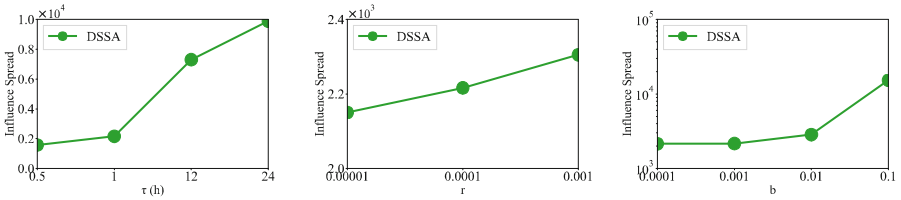
(d) $b = 0.001, r = 0.0001$ (e) $b = 0.001, \tau = 1h$ (f) $r = 0.0001, \tau = 1h$

Fig. 6. The results of measuring by days on Gowalla

tude or latitude is at most 111.11 km. Thus, 0.001 is at most 111.11 meters, and 0.00001 is at most 1.1111 m. The larger r is, the more edges it has in resulting graphs. Also, the larger r is, the more influence it can achieve in resulting graphs, which can be learned from Figs. 5(e), 6(e), 7(e) and 8(e). For the same reason as the time error parameter, the larger r is, the more edges we can get in the offline graph. Furthermore, the number of edges in the resulting graph increases.

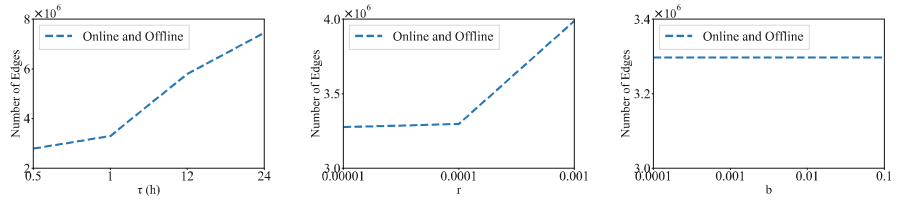


(a) $b = 0.001, r = 0.0001$ (b) $b = 0.001, \tau = 1h$ (c) $r = 0.0001, \tau = 1h$

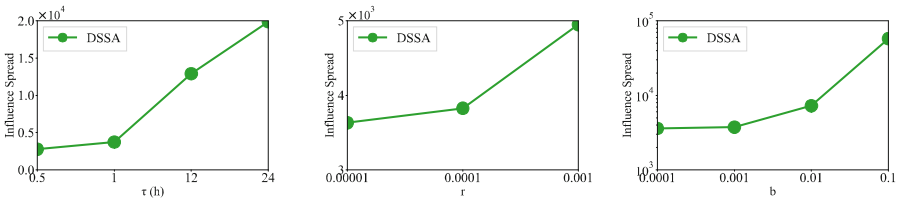


(d) $b = 0.001, r = 0.0001$ (e) $b = 0.001, \tau = 1h$ (f) $r = 0.0001, \tau = 1h$

Fig. 7. The results of measuring by check-ins on Brightkite



(a) $b = 0.001, r = 0.0001$ (b) $b = 0.001, \tau = 1h$ (c) $r = 0.0001, \tau = 1h$



(d) $b = 0.001, r = 0.0001$ (e) $b = 0.001, \tau = 1h$ (f) $r = 0.0001, t = 1h$

Fig. 8. The results of measuring by check-ins on Gowalla

In Figs. 5(c), 6(c), 7(c) and 8(c), we change the uniform weight b in online social network from 0.0001 to 0.1. The number of edges in the resulting graph remains the same as b increases. But the larger b is, the more influence it can achieve in the resulting graphs in Figs. 5(f), 6(f), 7(f) and 8(f). Different weights b can only change the influence rate between nodes in online graphs, but not change the number of edges in online and offline graphs. Thus the number of

Algorithm 1.

Input: an online graph $G^{on} = (V, E^{on}, B^{on})$; a check-ins dataset C; a maximum distance error r; a maximum time error τ;
Output: an compressed single-layer graph $G(V, E, B)$

for $i = 0 \rightarrow |V|$ **do**
  for $j = i + 1 \rightarrow |V|$ **do**
    $b_{i,j}^{off} \leftarrow MeetDayCount(C, i, j)/|D|$
    $b_{j,i}^{off} \leftarrow b_{i,j}^{off}$
    $b_{i,j} \leftarrow 1 - (1 - b_{i,j}^{on})(1 - b_{i,j}^{off})$
    $b_{j,i} \leftarrow b_{i,j}$
    put $b_{i,j}$ and $b_{j,i}$ into B
    if $b_{i,j} \neq 0$ **then**
      put $e_{i,j}$ and $e_{j,i}$ into E
    end if
  end for
end for
return $G(V, E, B)$

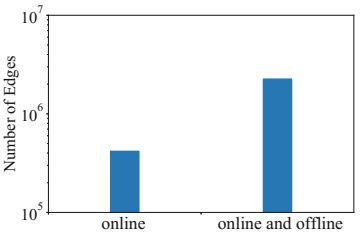
edges in the resulting graph remains the same. However, the increase of the online influence can increase the comprehensive influence spread in the resulting graph.

6.3 Comparison Between Online Influence Spread and Online-Offline Influence Spread

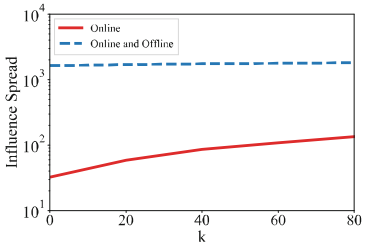
In order to evaluate the effect of influence spread in offline physical world, we compare the influence spread in online social network with the influence spread in the online-offline social networks. It can be seen from Figs. 9 and 10 that the number of edges and influence spread increase exponentially due to the effect of oral communication in the physical world. These comparative data show that our research topic is of great practical significance.

6.4 Algorithms Comparison

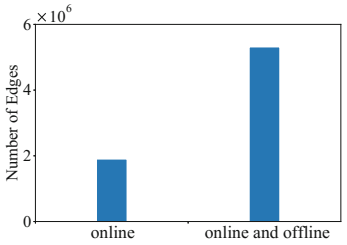
To show the superior performance of the DSSA we used, we compare it with three other RIS-based algorithms under IC model. Firstly, we compare the quality of the solution returned by all the algorithms. From Figs. 11(b), (d), 12(b) and 11(d), all the algorithms return comparable seed set quality without significant difference. Secondly, we examine the performance in terms of running time of all the algorithms. From Figs. 11(c), (e), 12(c) and (e), DSSA significantly outperforms the other tested algorithms by a huge margin. All of this is due to the fact that the TIM and IMM generate too many RR sets. DSSA overcomes this weakness and commits up to an order of magnitudes speedup.



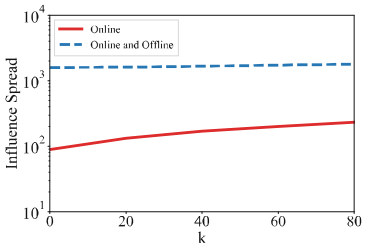
(a) $b = 0.001, r = 0.001, \tau = 24h$



(b) $b = 0.001, r = 0.001, \tau = 24h$

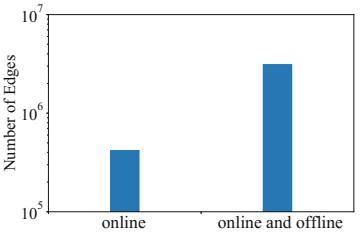


(c) $b = 0.001, r = 0.001, \tau = 24h$

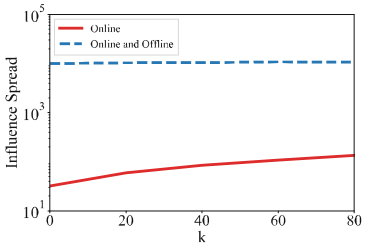


(d) $b = 0.001, r = 0.001, \tau = 24h$

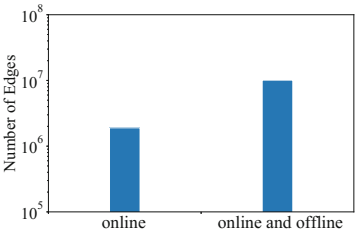
Fig. 9. The results of measuring by days on Brightkite and Gowalla



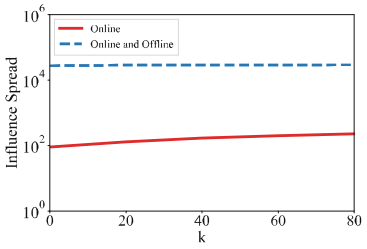
(a) $b = 0.001, r = 0.001, \tau = 24h$



(b) $b = 0.001, r = 0.001, \tau = 24h$



(c) $b = 0.001, r = 0.001, \tau = 24h$



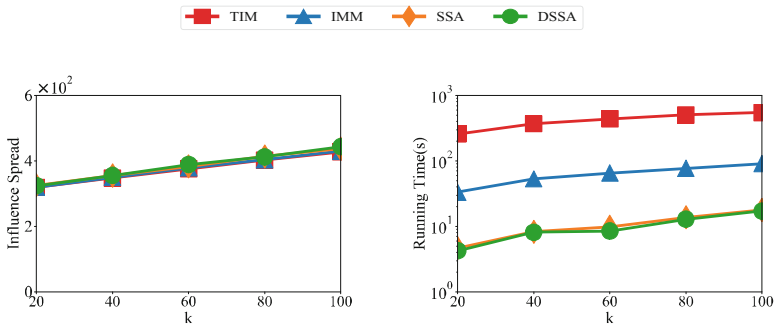
(d) $b = 0.001, r = 0.001, \tau = 24h$

Fig. 10. The results of measuring by check-ins on Brightkite and Gowalla

Algorithm 2.

Input: an online graph $G^{on} = (V, E^{on}, B^{on})$; a check-ins dataset C; a maximum distance error r; a maximum time error τ;
Output: an compressed single-layer graph $G(V, E, B)$

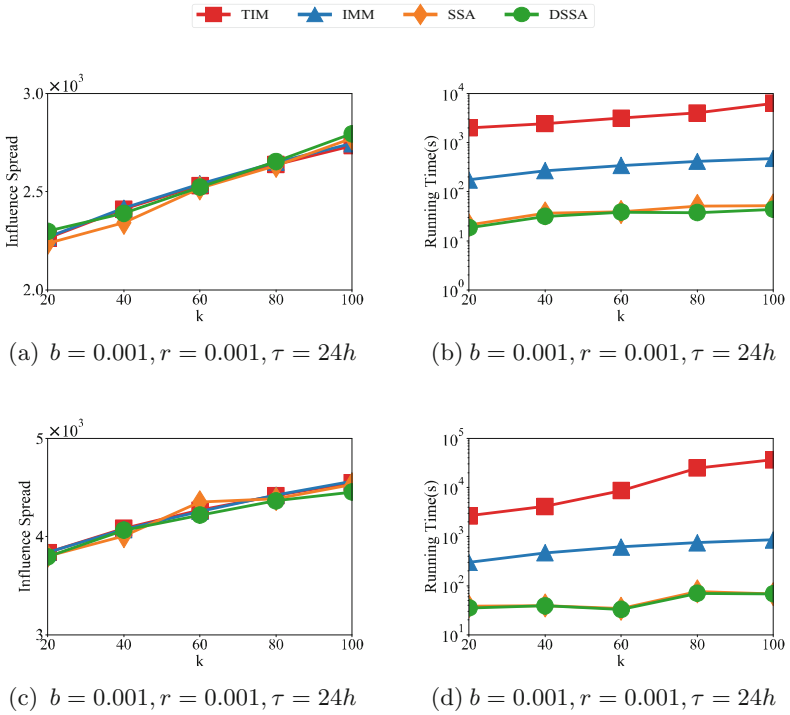
for $i = 0 \rightarrow |V|$ **do**
  for $j = 1 \rightarrow |V|$ **do**
    if $i = j$ **then**
      continue;
    end if
    $b_{i,j}^{off} \leftarrow MeetCheckinCount(C, i, j)/|C_i|$
    $b_{i,j} \leftarrow 1 - (1 - b_{i,j}^{on})(1 - b_{i,j}^{off})$
    put $b_{i,j}$ into W
    if $b_{i,j} \neq 0$ **then**
      put $e_{i,j}$ into E
    end if
  end for
end for
return $G(V, E, B)$



(a) $b = 0.001, r = 0.001, \tau = 24h$ (b) $b = 0.001, r = 0.001, \tau = 24h$

(c) $b = 0.001, r = 0.001, \tau = 24h$ (d) $b = 0.001, r = 0.001, \tau = 24h$

Fig. 11. The results of measuring by days on Brightkite and Gowalla



(a) $b = 0.001, r = 0.001, \tau = 24h$ (b) $b = 0.001, r = 0.001, \tau = 24h$

(c) $b = 0.001, r = 0.001, \tau = 24h$ (d) $b = 0.001, r = 0.001, \tau = 24h$

Fig. 12. The results of measuring by check-ins on Brightkite and Gowalla

7 Conclusion

In this paper, we aim to explore the influence maximization in location-based social networks. A two-layer network model and an information propagation model are proposed. Those models are motivated by the fact that influence propagates in both online social networks and the physical world. Also, we formalize the influence maximization problem in LBSNs and present an algorithm obtaining an approximation factor of $(1 - 1/e - \epsilon)$ in near-linear expected time. The experimental results show that the algorithm is efficient meanwhile offering strong theoretical guarantees. Furthermore, we compared the influence spread in the online social network with the influence spread in the online-offline social network and proved the practical significance of our research topic.

Acknowledgements. This work is partially supported by National Natural Science Foundation of China (NSFC) under Grant No.61772491, No.61472460, and Natural Science Foundation of Jiangsu Province under Grant No. BK20161256, and Anhui Initiative in Quantum Information Technologies AHY150300. Kai Han is the corresponding author.

References

1. Domingos, P., Richardson, M.: Mining the network value of customers. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 57–66. ACM, August 2001
2. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 137–146. ACM, August 2003
3. Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., VanBriesen, J., Glance, N.: Cost-effective outbreak detection in networks. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 420–429. ACM, August 2007
4. Goyal, A., Lu, W., Lakshmanan, L.V.: CELF++: optimizing the greedy algorithm for influence maximization in social networks. In: Proceedings of the 20th International Conference Companion on World Wide Web, pp. 47–48. ACM, March 2011
5. Chen, W., Wang, Y., Yang, S.: Efficient influence maximization in social networks. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 199–208. ACM, June 2009
6. Borgs, C., Brautbar, M., Chayes, J., Lucier, B.: Maximizing social influence in nearly optimal time. In: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 946–957. Society for Industrial and Applied Mathematics, January 2014
7. Tang, Y., Xiao, X., Shi, Y.: Influence maximization: near-optimal time complexity meets practical efficiency. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, pp. 75–86. ACM, June 2014
8. Tang, Y., Shi, Y., Xiao, X.: Influence maximization in near-linear time: a martingale approach. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, pp. 1539–1554. ACM, May 2015
9. Nguyen, H.T., Thai, M.T., Dinh, T.N.: Stop-and-stare: optimal sampling algorithms for viral marketing in billion-scale networks. In: Proceedings of the 2016 International Conference on Management of Data, pp. 695–710. ACM, June 2016
10. Cranshaw, J., Toch, E., Hong, J., Kittur, A., Sadeh, N.: Bridging the gap between physical location and online social networks. In: Proceedings of the 12th ACM International Conference on Ubiquitous Computing, pp. 119–128. ACM, September 2010
11. Pham, H., Shahabi, C.: Spatial influence-measuring followship in the real world. In: 2016 IEEE 32nd International Conference on Data Engineering (ICDE), pp. 529–540. IEEE, May 2016
12. Zhang, C., Shou, L., Chen, K., Chen, G., Bei, Y.: Evaluating geo-social influence in location-based social networks. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, pp. 1442–1451. ACM, October 2012
13. Li, J., Sellis, T., Culpepper, J.S., He, Z., Liu, C., Wang, J.: Geo-social influence spanning maximization. IEEE Trans. Knowl. Data Eng. **29**(8), 1653–1666 (2017)
14. Zhou, T., Cao, J., Liu, B., Xu, S., Zhu, Z., Luo, J.: Location-based influence maximization in social networks. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 1211–1220. ACM, October 2015

15. Cai, J.L.Z., Yan, M., Li, Y.: Using crowdsourced data in location-based social networks to explore influence maximization. In: IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, pp. 1–9. IEEE, April 2016
16. Yang, Y., Xu, Y., Wang, E., Lou, K., Luan, D.: Exploring influence maximization in online and offline double-layer propagation scheme. Inf. Sci. **450**, 182–199 (2018)
17. Dagum, P., Karp, R., Luby, M., Ross, S.: An optimal algorithm for Monte Carlo estimation. SIAM J. Comput. **29**(5), 1484–1496 (2000)
18. Brightkite Database Information from Stanford Network Analysis Project(SNAP). http://snap.stanford.edu/data/loc-Brightkite.html
19. Gowalla Database Information from Stanford Network Analysis Project(SNAP). http://snap.stanford.edu/data/loc-Gowalla.html