Soc. Netw. Anal. Min. (2014) 4:236
DOI 10.1007/s13278-014-0236-y

ORIGINAL ARTICLE

Link injection for boosting information spread in social networks

Stefanos Antaris · Dimitrios Rafailidis · Alexandros Nanopoulos

Received: 13 March 2014 / Revised: 13 September 2014 / Accepted: 1 November 2014 / Published online: 15 November 2014

Abstract Social media have become popular platforms for spreading information. Several applications, such as 'viral marketing', pause the requirement for attaining large-scale information spread in the form of word-of-mouth that reaches a large number of users. In this paper, we propose a novel method that predicts new social links that can be inserted among existing users of a social network, aiming directly at boosting information spread and increasing its reach. We refer to this task as 'link injection', because unlike most existing people-recommendation methods, it focuses directly on information spread. A set of candidate links for injection is first predicted in a collaborative-filtering fashion, which generates personalized candidate connections. We select among the candidate links a constrained number that will be finally injected based on a novel application of a score that measures the importance of nodes in a social graph, following the strategy of injecting links adjacent to the most important nodes. The proposed method is suitable for real-world applications, because the injected links manage to substantially increase the reach of information spread by controlling at the same time the number of injected links not to affect the user experience. We evaluate the performance of our proposed methodology by examining several real data sets from social networks under several distinct factors. The experimentation demonstrates the effectiveness of our proposed method, which increases the spread by more than a twofold factor by injecting as few as half of the existing number of links.

Keywords Information spread · Social networks · Viral marketing · Link injection

1 Introduction

The spread of information in social networks enables the users not only to communicate with their friends but also to exchange their ideas and share their opinions. This can be attained via information cascades, where information (e.g., photos, text posts, hyperlinks, etc.) can potentially reach a large portion of the network. It has been observed, however, that the spread of information (e.g., discussed topics) tend to be restrained within a closed group of friends and it hardly propagates beyond the community that initiated the discussion (Chaoji et al. 2012). Consequently, users may never get to know about a topic, if none of their friends mention it.

In a small number of cases, though, the world-of-mouth (WOM) in social networks can affect the diffusion of some pieces of information and its ability to receive viral dimensions (Li and Shiu 2012). This is exploited in applications such as 'viral marketing', which use social networks to spread marketing messages that inform the public about products, services, or brands (Dinev et al. 2008). Viral marketing has been described as particularly effective due to the higher likelihood of users to accept information posted by friends, compared to online advertisement (Kim and Srivastava 2007). The maximization of information spread in viral marketing is attained through the selection of few but influential users that initiate the spread and manage to propagate information in a viral fashion (Kempe et al. 2003).

S. Antaris (✉) · D. Rafailidis
Aristotle University of Thessaloniki, Thessaloníki, Greece
e-mail: santaris@csd.auth.gr

A. Nanopoulos
Katholische Universität Eichstätt-Ingolstadt, Eichstätt, Germany

1.1 Motivation

Although the selection of few (to keep the cost low) but influential seeds can help viral marketing, this approach is based on the premise that influential users are willing to initiate a viral campaign. Such a premise often does not hold, since influential users (called also "hubs") are not easy to become engaged (Hinz et al. 2011). Moreover, as aforementioned, the spread of information initiated by less influential users tends to remain constrained within a small community, unless this community contains influential users that can propagate it further. In particular, users with common interests usually participate in the same social groups, creating in this way communities. The nodes within such communities tend to be more densely inter-connected with each other than with the rest of the social graph that represents the structure of a social network (Papadopoulos et al. 2012; David and Jon 2010). As such, these nodes are in a way isolated from other communities, a fact that hinders the spread of information across communities.

This problem poses the requirement for alternative approaches that can help information of adequate interest to exceed the boundaries of small communities and reach a larger portion of a network. Such an approach that is being followed by social networks, is to increase the number of connections among users, aiming eventually at "bridging" isolated communities and, thus, permitting the spread of information between them. A standard method to increase the number of connections is the recommendation of Friend-of-Friend (FoF), where users receive recommendations to connect to other users according to the number of common friends. However, FoF recommendation aims primarily at increasing locally the 'cluster coefficient' by closing 'triangles' between users (David and Jon 2010) and is not designed directly to help in improving the information spread. What is required instead for the latter purpose is to focus directly on connecting influential users with other user of mutual interest, to increase the chances of information to spread beyond the boundaries of smaller communities.

1.2 Contribution and outline

In this paper, we capitalize on the aforementioned principle and propose a novel method for performing link injection in social networks in a way that focuses on increasing the connections adjacent to influential users. We identify influential users by following the concept of epidemic spread (-Tong et al. 2010), which analyzes the structure of the network and selects the top-*k* most important nodes which will affect at most the spread of information. The injected links are applied on the top-*k* most important nodes, to transform the social network to susceptible in large-scale spread of information that has the capacity to become viral. A set of candidate injected links are discovered using a collaborative-filtering algorithm which factorizes the adjacency matrix of the graph and generates the connectivity weights among its nodes. A predefined upper bound of the number of injected links is finally selected according to their adjacency to the identified top-*k* nodes. The use of an upper bound in the number of injected links supports the use of the proposed method in real-world applications, where the number of injected links is kept controlled not to affect the experience of users that are not willing to receive a large number of recommendations to connect to other users.

Extensive experimentation has been conducted on five real data sets from social networks to evaluate the performance of our proposed methodology. Based on the experimentation results, the link injection mechanism can boost the spread of information by more than a twofold factor using a rather limited number of injected links that does not exceed the half of the number of existing links.

The rest of this manuscript is organized as follows: Sect. 2 summarizes the related work. Section 3 presents the problem formulation. The description of the proposed methodology is presented in Sect. 4. The experimental procedure of our methodology is described in Sect. 5 and the results of the experimental evaluation are presented in Sect. 6. Finally, the manuscript is concluded in Sect. 7.

2 Related work

The diffusion of information, ideas, or innovation, and the propagation of influence over online social networks have attracted extensive researched interest in recent years (Guille et al. 2013; Jackson 2011; Bonchi et al. 2011). Several works have focused on the problem of influence maximization for viral-marketing applications, which aims at selecting influential users that will act as seeds to initiate a diffusion process (David and Jon 2010; Kiss and Bichler 2008; Kempe et al. 2003). Several seed-selection strategies have been proposed to select the most influential users (Kiss and Bichler 2008; David and Jon 2010; Goyal et al. 2013; Hinz et al. 2011).

Based on a large, real-user study, Hinz et al. (2011) have shown that the selection of influential users according to their centrality in the social network results in a more effective diffusion of information than selecting non-central users. Furthermore, a number of additional approaches (e.g., Kempe et al. 2003; Goyal et al. 2013) have proposed to use influence factors between users of a social network. These approaches are based on the fact that the user tendency to accommodate an opinion increases monotonically if his neighbors become active (Kempe et al. 2003). Moreover, the probability for a user to become activated may change based on the influence factor of the neighbor who is trying to affect the user (Kempe et al. 2005). For this purpose, two fundamental diffusion models, the Independent Cascade and the Linear Threshold, have been proposed to model diffusion processes (Kempe et al. 2003). Both of them require former knowledge of the influence factors between the users of the graph.

Identifying, though, the exact influence factors between users is NP-hard (Kempe et al. 2005; Chen et al. 2010). Several scalable (Chen et al. 2010; Leskovec et al. 2007) and heuristic (Kimura and Saito 2006; Chen et al. 2009; Narayanam and Narahari 2011) algorithms have been proposed to reduce the complexity. Moreover, machine learning algorithms (Anagnostopoulos et al. 2008; Tang et al. 2009; Saito et al. 2010; Goyal et al. 2010) are used to learn the parameters of the underlying influence diffusion model based on data from past cascades. Utilizing further user- and social-based features of the network can result in improved prediction of the diffusion rather than other heuristic measures (Bhatt et al. 2010). Nevertheless, all these approaches can be used effectively in the case when data from previous cascades are relevant to the application of interest. Otherwise, the estimated influence factors will not be representative, e.g., in case of a viral-marketing campaign about a new product type. Another problem is that several forms of recorded user actions during previous cascades may comprise private data that are often not allowed to be exploited.

Compared to the aforementioned approaches about influence maximization based on seed selection, our work is complementary, because they focus on the selection of the most influential seeds among the existing users in the network, whereas our focus is on the injection of *new* links between users. In this aspect, our approach is more related to users-recommendation algorithms, which consider the preferences of each user and their social ties to predict new connections among users that share social connections, i.e., FoF scheme, or they have common interests (Chen et al. 2009; Guy et al. 2009; Hannon et al. 2010). However, the goal of such existing approaches is not to recommended connections that aim at increasing the spread of information.

An alternative approach of user recommendation that places emphasis on information spread is proposed in Chaoji et al. (2012). This work is closely related to our approach, since it aims at recommending connections to boost information spread in social networks. However, it is based on knowledge about users' profiles, interests, and the content being shared among users over a fixed time period. Such knowledge may not be available or not allowed to be used, due to privacy issues. Moreover, the work of Chaoji et al. (2012) sets the total number of recommended users as the product of the number of existing links among all users, which may be prohibitively large.

In contrast to Chaoji et al. (2012), our approach does not employ knowledge about users' preferences and is based only on the structure of the social network. Furthermore, our approach controls the number of new links which will be inserted. More importantly, the total number of recommended users is a fraction of the number of existing links.

3 Problem formulation

In this section, we describe the investigated problem. In Table 1, we present the main symbols we use throughout the paper.

Given the graph $G = (N, L)$ that represents the structure of a social network, with N being the set of nodes and L being the set of links (pairwise social relationship), we propose a novel link-injection algorithm to create a more susceptible graph G' to boost the viral process. We model the graph G using an adjacency matrix $A \in \mathbb{R}^{|N| \times |N|}$. The values of the matrix A are initially 1 or 0, denoting the existence or the absence of a link between two nodes, respectively. In social networking graphs, the adjacency matrix is usually very sparse, since $|L| \ll |N|^2$.

The link-injection problem can be formulated as the generation of a graph $G' = (N, L \cup L')$ that has the same set of nodes as G and a set L' of additional links. The adjacency matrix of G' is denoted as A'. First, link injection is performed by identifying the top-k set $S \subset N$ of nodes (assuming that $|S| \ll |N|$) with the highest *diffusion coverage*. The diffusion coverage for each node $j \in N$ is denoted as $\Delta\lambda(j)$ and represents how important the node j is for the flow of information that can spread over the network G (the definition of this scoring is given in Sect. 4). The set of injected links L' is determined based on the principle that they will connect a subset of the top-k nodes in S with a subset of the rest nodes in N. We should mention that an injected link can also be established between two nodes in S. We follow this principle, because injected links that are adjacent to S nodes (i.e., at least one of the two nodes of the injected link belongs to S) can contribute more to information spread, since nodes in S have been selected for this purpose based on their $\Delta\lambda(S)$ scores.

From all possible $k \times (|N| - k)$ candidate links that can be injected, we select from L' only a small fraction m, where m is the maximum number (upper bound) of the potential injected links and is calculated as a factor of the number of the existing links in L_S, i.e. the links that are adjacent to S nodes. In this way, the total number of injected links can be constrained and, consequently, the injected links will not overwhelm the existing connections of S, not affecting thus the user experience. The selection of the candidate links that are going to be selected is determined according to their appearance likelihood. Specifically, a set of link recommendations is provided to users

Table 1 List of symbols

Symbol	Definition and description
G, G'	Social network graphs
A, A'	Adjacency matrices
$A(i,j)$	The element of the ith row and jth column of the matrix A
N	Set of nodes in graph G
L	Set of edges in graph G
S	Set of top-k nodes with the highest diffusion coverage
λ	First eigen-value of A
$\mathbf{u}$	First eigen-vector of A
$\Delta\lambda(i)$	Eigen-drop of the ith node
$\Delta\lambda(S)$	Eigen-drop of the top-k nodes of S
m	Upper bound of injected links

of sets S and N for adding new links among them, until m new (injected) connections have been created. To increase the accuracy of such recommendations, we have to predict the links that are more likely to be created. This is achieved using a link-prediction algorithm (Liben-Nowell and Kleinberg 2003; Schifanella et al. 2010). In our method we use a simple but powerful collaborative-filtering approach based on (non-negative) matrix factorization (Berry et al. 2003). We have to note that our experimental evaluation did not involve a user study where such recommendations could be provided to real users. Therefore, we simplified the experimental evaluation based on the premise that the generated links can be directly injected, which is justified by the high recommendation accuracy of the used link-prediction method.

4 Proposed method

In this section, we describe the three steps of our proposed method: (1) first, we define the *diffusion-coverage* score to measure the impact of each node of the social graph G on information spread and select in S the set of top-k nodes with the highest score. (2) In the second step, a set of candidate links is proposed that can be injected to connect nodes from S with the nodes of N, based on (non-negative) factorization of the adjacency matrix of graph G. (3) Finally, the injected links are selected from the candidates as the ones that have the highest likelihood.

4.1 Diffusion-coverage score

Initially, the goal is to identify the expected impact that each node of a social graph G has on the spread of information over G. We opt for measuring this impact based on how robust the graph G is after removing each node. Several approaches have been proposed for this purpose (Boldi et al. 2013; Borbora et al. 2013; Piraveenan et al. 2013; Tong et al. 2010). We choose to follow the principle of *interlacing*, which is expressed by Perron–Frobenius theorem and states that the first (largest) eigenvalue of the adjacency matrix of a graph reduces when removing a node or a link (Chung 2014). Let λ denote the first (largest) eigenvalue of a graph G. A large λ value indicates a stronger connectivity among the nodes of G through a large number of paths. In this case, G is expected to allow for effective spread of information among its nodes. Let $\Delta\lambda(i)$ (called eigen-drop) denote the reduction in λ after removing node $i \in N$ and all links adjacent to it. The larger this reduction is, the more impact node i is expected to have on information spread throughout G, since its removal makes G have a smaller λ value, which indicates a weaker connectivity among its remaining nodes.

After measuring the eigen-drop $\Delta\lambda(i)$ of each node $i \in N$ by removing i from G, the eigenvalue of the remaining graph changes. Therefore, generating the set S of top-k nodes with the highest total eigen-drop would require exponential cost, due to the need to examine all possible combinations of the k nodes, which becomes intractable for large social graphs. To overcome this problem, we follow a greedy approximate algorithm that has been used by Tong et al. (2010).[1] The greedy Algorithm 1 first computes the corresponding coordinate in the first eigenvector $\mathbf{u}$ (i.e., the eigenvector corresponding to the first eigenvalue λ) (step 1). Then, the impact factor of each node on the connectivity of the graph is calculated without considering any prior removal of the graph's nodes (steps 2–6). Therefore, the eigenvalue change of the graph's nodes is measured in $\mathbf{v}$ as follows:

$$v(i) = 2 \cdot \lambda \cdot u(i)^2 \tag{1}$$

Nodes with higher eigen-score $u(i)$ contribute to higher impact factor. The removal of a node with high impact factor contributes to the reduction of the eigen-drop $\Delta\lambda$ value rather than removing a node with a low eigen-score $u(i)$. To compute the top-k nodes of the set S with the highest diffusion coverage score, we iteratively select the node i with the highest eigen-drop value $\Delta\lambda(i)$.

In each iteration (steps 7–19), a matrix $H \in \mathbb{R}^{|N| \times |S|}$ based on A is created, where $A \in \mathbb{R}^{|N| \times |N|}$ is the adjacency matrix. The matrix H contains the existing links in L that are adjacent to the already selected nodes S.[2] The matrix H is then multiplied by the eigenvector $\mathbf{u}(\mathbf{S}) \in \mathbb{R}^{\mathbf{1} \times |\mathbf{S}|}$ (step 9)

[1] We have to note that the focus of Tong et al. (2010) is to identify which nodes should be removed from a network to make it more robust against epidemic spread. In contrast, our goal is to identify the nodes that make the graph more susceptible to the (viral) spread of information when injecting new links adjacent to these nodes.

[2] At the beginning of the algorithm, since the set S does not contain an item, the matrix H is also empty.

which contains the corresponding eigen-score of the S nodes and the result is stored into the vector $\mathbf{b} \in \mathbb{R}^{\mathbf{1}\times|\mathbf{S}|}$. Next, we can calculate the diffusion-coverage score $\Delta\lambda(i)$ of each node $i \in N \backslash S$ as follows (steps 10–16):

$$\Delta\lambda(i) = v(i) - 2 \cdot b(i) \cdot u(i), \quad \forall i \in N \backslash S \tag{2}$$

At the end of each iteration (step 17), we select the node i with the highest $\Delta\lambda(i)$. Finally, the $\Delta\lambda(S)$ with the diffusion coverage score of the top-k most important nodes of the set S is computed.

Algorithm 1:

```
   Input: $A$: the adjacency matrix of the graph
          $k$: the number of nodes
   Output: $S$: a set with $k$ nodes
1  compute the first eigen-value $\lambda$ of $A$;
2  let $\mathbf{u}$ be the corresponding eigen-vector $\mathbf{u}(j)(j = 1 \times n)$;
3  initialize $S$ to be empty;
4  for $j = 1$ to $|N|$ do
5      $v(i) = (2 \cdot \lambda - A(j,j)) \cdot u(j)^2$;
6  end
7  for $iter = 1$ to $k$ do
8      $H = A(:,S)$;
9      $b = H \cdot u(S)$;
10     for $j = 1$ to $N$ do
11         if $j \in S$ then
12             $\Delta\lambda(j) = -1$;
13         else
14             $\Delta\lambda(j) = v(j) - 2 \cdot b(j) \cdot u(j)$;
15         end
16     end
17     $i = argmax_j \Delta\lambda(j)$;
18     add $i$ to set $S$;
19 end
20 return $S$;
```

4.2 Link-injection strategy

After identifying the set of nodes S with the top-k largest $\Delta\lambda$ values, we opt for injecting links that will be adjacent to the nodes of S.

Depending on the link injection strategy that we follow, a predefined maximum number of links m to the injected matrix A' is defined. Let L_S be the set of the links currently existing to the top-k nodes of the set S, where $L_S \subset L$. The predefined threshold m is expressed as the multiplication $m = |L_S| * p$, where p is a constant factor.

Moreover, not all the nodes are injected with the equal number of links, since each node has different importance in the diffusion coverage of the graph. An upper bound $\mathrm{ub}(i)$, $i = 1, \ldots, |N|$ of predefined number of links, available to be inserted on each node, is calculated based on the multiplication $d(i) \cdot p$, where $\mathbf{d}$ is a vector which contains the degrees of the nodes and p is a constant factor. The proposed link assignment algorithm is presented in Algorithm 2, considering both the nodes' diffusion coverage score and the NMF value for the link injection.

The set of candidate links for injection is calculated as follows. The candidate set is generated in a collaborative-filtering fashion, by factorizing the adjacency matrix of the graph and predicting likelihood of all non-existing links that can be injected. For this purpose we use the non-negative matrix factorization (Berry et al. 2003) of the adjacency matrix A of graph G, which reveals the latent associations between nodes based on their existing links. By factorizing the $A \in \mathbb{R}^{|N|\times|N|}$ adjacency matrix according to NMF, a new matrix $A_{\mathrm{NMF}} \in \mathbb{R}^{|N|\times|N|}$ is computed based on $A_{\mathrm{NMF}} = WU$, where $W \in \mathbb{R}^{|N|\times D}$, $U \in \mathbb{R}^{D\times|N|}$, and D is the number of latent factors. NMF generates these features. NMF identifies the latent factors that express associations between the existing user's links in the adjacency matrix A. Each column in the product matrix WU is a linear combination of the D column vectors in matrix W with coefficients supplied by matrix U. Therefore, the entries of the factorized matrix A_{NMF} are built by a small set of D hidden factors, which can be considered as clusters (groups) of related users. D can be significantly less than N and is usually expressed as a percentage of $|N|$.

NMF minimizes the objective function $F(W,U) = ||A - WU||_F^2$, where $||\cdot||_F^2$ denotes the Frobenius norm. In our approach, we used the alternating non-negative least square algorithm of Liu et al. (2013), which follows an iterative approach. The value of elements of A_{NMF} that correspond to node pairs, for which there is no existing link in G, predict the likelihood of new links that can injected between them. Conversely to other factorization methods, such as the singular valued decomposition

(SVD), NMF generates matrix A_{NMF} with non-negative elements, which is required in our case, because the elements correspond to occurrence likelihood of new links.

The set of candidate links can be, therefore, selected among the predicted links with the highest likelihood. As described in Sect. 3, such links can be recommended to the users of a social network. Nevertheless, we will finally focus on a small subset of the candidate links that offers the highest chances to enable more effective information spread. This procedure is described next in Sect. 4.3.

4.3 Link-injection algorithm

In the proposed link-injection algorithm, our goal is to assign more links to the nodes with high diffusion coverage scores rather than nodes with low ones. In doing so, we create a susceptible graph by promoting the most important nodes to get connected with nodes/friends of high latent association.

Our proposed link assignment strategy is presented in Algorithm 2. The input of the algorithm is the adjacency matrix A; the NMF matrix A_{NMF}; the vector $\Delta\lambda(S)$ with the diffusion coverage scores of the top-k nodes in the set S; the vector $\mathbf{d}$ with the degree of each node; the vector $\mathbf{ub}$ with the upper bound of the links that each node can be injected; and m the maximum number of links that are allowed to be inserted into the adjacency matrix A'. First, in line 1, we assign the existing links in the adjacency matrix A to A'. Then, in steps 2–6, we construct the matrix B which contains the weight of each potential link to be injected to the matrix A'. Since our aim is to promote the link injection to the nodes with the highest diffusion coverage value, while connecting nodes with high latent associations, in step 4 we multiply the diffusion coverage $\Delta\lambda(i)$ of the node i with the NFM weight $A_{\mathrm{NMF}}(i, j)$ for each potential link of the node i with the node j. In steps 7–23, the link assignment algorithm is applied according to the values of the matrix B. In lines 8 and 9, we identify the coordinates of the *sourceNode* and the *targetNode*, with the maximum value. If the degrees of any of the *sourceNode* and *targetNode* do not exceed their upper bound constraints, then the link is assigned to the injected matrix A'. Furthermore, the degrees of both the *sourceNode* and the *targetNode* are increased and the m parameter is decreased by two. On the other hand, if one of the $d(\mathit{sourceNode})$ or $d(\mathit{targetNode})$ has exceeded its upper bound constraint, then the link is not injected into the matrix A' and the weight of the link is removed by the matrix B, as shown in steps 18 and 19. As a result, in step 10 we iteratively examine if the B matrix is empty while we still have remaining links to inject. This can be accomplished by exceeding the limits of the links inserted into each node $i \in S$.

Algorithm 2: Link Assignment Algorithm

```
    Input: $A$: the adjacency matrix of the graph
    $A_{\mathrm{NMF}}$: the NMF Matrix
    $\Delta\lambda(S)$ : the vector with the diffusion coverage score of each node in the set $S$
    d: the vector with the degree of each node
    ub: vector with the upper bound of links per node
    $m$: the number of maximum link injection
    Output: $A'$: the link injected matrix
1   initialize $A'$ with the existing links of $A$;
2   foreach $i \in S$ do
3       for $j = 1$ to $|N|$ do
4           $B(i,j) = \Delta\lambda(i) \cdot A_{\mathrm{NMF}}(i,j)$;
5       end
6   end
7   while $m > 0$ do
8       $sourceNode$ = identify the row with the highest value in $B$;
9       $targetNode$ = identify the column with the highest value in $B$
10      if $B(sourceNode, targetNode) \neq 0$ then
11          if $(d(sourceNode) < ub(sourceNode))$ &
            $(d(targetNode) < ub(targetNode))$ then
12              $A'(sourceNode, targetNode) = 1$;
13              $A'(sourceNode, targetNode) = 1$;
14              $d(sourceNode) += 1$;
15              $d(targetNode) += 1$;
16              $M -= 2$;
17          end
18          $B(sourceNode, targetNode) = 0.0$;
19          $B(targetNode, sourceNode) = 0.0$;
20      else
21          $m = 0$;
22      end
23  end
24  return $A'$;
```

According to the proposed link assignment algorithm, nodes with high diffusion coverage scores are suitable to gain maximum link injection compared to nodes with low ones. Moreover, the multiplication of the nodes' diffusion coverage score with its NMF score (see step 4) is applied to identify the suitable nodes which significantly help the diffusion coverage process by applying link injection. More particular, some nodes may have low NMF scores but high diffusion coverage scores. Such nodes will not gain a lot of injected links, because we assume that in these cases no major benefit will be achieved in the spread of influence, because the low NMF scores indicate low affinity between such nodes.

The computational complexity of the proposed algorithm depends on the number of most vulnerable nodes $k = \%|N|$ and the number of maximum link injection $m = \%|L_s|$. Based on Algorithm 2, the initialization of matrix B requires a $O(k \cdot |N|)$ cost. Moreover, the assignment process (lines 7- 23) examines all m potential links, which results in an additional $O(m)$ cost. Summarizing, the total complexity of the Link Assignment Algorithm is $O(k \cdot |N|) + O(m)$.

5 Experimental setup

The objective of our experimental evaluation is to examine the impact of link injection on the spread of information. To consider a large variety of information spreads under several parameter settings, we describe in Sect. 5.1 the underlying diffusion model that extends the Independent Cascade (IC) and the Linear Threshold (LT) model (K-empe et al. 2003).

5.1 Diffusion models

To parameterize information-spread processes, we examine both the Independent Cascade (IC) and Linear Threshold (LT) models (Kempe et al. 2003). Both are widely used to model information diffusion in social networks. We consider an underlying network graph $G = (N, L)$ of N nodes and L edges, and a seed set I of nodes, $I \subset N$. In the rest of this section, the IC and LT model are presented.

5.2 Independent cascade model

IC models the individual influence that each user has on his neighbors. IC starts by activating each user in the seed set I and proceeds in discrete steps t by attempting to activate the rest nodes. Let $I_t \subset N$ be the set of nodes that are activated at step $t \geq 0$, with $I_0 = I$. At the step $t+1$, each user $v \in I_t$ has a single chance to activate each of the currently inactive users $w \in N$ that are neighbors of (i.e., connected to) v. User v succeeds in activating a neighbor w with probability p_{vw} equal to the (normalized) influence factor that v has on w.[3] If v succeeds then w becomes active in step $t+1$ and, thus, w becomes a member of the set I_{t+1}. Otherwise, v makes no further attempts to activate w. The process continues recursively and each node in I_{t+1} tries to activate its neighbors. The process stops at time $t^* \geq 0$ if $I_{t^*} = \emptyset$, since no more activations are possible.

In IC, social influence is the only factor that determines the activation of users. However, the inherent preference that users have to the diffused information (product, brand, etc.) is also a factor that determines the activations (Lawton and Gregor 2003). To account for this factor, we extend IC accordingly, by associating each user w with a random variable θ_w that follows the Beta distribution $\theta_w \sim Beta(\alpha, \beta)$ and characterizes the *resistance* of the user to become activated in the presence of social influence. Evidently, lower resistance indicates higher inherent preference for the diffused information and thus more easy activation due to social influence. The reason for using Beta distribution is twofold: (1) first, it returns values within a pre-specified numerical range (i.e., $[0, 1]$), which can quantify the resistance of users to adopt the diffused information as ranging between weak and strong.[4] Thus, θ_v takes values in the range $[0, 1]$, where values closer to 0 correspond to weaker resistance, i.e., user v is more likely to get activated due to social influence. (2) Second, Beta is a very flexible distribution that can be easily controlled by its two parameters α and β. The mean value is equal to $\alpha/(\alpha + \beta)$ and, thus, by setting $\beta > \alpha$ the distribution is shifted more towards weaker resistance and by setting $\beta < \alpha$ towards higher resistance. In the special case where $\alpha = \beta = 1$, Beta becomes identical to the uniform distribution, which corresponds to a 'neutral' setting where all users have identical resistance. For this reason, in our experimental results presented in the next Sect. 6, we set $\beta = 1$ and vary the value of α to progressively depart from such a uniform setting. We focus on the more challenging cases, by considering values $\alpha > \beta$, which, as explained above, correspond to stronger resistance, which in turn hinders activations.

Based on the above, each existing link between two users, users v and w, is assigned a weight W_{vw} equal to:

$$W_{vw} = p_{vw} + \gamma \times max(\theta_w; 1 - \theta_w) \times \theta_w \tag{3}$$

where, as mentioned above, p_{vw} is the (normalized) influence factor that v has on w. In Eq. 3, $\gamma \times max(\theta_w, 1 - \theta_w)$ represents the fact that users with stronger inherent

[3] Each p_{vw} is computed by dividing the sum of all weights of incoming ties to w.

[4] Please note that several other probability distributions do not satisfy this property, by having an unbounded support.

preferences tend to adhere to them (Mussweiler and Strack 2000). Thus, the more extreme θ_w is (i.e., the closer it is to 0 or 1), the more significant the impact of θ_w becomes. The value of γ is set to -1, if θ_w is not less than 0.5 (i.e., indicates higher resistance); and to $+1$, if θ_w is less than 0.5 (i.e., indicates lower resistance). Therefore, when a user v tries to activate another user w, according to this extended IC model, the resulting probability of activation p'_{vw} is calculated as follows:

$$p'_{vw} = \begin{cases} 0, & \text{if} \quad \mathrm{W_{vw}} < 0 \\ 1, & \text{if} \quad W_{vw} > 1 \\ W_{vw}, & \text{otherwise} \end{cases} \tag{4}$$

In summary, according to Eqs. 3 and 4, the activation of each user w depends both on the social influence (expressed by p_{vw}) and on the inherent preferences of w as characterized by resistance θ_w.[5]

IC assumes that all activated users will try to activate their neighbors. Nevertheless, not all activated users—no matter how positive their opinion about a product or a brand is—will pass on the message by trying to activate their neighbor users, because they may just keep it to themselves or forget about the whole experience all together. To take this into account, we assume that all users have a 'stopping probability' (equal for all nodes) and when they become activated, they try in turn to activate their neighbors according to this 'stopping probability'. Therefore, the higher the 'stopping probability', the higher is the 'difficulty' of the social network since more users are reluctant to participate in the viral process.

We have to emphasize that, in contrast to existing research (Chaoji et al. 2012), we assume no knowledge of influence factors (i.e., the aforementioned probabilities of the form: p_{vw} for each pair of users v and w) during the prediction of the injected links. We examine influence factors in the IC model only during the evaluation of the effectiveness of the injected links.

5.3 Linear threshold model

The LT model differs from the aforementioned IC model, because it represents the combined peer-pressure effect within the social network. Given a network graph $G = (N, L)$ of N nodes and L edges, LT starts the diffusion process on a seed set I at time step t. At each step $t+1$ each user w is influenced by each neighbor v according to a weight b_{vw}, such that $\sum_{v \in N} b_{vw} \leq 1$. If this sum is larger than a threshold $\theta_w \in [0, 1]$, then w becomes activated.

Similarly to the aforementioned case for IC, the threshold follows again a Beta distribution, i.e., $\theta_w \sim Beta(\alpha, \beta)$. Thus, the threshold characterizes the *resistance* of the user to become activated in the presence of social influence from all neighbors.

5.4 Data sets

In our experiments we used five real data sets. The first two data sets of trust networks are provided by the Arizona State University [6] featuring the Ciao and the Epinions data sets. Ciao consists of 7,317 users. Each user can rate items by writing reviews and establish trust networks with their like-minded users. In this network, 177,727 connections exist which indicate the relationships between the users. Similar to the Ciao data set, Epinions consists of 18,098 users and 529,162 trust connections between the users. The third data set is the Youtube data set (Tang et al. 2009). In this data set the contact network between the Youtube users is provided, consisting of 13,723 users and 167,253 connections. Each connection corresponds to at least one shared favorite video between two users. Moreover, in our experiments we used two data sets from Facebook and Twitter. The Facebook data set consists of 46,952 users and 274,086 connections and the Twitter data set contains 456,631 users and 14,855,875 connections.

In both the Ciao and Epinions data sets, we create a graph based on the common preferences of the users. In doing so, we consider that positive preferences are expressed if two users rate an item equal to 5. In the Youtube data set the common preferences are considered as the number of the shared favorite videos of the users. For each data set the adjacency matrix A of the graph is created, corresponding to the connections among the users. In Facebook a connection between a user u and a user v is weighted according to the number of wall posts that u has sent to v. Furthermore, each Twitter connection between two users u and v is associated to the number of retweets that u has performed to tweets from v. The characteristics of all examined data sets are summarized in Table 2.

In the rest of our experiments, we consider the PageRank algorithm as the seed-selection strategy, since it outperforms the other network similarity measures, i.e. betweeness, degree centrality, etc. (David and Jon 2010). Moreover, the selected number of nodes for the initial seed

[5] Notice that in Eq. 4, the truncation of weight W_{vw} into the interval $[0, 1]$ results in an activation probability p'_{vw}. The reasoning behind this truncation is as follows: in IC, the attempt of a user v to activate a neighbor user w is implemented by generating a random number r that follows uniform distribution in the interval $[0, 1]$. The value of r is then compared to the weight W_{vw}. User w becomes activated, if $r < W_{vw}$. Thus, negative values of the weight W_{vw} correspond to an activation probability $p'_{vw} = 0$, since in this case it always holds that $r \nless W_{vw}$. Similarly, values of the weight W_{vw} that are higher than 1, correspond to an activation probability $p'_{vw} = 1$, since in this case it always holds that $r < W_{vw}$.

[6] http://www.public.asu.edu/~jtang20/datasetcode/truststudy.htm.

Table 2 The five real-world data sets of Ciao, Epinions, Youtube, Facebook and Twitter

Data set	Users	Connections	Average degree	Diameter	Clustering coefficient
Ciao	7,317	177,727	23.106	10	0.218
Epinions	18,098	529,162	25.898	9	0.209
Youtube	13,723	167,253	10.176	12	0.159
Facebook	46,952	274,086	6.726	20	0.103
Twitter	456,631	14,855,875	28.642	11	0.1887

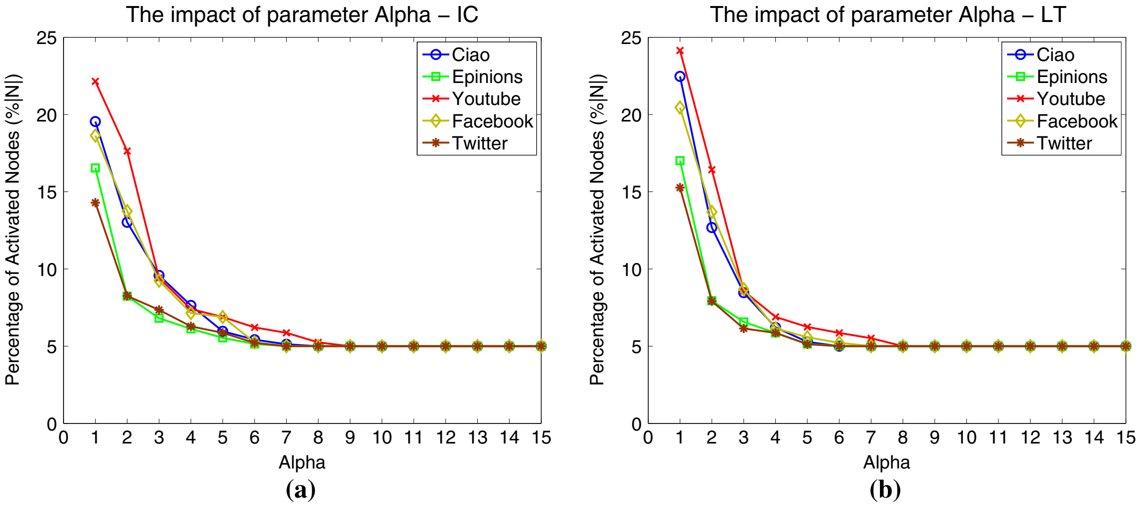


Fig. 1 The impact of parameter α on IC and LT models for Ciao, Epinions, Youtube, Facebook and Twitter data sets, in terms of percentage of activated nodes ($\%|N|$)

set is expressed as a percentage (%) of $|N|$, i.e., of the total number of nodes. The default percentage for each data set is 5 %; however, we examine the impact of the seed size separately in Sect. 6.4. The default value for ‘stopping probability’ is set to 0.25. In our experiments we report average results out of 100 trials, since the examined diffusion models are probabilistic.

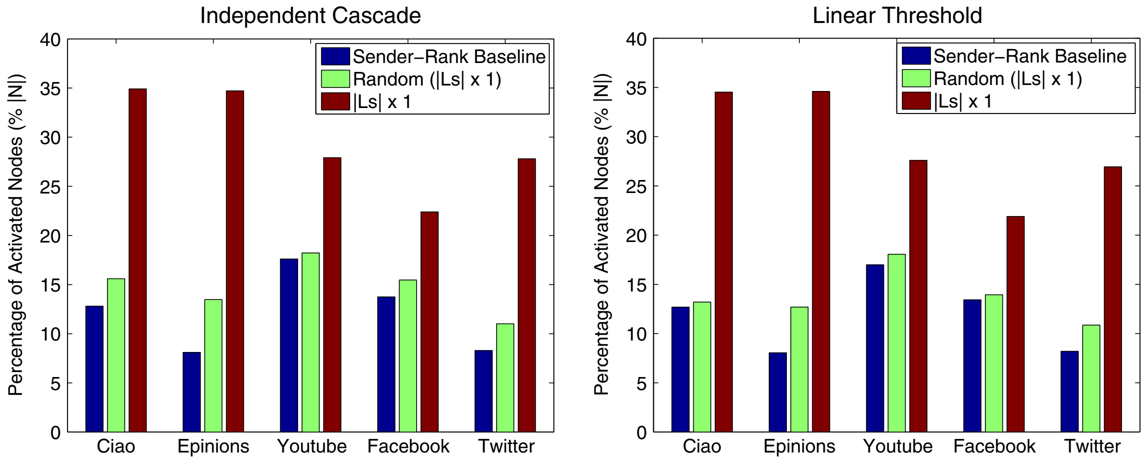
6 Experimental results

6.1 Susceptibility to information cascades

In Sect. 5.1 we defined that the probability of a user v to activate his neighbor w is affected by the Beta distribution which assigns θ values to the nodes. To calculate the θ values, two basic parameters need to be defined, the α and β parameters. The α parameter affects the activation probability of each node, since lower values of α tend the network to be more susceptible to the information-cascade process, by being more willing to accept the influence provided by their neighbors. On the other hand, larger values of α contribute to a more conservative network, where users are not easily affected by their neighbors opinions and, thus, information cascades are more difficult to develop. To evaluate the impact of the α parameter, we have conducted several experiments by varying the α parameter while the β parameter was set to 1 (Sect. 5.1). In Fig. 1, we present the corresponding experimental results.

As expected, for α equal to 1 we have the maximum number of activated nodes, whereas an increase in the value of α reduces the percentage of activated nodes. In the rest of experiments, for IC we set α equal to 2 and for LT equal to 1.78. These values correspond to a realistic and challenging scenario, where users are mostly not very susceptible to information cascades, but such cascades are on the other hand not impossible to take place.[7]

[7] The small difference in the α values used with IC and LT (2 and 1.78, respectively) is justified by the results in Fig. 1a, b, which refer to the performance of the SenderRank baseline and, thus, the small discrepancies in the number of activations are due to the subtle differences between the two diffusion models themselves. We selected the α value for LT accordingly (based on linear interpolation) so that we can more clearly identify in the sequel the performance gains due to link injection, after having first aligned the performance of the SenderRank baseline w.r.t. the two diffusion models.



(a) (b)

Fig. 2 Comparison against the SenderRank and Random baselines. The proposed Link Injection method and the Random baseline use the $m = 1 \times |L_S|$ upper bound of injected links, where $|L_S|$ is the number of links of the top-k most important nodes with the highest diffusion coverage score

6.2 Comparison against baselines

In the following set of experiments we evaluate the impact of our link injection strategy on information spread, by comparing the number of activations it achieves compared to baseline methods. As described in Sect. 4.3, we inject new links to the nodes with high score of diffusion coverage. Since the top-k most important nodes of the graph contain $|L_S|$ existing edges, we have conducted several experiments by varying the number m (upper bound) of maximum links to be injected, based on the $|L_S|$ number of edges. We evaluate the proposed methodology against the case where no new connections are being inserted (we refer to this case as SenderRank baseline). Moreover, we present the impact of our link injection strategy, compared against a random-selection baseline, where the same number of links are assigned to randomly selected nodes of the graph. In this experiment, the upper bound m of the injected links added to the top-k most important nodes is defined as the $\times 1$ of the $|L_S|$ existing edges, while k is set to the product of $\times 0.1$ and the $|N|$ number of nodes in the network.

Figure 2a presents the results of this comparison for all examined data sets in the case where IC is the underlying diffusion model, whereas Fig. 2b in the case where LT is the underlying diffusion model. As shown, for both diffusion models, the proposed link injection methodology boosts information propagation for all data sets. We performed double t tests and found that the reported differences of the proposed method against both the SenderRank and Random baselines are statistically significant at level 0.05 for all data sets and for both IC and LT models.

Based on the finding that the relative behavior of all examined methods is similar for both the IC and LT models, for reasons of brevity, in the following we focus on the IC model.

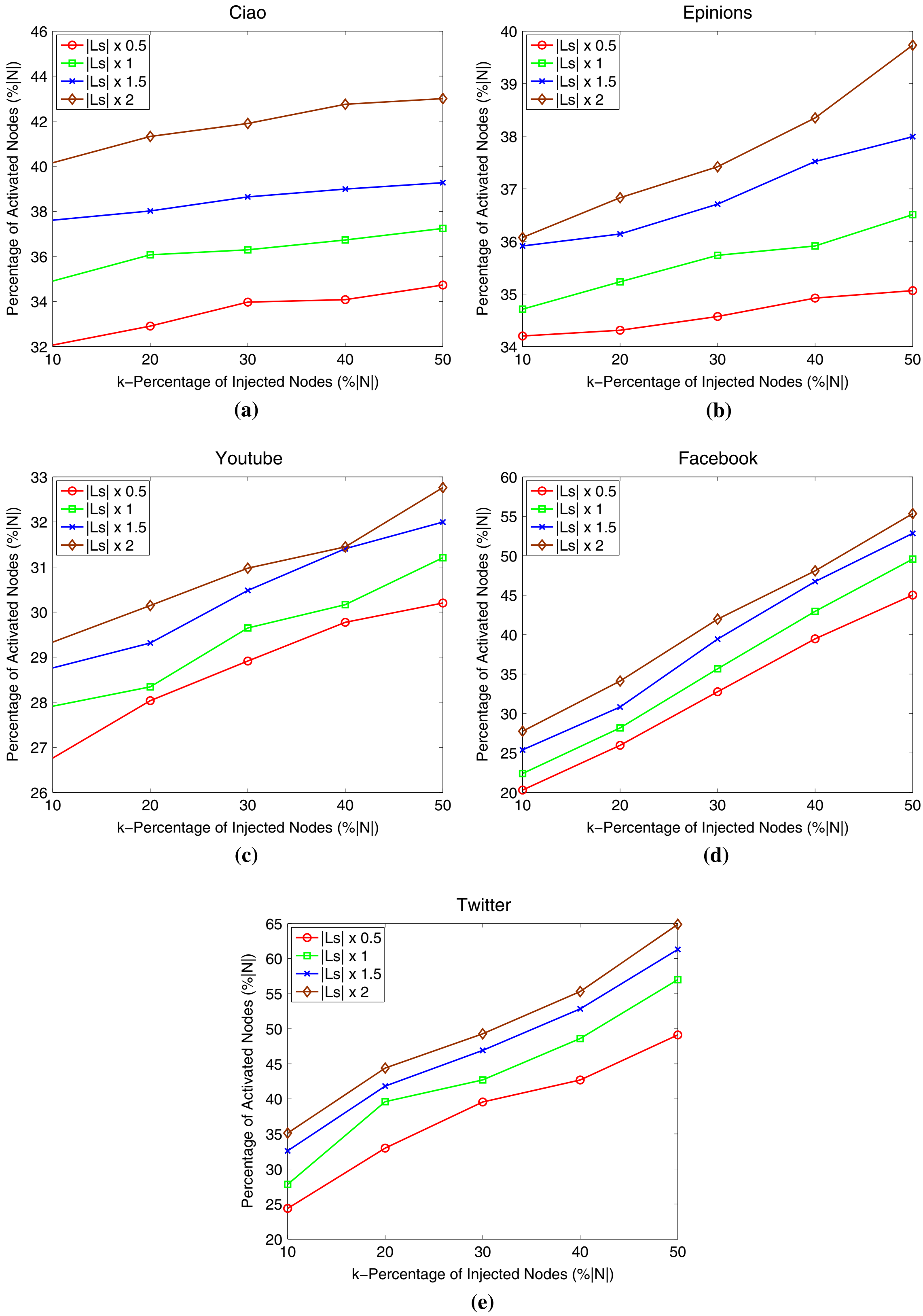
6.3 Upper bound of the number of injected links

In Fig. 3, we evaluate the impact of the maximum link-injection number m on the overall information spread, while the factor k of the most important nodes is increased. Also, the number of already existing edges in L_S is varied from $\times 0.5$ to $\times 2$. By increasing the maximum number of injected links, the percentage of activated nodes is also increased. Moreover, the factor k of the most important nodes is also essential. Since link injection in more than the 50 % of the total number of nodes is not a realistic scenario, in our experiments the k parameter ranges from 10 to 50 % of the total number of nodes $|N|$. Based on the results of Fig. 3, in case that link injection is performed in more k nodes with the highest diffusion coverage score, wider information spread is achieved.

We have to note that, although the number of maximum link injection (upper bound m) varies from $\times 0.5$ to $\times 2$, our link assignment algorithm (Sect. 4.3) does not assign all the links to the top-k most important nodes. This occurs due to the upper bound m of links which are candidate to be inserted to each node. In Fig. 4, we present the number of injected links in the five data sets. As expected, the number of injected links never exceeds the total number of links contained in the overall graph ($< 100\ \% \times |L|$).

6.4 Seed size

Moreover, we examined the performance of our methodology by varying the number of the selected nodes to



(a)

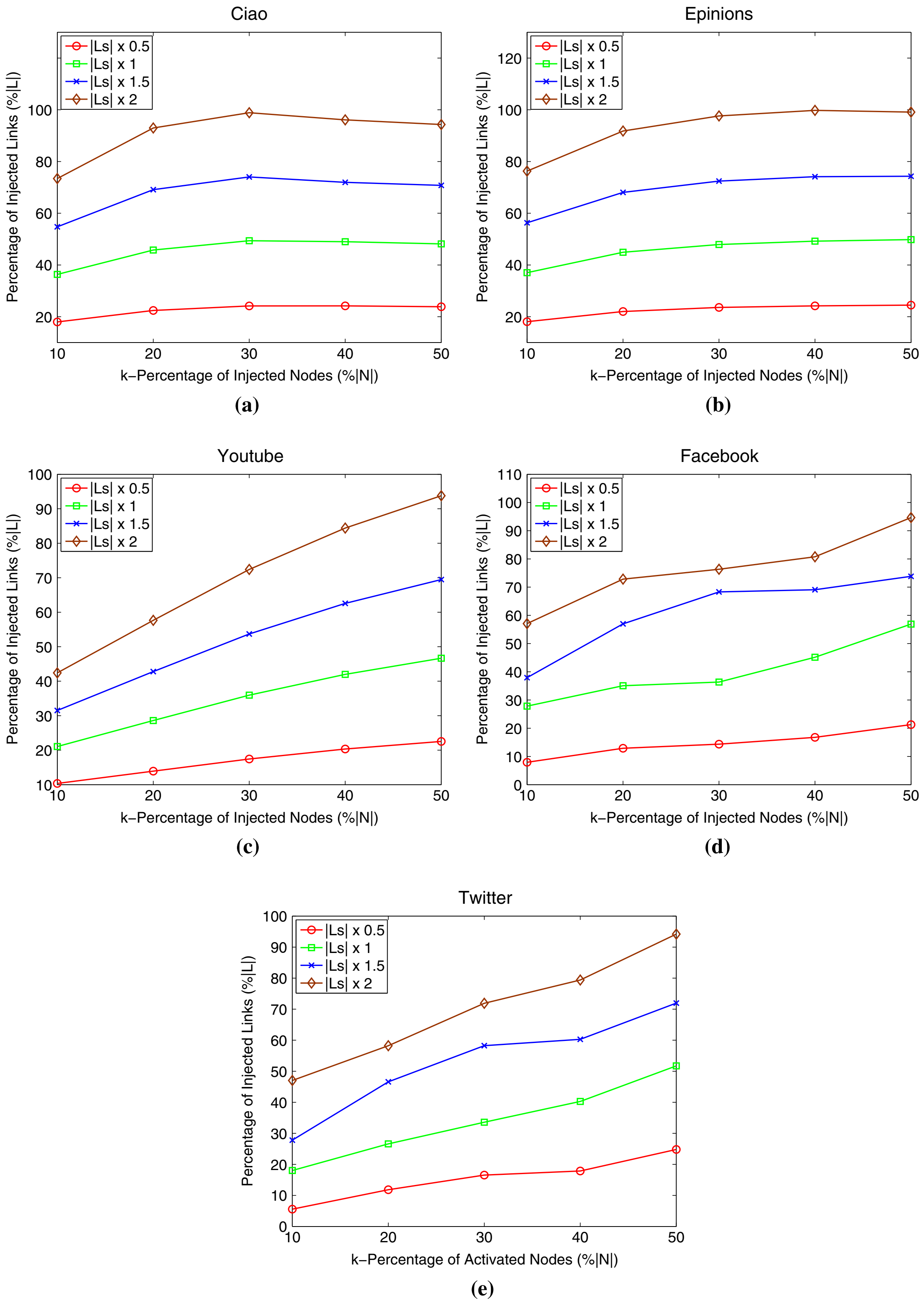
(b)

(c)

(d)

(e)

Fig. 3 The impact of the link injection factor on the information cascade process for **a** Ciao, **b** Epinions, **c** Youtube, **d** Facebook and **e** Twitter, in terms of percentage of activated nodes ($\%|N|$), by varying the k-percentage of injected nodes ($\%|N|$) with the highest diffusion coverage score



(a)

(b)

(c)

(d)

(e)

Fig. 4 The impact of the link injection algorithm on the actual injected links ($\%|L_S|$) in **a** Ciao, **b** Epinions, **c** Youtube, **d** Facebook and **e** Twitter, by varying the k-percentage of injected nodes ($\%|N|$) with the highest diffusion coverage score

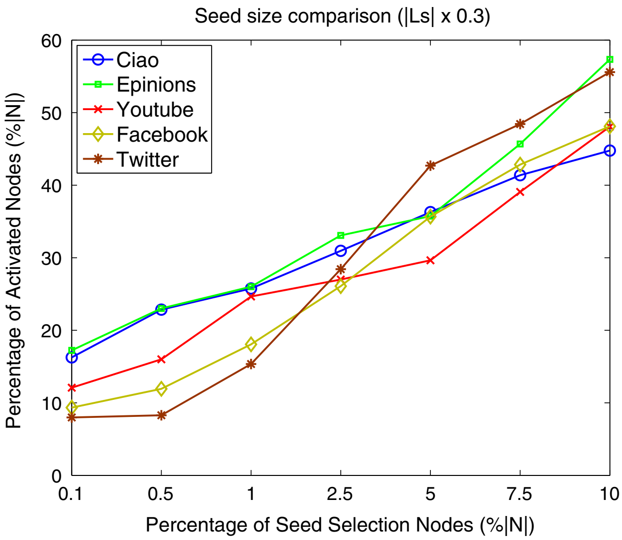


Fig. 5 The impact of the seed set size

initialize the information propagation. In Fig. 5, we present our experiments for different factors ($\%|N|$) of initial selected nodes, while our injection factor strategy is preserved in each variation ($k = |N| \times 0.3, m = |L_S| \times 1$). By increasing the number of initial selected nodes our approach contributes to higher spread of information through the network.

6.5 Acceptance of recommended links

For the link injection in social networks perspective, a set of recommended links is provided to the users. To increase the accuracy of such recommendations, our proposed link prediction method follows a collaborative-filtering approach (see Sect. 4), recommending thus links that are likely to be accepted by the users. The predefined upper bound of the number of recommended links guarantees that the experience of users, who usually are not willing to receive a large number of link recommendations, is not significantly affected. However, we have to note that our experimental evaluation does not involve a user study, where such link recommendations are provided to real users. Thus, so far, we have performed an indirect experimental evaluation, based on the premise that the generated links can be injected directly. This assumption is justified by the high recommendation accuracy of the used link-prediction method, as shown in Sect. 3. To evaluate the accuracy of the examined link-prediction methodology based on the NMF, the Area Under the Curve (AUC) measure (Ling et al. 2003) has been applied to the five data sets, namely Ciao, Epinions, Youtube, Facebook and Twiter. We have randomly partitioned the data sets into testing and training subsets, where the half of the dataset consist the training subset and the rest of the data set consists the testing subset. According to this partition, AUC for Ciao, Epinions, Youtube, Facebook and Twitter is 0.94, 0.87, 0.91, 0.79 and 0.88, respectively. This result demonstrates that the examined link-prediction method is effective and can be used as a basis of the proposed approach.

Nevertheless, to evaluate the sensitivity of the proposed method w.r.t. the probability of accepting the recommended links, we conducted the following experiment: a fraction of the recommended links is assumed to be accepted and finally injected into the social network. This experiment, thus, simulates the impact of the overall acceptance probability of the recommended links. The results are presented in Fig. 6. For the proposed method, we followed a 30 % link-injection strategy by varying the percentage of the recommended links that will be accepted and injected.

As expected, the proposed method achieves higher percentage of activated nodes by increasing the percentage of accepted links. Even for relatively small percentages, the proposed method still outperforms the baseline (when no injected links are used). This demonstrates that the proposed method can still be useful for applications where users are not very willing to accept the recommended links.

7 Discussion

Our proposed methodology is based on the premise that nodes with high diffusion coverage will agree to add connections to a number of other nodes to enhance the ability of a social network to spread information more effectively. Such a premise is reasonable, since the incentive of important nodes in social networks, i.e., the nodes with the highest diffusion coverage, is to increase the number of their social connections. In friendship social networks, such as Facebook, the friendship connection is confined since connections require the acceptance of both sides. On the other hand, in content-centric networks, such as Twitter and Google+, the expansion of friendship connections could be achieved more easily, since the important nodes do not have to accept the friendship connection. In doing so, our proposed link injection does not demand any other form of engagement from the important nodes. Since the information propagation has been initiated, they can follow their own decisions to accept and spread further the propagated information.

Nevertheless, our approach presents the advantage of keeping the number of generated recommendations controlled. The reason is that, although important nodes may be willing to increase the number of users connected to them, they may not be available (or willing) to accept a

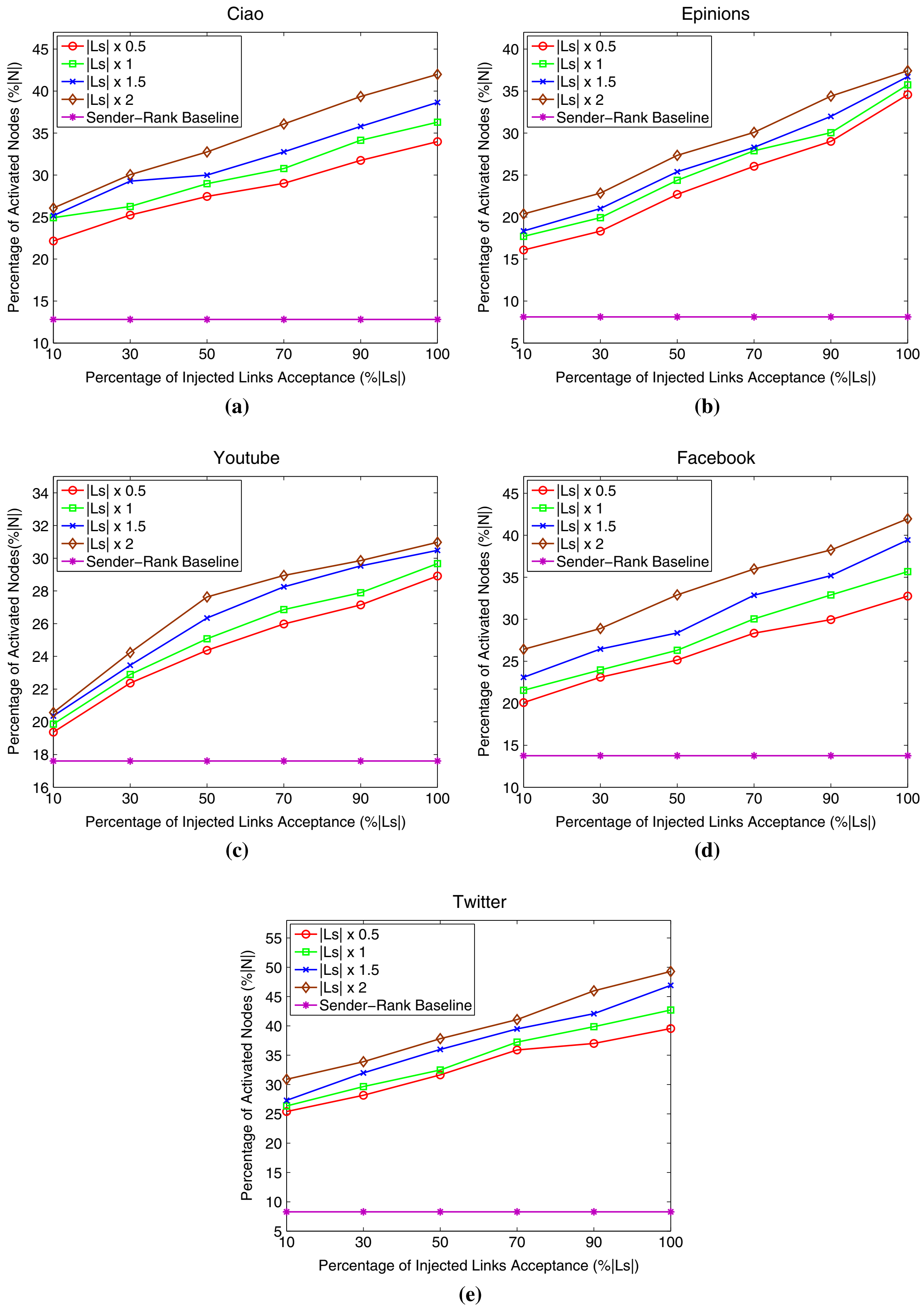


Fig. 6 The impact of the acceptance probability of the recommended links for: **a** Ciao, **b** Epinions, **c** Youtube, **d** Facebook and **e** Twitter

Table 3 The impact of the link injection strategy to the characteristics of the graph

Data set	Average degree	Diameter	Clustering coefficient
Ciao	23.106	10	0.218
Ciao ($\|N\| \times 0.3, \|L_S\| \times 1$)	25.019	10	0.267
Epinions	25.898	9	0.209
Epinions ($\|N\| \times 0.3, \|L_S\| \times 1$)	27.356	8	0.257
Youtube	10.176	12	0.159
Youtube ($\|N\| \times 0.3, \|L_S\| \times 1$)	11.862	10	0.164
Facebook	6.726	20	0.103
Facebook ($\|N\| \times 0.3, \|L_S\| \times 1$)	6.821	18	0.141
Twitter	28.642	11	0.1887
Twitter ($\|N\| \times 0.3, \|L_S\| \times 1$)	29.217	10	0.204

very large number of recommendations at a time (for instance, uncontrolled acceptance of links may incur fake connections or other forms of fraud). In our proposed link-injection methodology, only a controlled number of injected links are suggested based on the upper bound constrain of the links that each node may accept. This allows for experimentation in real-world applications, where only a few links can be injected without affecting the experience of the users.

Finally, Table 3 presents the impact of the proposed link-injection strategy on the structure of the graph of each data set, where the diameter and the clustering coefficient are selected as representative statistics of the graph structure. As shown, the proposed method results in general in a smaller diameter and in a larger clustering coefficient. This provides another explanation of the fact that the proposed method outperformed the baselines in the presented experimental evaluation, since it can increase the interconnectivity of users—as expressed at the macroscopic level by the smaller diameter and the larger clustering coefficient. Nevertheless, the proposed methodology opts for the recommendation of new links to the users without substantially altering the characteristics of the graph that represents the links of a social network. The results of Table 3 indicate that the resulting changes in the graph structure are not radical and, thus, the major characteristics of the original graph are being preserved.

8 Conclusions

In this paper, we have presented a link injection methodology to boost the spread of information in social networks. We examine a score for diffusion coverage, which can identify users that can help information spread more effectively. Link injection is attained in a controlled manner, by providing only a predefined upper bound of the number of recommendations to the most important nodes to develop a feasible solution in real-world applications. Extensive experiments on five widely used data sets have demonstrated the effectiveness of our proposed link injection strategy. The proposed method achieves to increase the spread at a rate of 22.1, 26.6, 10.3, 8.7 and 19.5 % on the Ciao, Epinions, Youtube, Facebook and Twitter data sets, respectively.

For future work, we will consider a link-injection methodology based on recommendation strategies to be used in both real-world friendship and content-centric social networks. The effectiveness of our proposed methodology can be examined with a real-user study involving actual recommendations. Finally, alternate models in dynamic network graphs needs further investigation.

References

Anagnostopoulos A, Kumar R, Mahdian M (2008) Influence and correlation in social networks. In: Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '08. ACM, New York, pp 7–15. doi:10.1145/1401890.1401897

Berry MW, Browne M, Langville AN, Pauca VP, Plemmons RJ (2003) Algorithms and applications for approximate nonnegative matrix factorization. Comput Stat Data Anal 52(1):155–173

Bhatt R, Chaoji V, Parekh R (2010) Predicting product adoption in large-scale social networks. In: Proceedings of the 19th ACM international conference on information and knowledge management, CIKM '10. ACM, New York, pp 1039–1048. doi:10.1145/1871437.1871569

Boldi P, Rosa M, Vigna S (2013) Robustness of social and web graphs to node removal. Soc Netw Anal Min 3(4):829–842

Bonchi F, Castillo C, Gionis A, Jaimes A (2011) Social network analysis and mining for business applications. ACM Trans Intell Syst Technol 2(3):22:1–22:37

Borbora Z, Ahmad M, Oh J, Haigh K, Srivastava J, Wen Z (2013) Robust features of trust in social networks. Soc Netw Anal Min 3(4):981–999

Chaoji V, Ranu S, Rastogi R, Bhatt R (2012) Recommendations to boost content spread in social networks. In: Proceedings of the 21st international conference on World Wide Web, WWW '12. ACM, New York, pp 529–538

Chen J, Geyer W, Dugan C, Muller M, Guy I (2009) Make new friends, but keep the old: recommending people on social networking sites. In: Proceedings of the SIGCHI conference on human factors in computing systems, CHI '09. ACM, New York, pp 201–210

Chen W, Wang Y, Yang S (2009) Efficient influence maximization in social networks. In: Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '09. ACM, New York, pp 199–208

Chen W, Yuan Y, Zhang L (2010) Scalable influence maximization in social networks under the linear threshold model. In: Proceedings of the 2010 IEEE international conference on data mining, ICDM '10. IEEE Computer Society, Washington DC, pp 88–97. doi:10.1109/ICDM.2010.118

Chung F (2014) Spectral graph theory. No. 92 in CBMS Regional Conference Series. Conference Board of the Mathematical Sciences. http://books.google.gr/books?id=YUc38_MCuhAC

David E, Jon K (2010) Networks, crowds, and markets: reasoning about a highly connected world. Cambridge University Press, New York

Dinev T, Hu Q, Yayla A (2008) Is there an on-line advertisers' dilemma? A study of click fraud in the pay-per-click model. Int J Electron Commer 13(2):29–60

Goyal A, Bonchi F, Lakshmanan L, Venkatasubramanian S (2013) On minimizing budget and time in influence propagation over social networks. Soc Netw Anal Min 3(2):179–192

Goyal A, Bonchi F, Lakshmanan LV (2010) Learning influence probabilities in social networks. In: Proceedings of the third ACM international conference on web search and data mining, WSDM '10. ACM, New York, pp 241–250. doi:10.1145/1718487.1718518

Guille A, Hacid H, Favre C, Zighed DA (2013) Information diffusion in online social networks: a survey. SIGMOD Rec 42(2):17–28. doi:10.1145/2503792.2503797

Guy I, Ronen I, Wilcox E (2009) Do you know?: Recommending people to invite into your social network. In: Proceedings of the 14th international conference on intelligent user interfaces, IUI '09. ACM, New York, pp 77–86

Hannon J, Bennett M, Smyth B (2010) Recommending twitter users to follow using content and collaborative filtering approaches. In: Proceedings of the fourth ACM conference on recommender systems, RecSys '10. ACM, New York, pp 199–206

Hinz O, Bernd S, Christian B, Becker JU (2011) Seeding strategies for viral marketing: an empirical comparison. J Mark 75(6):55–71

Jackson MO (2011) An overview of social networks and economic applications. In: Benhabib J, Bisin A, Jackson MO (eds) Handbook of Social Economics, vol 1A, Elsevier, Amsterdam, pp 511–585

Kempe D, Kleinberg J, Tardos E (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, KDD '03. ACM, New York, pp 137–146

Kempe D, Kleinberg J, Tardos E (2005) Influential nodes in a diffusion model for social networks. In: Proceedings of the 32nd international conference on automata, languages and programming, ICALP'05. Springer, Berlin, pp 1127–1138. doi:10.1007/11523468_91

Kim YA, Srivastava J (2007) Impact of social influence in e-commerce decision making. In: Proceedings of the ninth international conference on electronic commerce, ICEC '07. ACM, New York, pp 293–302

Kimura M, Saito K (2006) Tractable models for information diffusion in social networks. In: Proceedings of the 10th European conference on principle and practice of knowledge discovery in databases, PKDD'06. Springer, Berlin, pp 259–271. doi:10.1007/11871637_27.

Kiss C, Bichler M (2008) Identification of influencers—measuring influence in customer networks. Decis Support Syst 46(1):233–253

Lawton B, Gregor S (2003) Internet marketing communications: interactivity and integration. In: Seeking success in e-business. Kluwer Academic Publishers, Norwell, pp 239–257

Leskovec J, Krause A, Guestrin C, Faloutsos C, VanBriesen J, Glance N (2007) Cost-effective outbreak detection in networks. In: Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '07. ACM, New York, pp 420–429. doi:10.1145/1281192.1281239

Li YM, Shiu YL (2012) A diffusion mechanism for social advertising over microblogs. Decis Support Syst 54(1):9–22

Liben-Nowell D, Kleinberg J (2003) The link prediction problem for social networks. In: Proceedings of the twelfth international conference on information and knowledge management, CIKM '03. ACM, New York, pp 556–559

Ling C, Huang J, Zhang H (2003) Auc: a better measure than accuracy in comparing learning algorithms. In: Xiang Y, Chaib-draa B (eds) Advances in artificial intelligence. Lecture notes in computer science, vol 2671. Springer, Berlin, pp 329–341

Liu H, Li X, Zheng X (2013) Solving non-negative matrix factorization by alternating least squares with a modified strategy. Data Min Knowl Discov 26(3):435–451

Mussweiler T, Strack F (2000) Numeric judgments under uncertainty: the role of knowledge in anchoring. J Exp Soc Psychol 36:495–518

Narayanam R, Narahari Y (2011) A shapley value-based approach to discover influential nodes in social networks. IEEE Trans Autom Sci Eng 8(1):130–147. doi:10.1109/TASE.2010.2052042

Papadopoulos S, Kompatsiaris Y, Vakali A, Spyridonos P (2012) Community detection in social media. Data Min Knowl Discov 24(3):515–554

Piraveenan M, Thedchanamoorthy G, Uddin S, Chung K (2013) Quantifying topological robustness of networks under sustained targeted attacks. Soc Netw Anal Min 3(4):939–952

Saito K, Kimura M, Ohara K, Motoda H (2010) Selecting information diffusion models over social networks for behavioral analysis. In: Proceedings of the 2010 European conference on machine learning and knowledge discovery in databases: part III, ECML PKDD'10. Springer, Berlin, pp 180–195. http://dl.acm.org/citation.cfm?id=1889788.1889801

Schifanella R, Barrat A, Cattuto C, Markines B, Menczer F (2010) Folks in folksonomies: social link prediction from shared metadata. In: Proceedings of the third ACM international conference on web search and data mining, WSDM '10. ACM, New York, pp 271–280

Tang J, Sun J, Wang C, Yang Z (2009) Social influence analysis in large-scale networks. In: Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '09. ACM, New York, pp 807–816. doi:10.1145/1557019.1557108

Tang L, Wang X, Liu H (2009) Uncovering groups via heterogeneous interaction analysis. In: Proceedings of the 9th international conference on data mining, ICDM '09, pp 503–512

Tong H, Prakash BA, Tsourakakis C, Eliassi-Rad T, Faloutsos C, Chau DH (2010) On the vulnerability of large graphs. In: Proceedings of the 2010 IEEE international conference on data mining, ICDM '10. IEEE Computer Society, Washington DC, pp 1091–1096