Identifying influential spreaders by weighted LeaderRank

Qian Li [a,b,c], Tao Zhou [a,*], Linyuan Lü [d,e], Duanbing Chen [a]

[a] *Web Sciences Center, University of Electronic Science and Technology of China, Chengdu 611731, People's Republic of China*

[b] *Big Data Lab, Cloud Valley, Beijing 100176, People's Republic of China*

[c] *College of Physics and Technology, Guangxi Normal University, Guilin 541004, People's Republic of China*

[d] *Institute of Information Economy, Alibaba Business College, Hangzhou Normal University, Hangzhou 310036, People's Republic of China*

[e] *Department of Physics, University of Fribourg, Fribourg CH-1700, Switzerland*

HIGHLIGHTS

- Ranking influential spreaders is significant for many real problems.
- Weighted LeaderRank outperforms degree centralities.
- Weighted LeaderRank provides more accurate and robust solution than LeaderRank.

Identifying influential spreaders is crucial for understanding and controlling spreading processes on social networks. Via assigning degree-dependent weights onto links associated with the ground node, we proposed a variant to a recent ranking algorithm named Leader-Rank (Lü et al., 2011). According to the simulations on the standard SIR model, the weighted LeaderRank performs better than LeaderRank in three aspects: (i) the ability to find out more influential spreaders; (ii) the higher tolerance to noisy data; and (iii) the higher robustness to intentional attacks.

1. Introduction

The spreading processes of epidemic and information attract increasing attention in complex network studies [1], and researchers tried to find the reason why information spread so quickly [2,3] as well as how to decelerate the spreading [4]. Among many ingredients for quick and wide spreading, influential spreaders play a major role [5–7]. Accordingly, immunization on large-degree nodes (they are usually considered to be more influential) is a highly efficient method to control epidemic spreading [8–10]. It is of great theoretical and practical significance to identify influential spreaders in networks, and similar methods can be applied in ranking scientists [11,12], publications [12], athletes [13] and finding influential directors [14].

How to identify influential spreaders effectively and efficiently is a big challenge up to now. A number of centrality indices have been proposed to address this problem, such as degree centrality, closeness centrality [15], betweenness centrality [16], and eigenvector centrality [17]. Degree centrality is a straightforward and efficient metric but less relevant, because a node

---

* Corresponding author.
   E-mail address: zhutou@ustc.edu (T. Zhou).

having a few highly influential neighbors may be more influential than a node having a larger number of less influential neighbors, while the computation of closeness, betweenness and eigenvector centrality is highly time-consuming and thus usually not feasible for large-scale networks. Some methods have already been proposed in the literature to identify influential spreaders in an effective and efficient way [18–26]. Kitsak et al. [19] proposed a coarse-grained method by using k-core decomposition to quantify a node's influence based on the assumption that nodes in the same shell have similar influence and nodes in higher-level shells are likely to infect more nodes. Zeng et al. [25] found several limitations of the above method and proposed a mixed degree decomposition method, as an improved version of the traditional k-core decomposition. Liu et al. [26] argued that nodes in the same shell may indeed have considerably different spreadabilities, and proposed an improved method that can properly rank nodes in the same shell. Chen et al. [23] devised a semi-local index by considering the next nearest neighborhood, which performs as good as global indices while has much lower computational complexity, and thus obtains a good trade-off on effectiveness and efficiency. In addition, Chen algorithm [23] can well identify influential nodes in a hierarchical tree that cannot be managed by the k-core decomposition. Furthermore, Chen et al. [24] proposed an improved index for directed networks, which accounts for the effects of local clustering and shows better performance.

Recently, Lü et al. [27] proposed the LeaderRank algorithm to identify influential spreaders in directed networks, which is a simple variant of PageRank [28], namely a so-called ground node connected with every other node by a bidirectional link is introduced into the original network, and then the standard random walk process is applied to dig out influential spreaders. Thanks to this simple change, LeaderRank outperforms PageRank in several aspects: (i) LeaderRank converges faster since the network is strongly connected with diameter being only 2; (ii) the influential nodes identified by LeaderRank can spread information faster and wider than those by PageRank; (iii) LeaderRank has higher tolerance to noisy data, and (iv) LeaderRank has higher robustness to intentional attacks. In this paper, we further improve the LeaderRank algorithm by allowing nodes with more fans get more scores from the ground node, that is, replacing the standard random walk by a biased random walk. Experiments on three real social networks (i.e., Delicious, Epinions and Slashdot) show that the weighted LeaderRank can considerably improve the performance of the original LeaderRank.

2. Algorithm

Given a network consisting of N nodes and M directed links, a ground node connected with every node by a bidirectional link is added. Then, the network becomes strongly connected and consists of $N + 1$ nodes and $M + 2N$ links (a bidirectional link is counted as two links with inverse directions). LeaderRank directly applies the standard random walk process to determine the score of every node. Accordingly, if the score of node i at time step t is $s_i(t)$, the dynamics can be described by an iterative process as

$$s_i(t + 1) = \sum_{j=1}^{N+1} \frac{a_{ji}}{k_j^{\text{out}}} s_j(t), \tag{1}$$

where a_{ji} is the element of the corresponding $(N + 1)$-dimensional adjacency matrix, which equals 1 if there is a directed link from j to i and 0 otherwise, and k_j^{out} is the out-degree of node j. The process starts with the initialization where all node scores are 1 and will soon converge to a unique steady state denoted as s_i^{∞} ($i = 1, 2, \ldots, N, N + 1$). LeaderRank ranks all nodes according to s_i^{∞}, and the nodes with larger final scores are considered to be more influential in spreading.

As we have mentioned in the Introduction, LeaderRank outperforms PageRank in several aspects. In this paper, we intend to further improve LeaderRank by introducing a weighted mechanism. Considering a social network where a user i is called a fan of user j if there is a directed link from i to j, namely i could receive information from j and thus j will receive scores from i (if a node's fans are of high influence, this node will be highly influential as well). Obviously, the number of fans (i.e., in-degree) is an important local indicator for a user's influence in spreading. Therefore, based on LeaderRank, we allow nodes with different in-degrees get different scores from the ground node (other possible weighting schemes will be discussed in the last section). Accordingly, the network is described by an $(N + 1)$-dimensional weighted adjacency matrix W. As illustrated in Fig. 1: (i) if $a_{ij} = 0$, then $w_{ij} = 0$; (ii) for any normal node i and the ground node g, $w_{gi} = (k_i^{\text{in}})^{\alpha}$ and $w_{ig} = 1$, where α is a free parameter; (iii) for all other cases, $w_{ij} = 1$. After determining the weight of every link, the dynamics follows a biased random walk [29], namely the score from node j to node i is proportional to the weight w_{ji}:

$$s_i(t + 1) = \sum_{j=1}^{N+1} \frac{w_{ji}}{\sum_{l=1}^{N+1} w_{jl}} s_j(t). \tag{2}$$

Same to LeaderRank, we use final scores in the steady state to quantify nodes' influences.

3. Results

3.1. Data

To validate the effectiveness of the weighted LeaderRank algorithm, we test it on three real social networks: (i) Delicious [30]—a social bookmarking web site, where users could subscribe to leaders' collections as their sources of information;
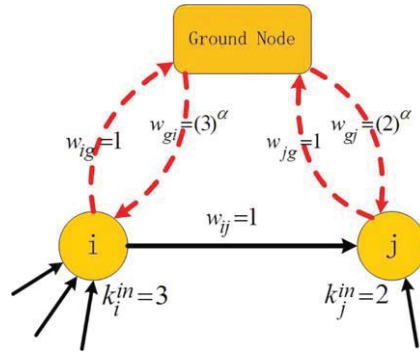
2

Fig. 1. Illustration of the weighted LeaderRank algorithm, where a user having more fans would get more score from the ground node g. The weight of the directed link from the ground node to node i is $w_{gi} = (k_i^{in})^\alpha$, and other weights are all one.

Table 1
Basic statistics of three real networks, including the number of nodes N, the number of links M and the average degree $\langle k \rangle = \langle k^{in} \rangle = \langle k^{out} \rangle$.

Networks	N	M	$\langle k \rangle$
Delicious	582 377	1 686 131	2.90
Epinions	75 879	508 837	6.71
Slashdot	77 360	828 161	10.71

(ii) Epinions [31]—a product review web site, where users could select to trust or distrust the reviews written by others; (iii) Slashdot [32]—a technological news web site, which allows users tag others as friends or foes. Basic statistics of these three real networks are presented in Table 1.

3.2. Comparison with degree centralities

We employ the standard susceptible–infected–removed (SIR) model [33] to estimate the spreading influence of the top-ranked users. In the SIR model, each node can be in one of the three possible states: susceptible, infected and removed. Initially, a single node is set to be infected, and at each time step, each infected node will infect all its susceptible neighbors with probability β and then be removed (dead or recovered with immunity) with probability γ. Without the loss of generality, we set $\gamma = 1$. The dynamical process stops when no infected node is present.

We first compare the spreading processes activated by top-ranked nodes from weighted LeaderRank and the traditional degree centralities. There are three degree centralities for directed networks: in-degree k^{in}, out-degree k^{out} and total degree $k^{tot} = k^{in} + k^{out}$. Taking in-degree as an example. If among the two top-L lists by weighted LeaderRank and in-degree, there are n different nodes, we compare the spreading processed activated by these n different nodes, respectively. For example, if $L = 4$ and for weighted LeaderRank and in-degree, the top-ranked lists are {7, 28, 392, 1146} and {28, 168, 433, 1146}, then $n = 2$ and we will compare the spreading processes with initially infected nodes {7, 392} for weighted LeaderRank and {168, 433} for in-degree, respectively. In the comparison, we fix $\alpha = 1$ since this value is close to the optimal value for all three real networks (see later). Notice that, when β is very small, the disease cannot spread out and the comparison is less meaningful, while when β is very large, almost every individual will get infected, and thus the advantage of our algorithm is not obvious. Only when β lies in the middle, the comparison is meaningful. In the following simulations, we always set the values of β as 0.06, 0.015 and 0.015 for Delicious, Epinions and Slashdot, respectively. We have checked that the qualitative results are not sensitive to β unless β is very large or very small.

Figs. 2–4 respectively report the number of nodes, N_I, having been infected (i.e., infected and removed nodes) as a function of time for Delicious, Epinions and Slashdot. In each figure, we compare the weighted LeaderRank with three degree centralities for three typical lengths of top-ranked lists: $L = 20$, $L = 50$ and $L = 100$. Obviously, in each of the 27 reported cases, the weighted LeaderRank at $\alpha = 1$ outperforms degree centralities.

3.3. Difference between LeaderRank and weighted LeaderRank

Note that, when $\alpha = 0$, the weighted LeaderRank degenerates to the original LeaderRank. Considering the top-L influential nodes respectively produced by LeaderRank and weighted LeaderRank given a specific α, if there are n different nodes in the two lists, we define the *difference* of such two algorithms as n/L. Fig. 5 reports the difference between LeaderRank and Weighted LeaderRank for three real networks, indicating that the top-L list is sensitive to the parameter α. Generally speaking, the larger the absolute value of α is, the bigger the difference is.
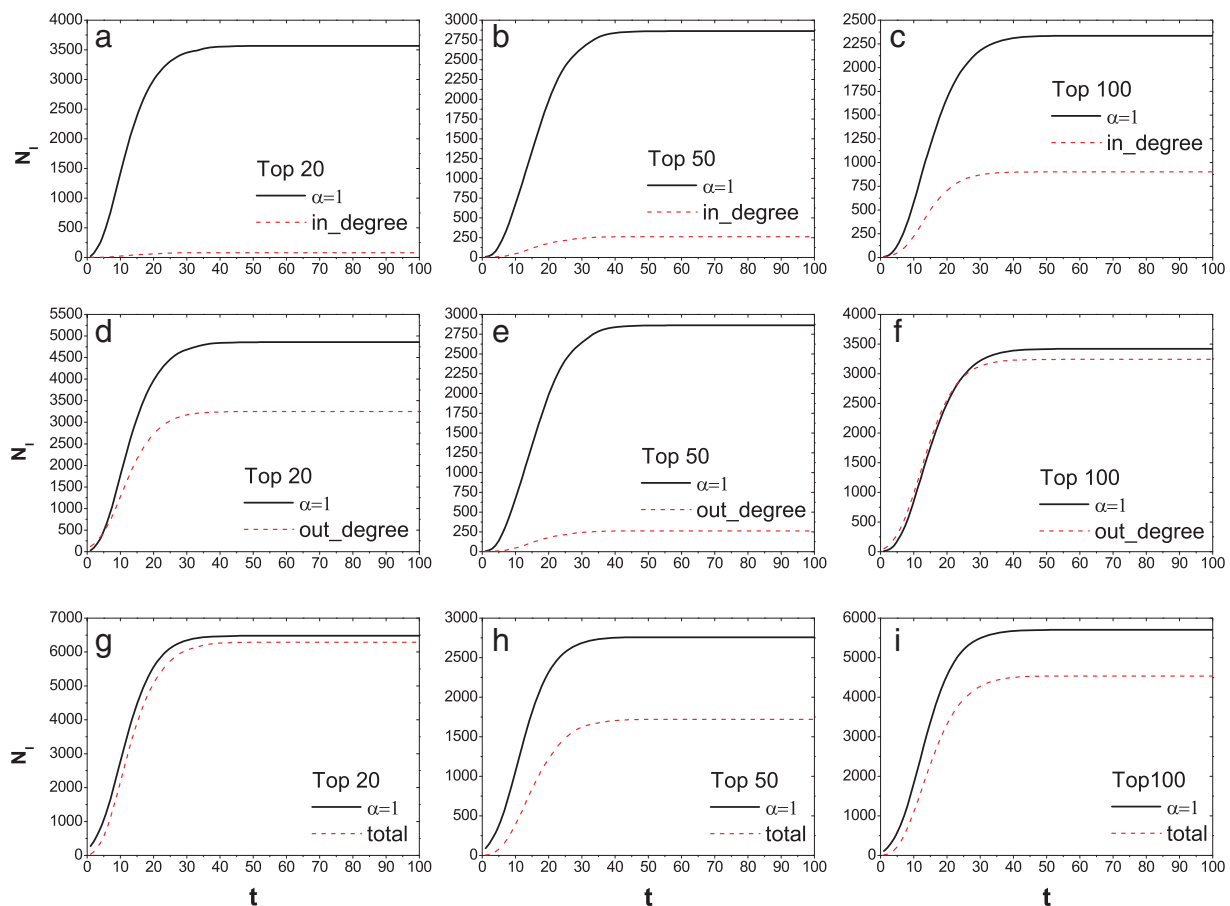
Fig. 2. The number of nodes having been infected as a function of time for Delicious. The three columns, from left to right, are respectively for $L = 20$, $L = 50$ and $L = 100$, while the three cows, from up to down, are respectively for in-degree, out-degree and total degree. The black and red curves denote the results from weighted LeaderRank and degree centralities, respectively. The parameters are set as $\alpha = 1$ and $\beta = 0.06$, and the results are obtained by averaging over 100 independent runs. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.4. Spreadability

Given the network, the spreadability of a node i, denoted by SP_i, is defined as the average number of removed nodes at the steady state if node i is set to be the only initially infected node. In this paper, the spreadability of a given node is obtained by averaging over 100 independent runs.

Given the top-ranked lists by the LeaderRank and the weighted LeaderRank with parameter α, if there are n different nodes, then the average values of spreadability of the n nodes for the weighted LeaderRank and LeaderRank are respectively denoted by $\overline{SP}(\alpha)$ and $\overline{SP}(0)$. Accordingly, we define the relative spreadability as

$$\eta = \frac{\overline{SP}(\alpha) - \overline{SP}(0)}{\overline{SP}(0)}. \tag{3}$$

Clearly, $\eta > 0$ indicates that the weighted LeaderRank performs better than the LeaderRank.

Fig. 6 shows the relative spreadability, which demonstrates that the weighted LeaderRank can considerably improve the spreadability of top-ranked nodes compared to the original LeaderRank, and the optimal performance is around $\alpha = 1$ for the three present cases. Denote $s_i^\infty(\alpha)$ the final score of node i by the weighted LeaderRank with parameter α, then the change of i's score from the original LeaderRank reads

$$\Delta_i = s_i^\infty(\alpha) - s_i^\infty(0). \tag{4}$$

We quantify the correlation between score changes and spreadabilities by the well-known Kendall's Tau [34]. For a pair of nodes i and j, if $SP_i > SP_j$ and $\Delta_i > \Delta_j$ or $SP_i < SP_j$ and $\Delta_i < \Delta_j$, we call this pair is a concordant pair, otherwise it is called a discordant pair. If the numbers of concordant pair and discordant pair are M_{con} and M_{dis}, then the Kendall's Tau is

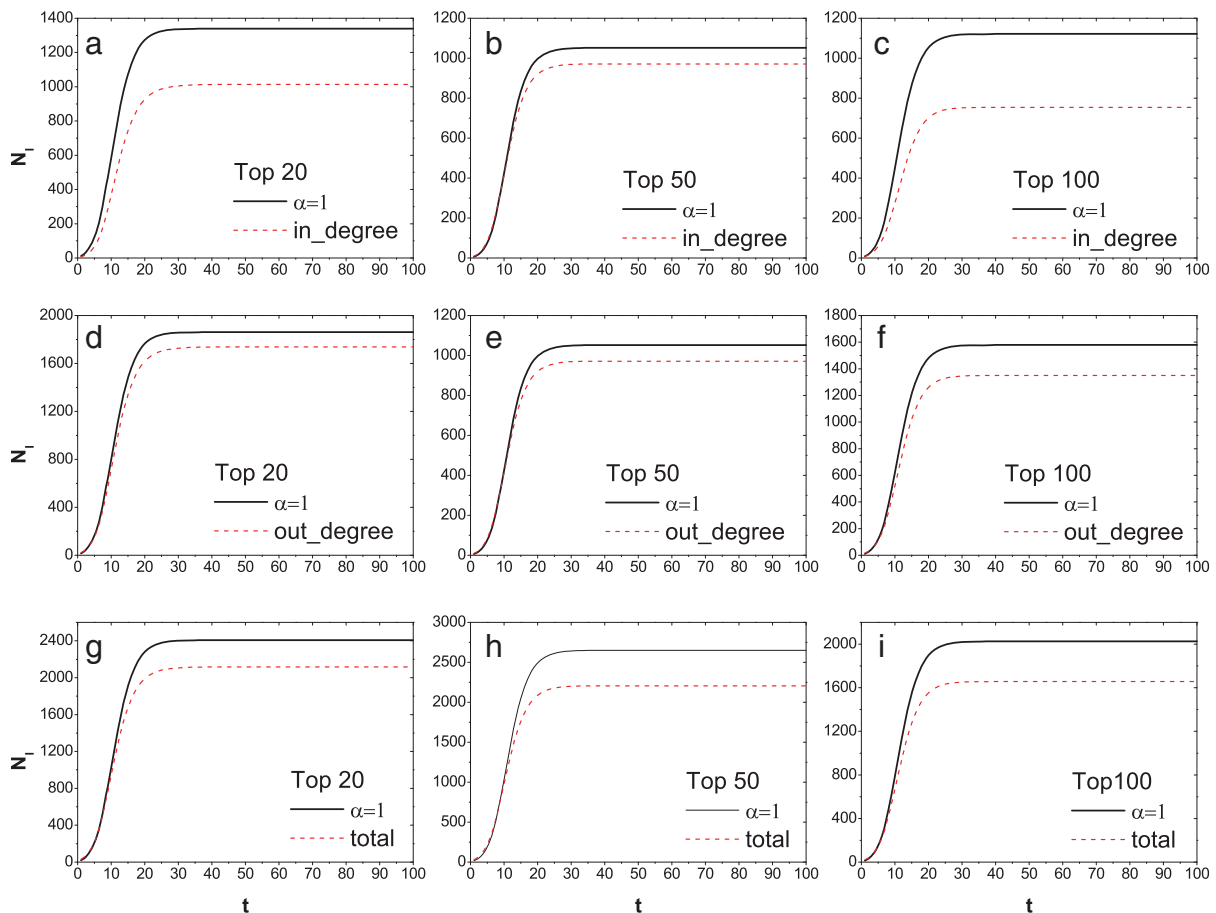$$\tau = \frac{M_{con} - M_{dis}}{\frac{1}{2}N(N - 1)}. \tag{5}$$

4

Fig. 3. The number of nodes having been infected as a function of time for Epinions. The three columns, from left to right, are respectively for $L = 20$, $L = 50$ and $L = 100$, while the three cows, from up to down, are respectively for in-degree, out-degree and total degree. The black and red curves denote the results from weighted LeaderRank and degree centralities, respectively. The parameters are set as $\alpha = 1$ and $\beta = 0.015$, and the results are obtained by averaging over 100 independent runs. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

$\tau > 0$ indicates a positive correlation and $\tau = 1$ is the ideal case where SP_i and Δ_i are completely in the same order. As shown in Fig. 6(b), (e) and (h), the peaks of Kendall's Tau are also around $\alpha = 1$, in accordance with the optimal α for η. Results with higher resolution, as shown in Fig. 6(c), (f) and (i) show the optimal values are $\alpha = 1.1$, $\alpha = 0.8$ and $\alpha = 1.1$, respectively.

3.5. Robustness

Real networks are usually noisy and thus many efforts have been made to detect missing links and remove spurious links [35]. Therefore, a good ranking algorithm should be tolerant to noisy data. Here we randomly add or remove some links from the original networks to see the change of scores, as

$$I_s = \sum_{i=1}^{N} |\tilde{s}_i^\infty - s_i^\infty|, \tag{6}$$

where $\tilde{s}_i^\infty$ is the final score of node i after adding or removing a certain number of links. Of course, I_s increases as the increasing of randomly adding or removing links, as shown in Fig. 7. While comparing with the original LeaderRank, the weighted LeaderRank with $\alpha = 1$ is more tolerant, as indicated by its smaller I_s.

Malicious activities are common in the Internet, in particular in social networks and e-commerce user–product bipartite networks when users manipulate to gain skewed reputation [36], and thus to design robust ranking algorithms against spammers attracts increasing attention recently [37]. Here we consider a representative example called *sybil attack* [38], in which spammers deliberately create fake entities to obtain disproportionately high rank. Under the Sybil attack, a spam user i, with original rank being denoted by r_i, creates v fake fans all pointing to i itself. Then i's new rank is r_i'. Clearly, r_i' should be smaller than r_i. The smaller difference $|r_i' - r_i|$ corresponds to higher robustness.
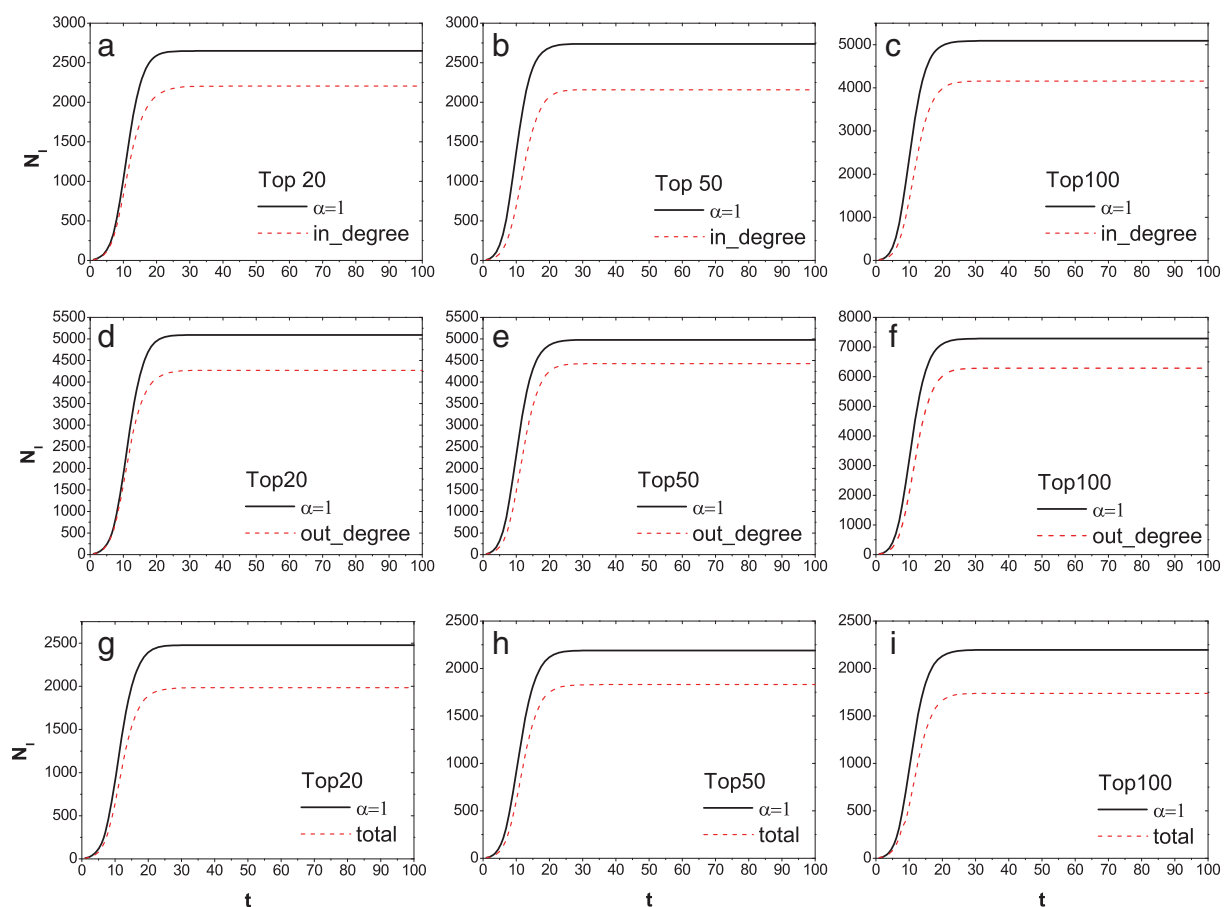
5

Fig. 4. The number of nodes having been infected as a function of time for Slashdot. The three columns, from left to right, are respectively for $L = 20$, $L = 50$ and $L = 100$, while the three cows, from up to down, are respectively for in-degree, out-degree and total degree. The black and red curves denote the results from weighted LeaderRank and degree centralities, respectively. The parameters are set as $\alpha = 1$ and $\beta = 0.015$, and the results are obtained by averaging over 100 independent runs. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
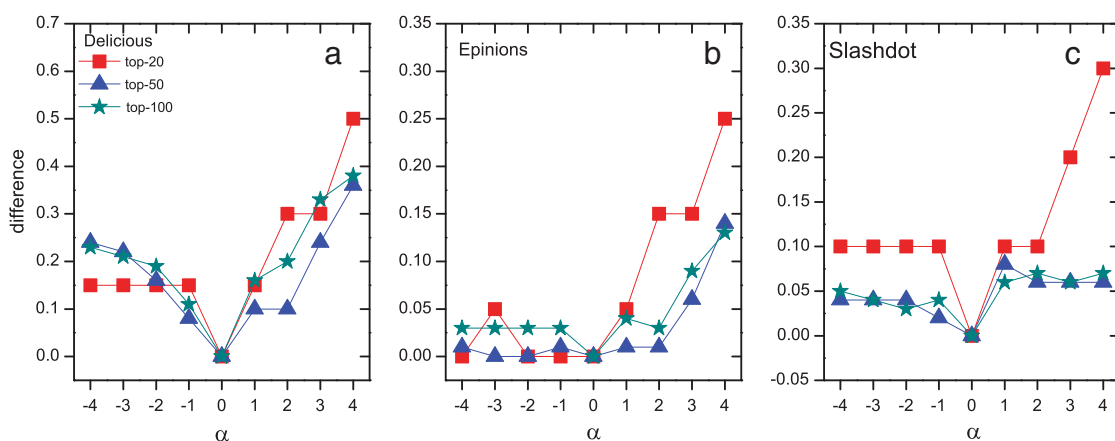

Fig. 5. The fraction of different nodes in top-L ($L = 20, 50, 100$) list of influential nodes between LeaderRank ($\alpha = 0$) and weighted LeaderRank. (a), (b) and (c) display the results for Delicious, Epinions and Slashdot, respectively.

Specifically, each time we consider a spam user who creates $v = 20, 50$ or 100 fake fans and compare the manipulated ranks of this user by LeaderRank and weighted LeaderRank with $\alpha = 1$. Only the top-100 users (i.e., $r_i \in \{1, 2, \ldots, 100\}$) are under investigation. Fig. 8 reports the simulation results, where the horizontal axis shows the original rank of a user, and the vertical axis shows her manipulated rank after the addition of v fake fans. As shown in Fig. 8, the weighted
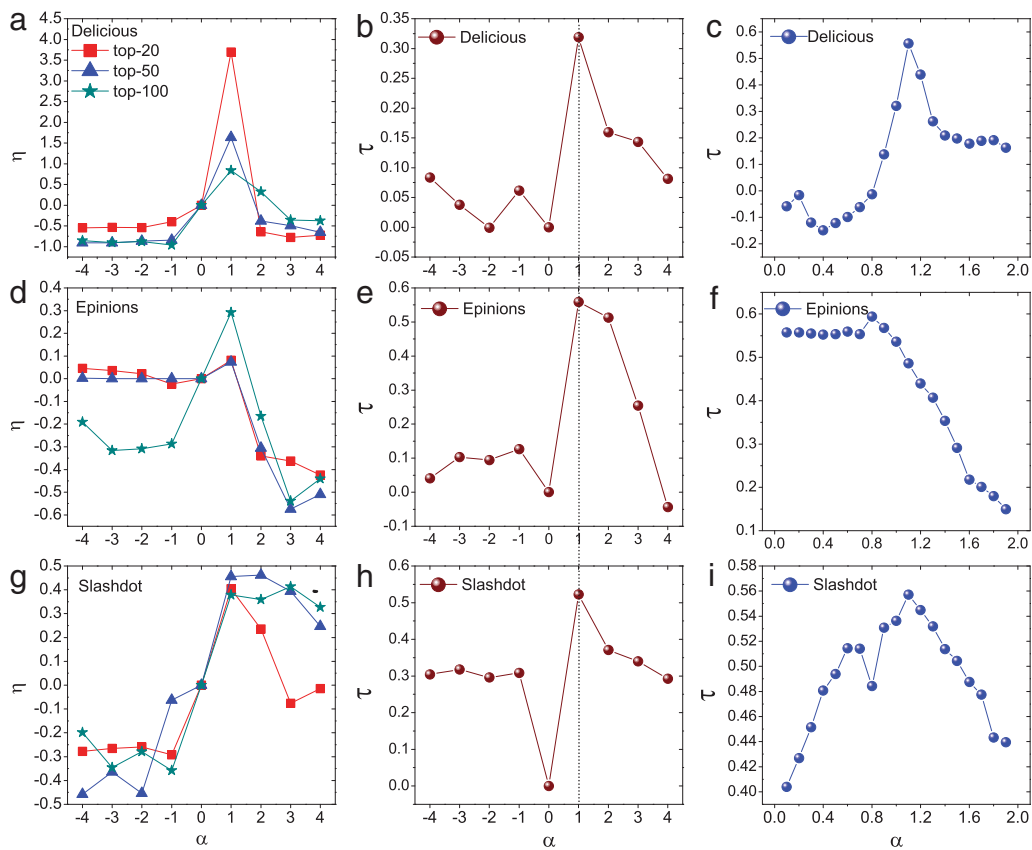
Fig. 6. Relative spreadability at $L = 20$, 50 and 100 for (a) Delicious, (d) Epinions, and (g) Slashdot, with parameters being $\beta = 0.06$, 0.015 and 0.015, respectively. The correlation between the change of nodes' scores and their spreadabilities, quantified by the Kendall's Tau, for (b)(c) Delicious, (e)(f) Epinions, and (h)(i) Slashdot.
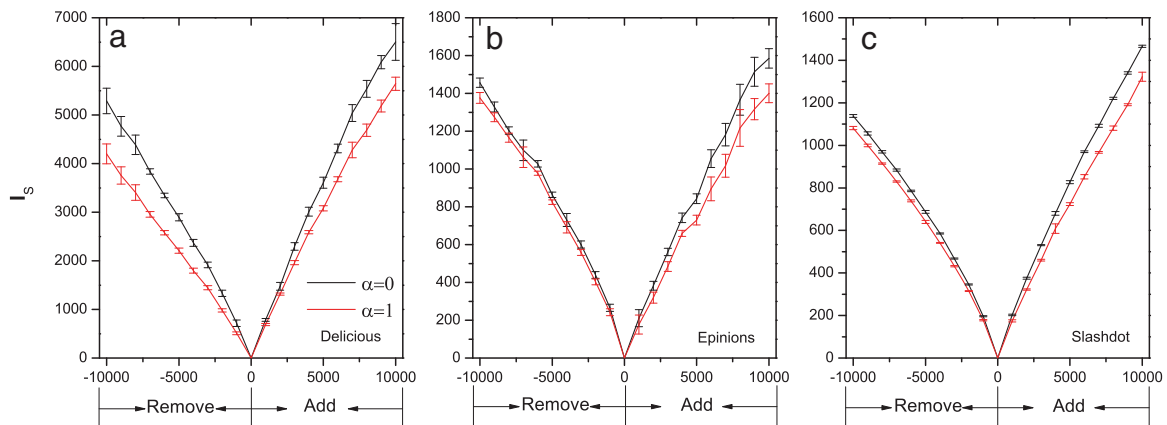

Fig. 7. The change of scores, I_s, as a function of the number of links being randomly added or removed for (a) Delicious, (b) Epinions, and (c) Slashdot. All data points are average values over 100 independent runs with error bars showing the standard deviations.

LeaderRank with $\alpha = 1$ is more robust against sybil attack as the change of ranks is much smaller than that associated with LeaderRank.

4. Conclusion and discussion

LeaderRank may serve as a prototype of ranking algorithms applicable to rank users in social networks. Compared to PageRank, it can find higher influential spreaders with faster convergence, and is more robust to noise and spammers. In
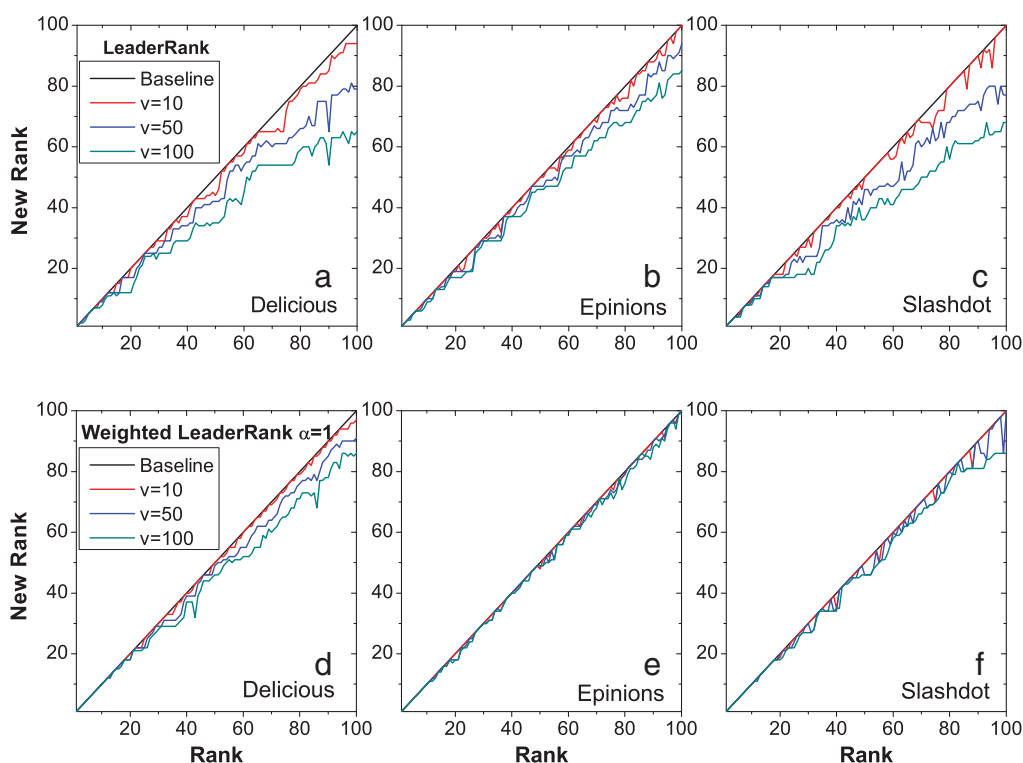
7

Fig. 8. The manipulated rank as obtained by LeaderRank and weighted LeaderRank with $\alpha = 1$, after the addition of v fake fans ($v = 10, 50, 100$) for (a)(d) Delicious, (b)(e) Epinions, and (c)(f) Slashdot.

this paper, we further improve the LeaderRank by allowing nodes with more fans get more scores from the ground node. With almost the same converging speed (we have checked by simulations), this so-called weighted LeaderRank performs better than LeaderRank in three aspects: (i) the ability to find out more influential spreaders; (ii) the higher tolerance to noisy data; and (iii) the higher robustness to intentional attacks.

Since the in-degree of a node (i.e., the number of fans) directly indicates its influence, it is natural to weight nodes according to their influence as applied in this paper. In contrast, it is strange to weight a node according to how many nodes it follows (i.e., the number of its leaders, namely its out-degree). In addition, to weight a node by the number of its leaders is not robust since a node is easy to add as many as leaders. We can also set $w_{gi} = 1$ and $w_{ig} = (k_i^{in})^\alpha$. However, it is also not easy to explain: why a node with more fans needs to give more to the ground node? We have checked that all the three above-mentioned weighting schemes give worse performance than the present one. Using two or more parameters in a more complicated form can give a slightly better performance, but it is very ugly and not easy to be understood.

Acknowledgments

References

[1] T. Zhou, Z.-Q. Fu, B.-H. Wang, Epidemic dynamics on complex networks, Prog. Nat. Sci. 16 (2006) 452–457.
[2] L. Lü, D.-B. Chen, T. Zhou, The small world yields the most effective information spreading, New J. Phys. 13 (2011) 123005.
[3] B. Doerr, M. Fouz, T. Friedrich, Why rumors spread so quickly in social networks, Commun. ACM 55 (2012) 70–75.
[4] M. Schläpfer, L. Buzna, Decelerated spreading in degree-correlated networks, Phys. Rev. E 85 (2012) 015101(R).
[5] S. Aral, D. Walker, Identifying influential and susceptible members of social networks, Science 337 (2012) 337–341.
[6] R.A.P. da Silva, M.P.V. da Viana, L.da F. Costa, Predicting epidemic outbreak from individual features of the spreaders, J. Stat. Mech. (2012) P07005.
[7] J. Borge-Holthoefer, Y. Moreno, Absence of influential spreaders in rumor dynamics, Phys. Rev. E 85 (2012) 026116.
[8] R. Pastor-Satorras, A. Vespignani, Immunization of complex networks, Phys. Rev. E 65 (2002) 036104.
[9] W.-J. Bai, T. Zhou, B.-H. Wang, Immunization of susceptible-infected model on scale-free networks, Physica A 384 (2007) 6560–6662.
[10] L. Hébert-Dufresne, A. Allard, J.G. Young, L.J. Dubé, Global efficiency of local immunization on complex networks, Sci. Rep. 3 (2013) 2171.
[11] F. Radicchi, S. Fortunato, B. Markines, A. Vespignani, Diffusion of scientific credits and the ranking of scientists, Phys. Rev. E 80 (2009) 056103.

[12] Y.-B. Zhou, L. Lü, M. Li, Quantifying the influence of scitntists and their publications: distinguishing between prestige and popularity, New J. Phys. 14 (2012) 033033.
[13] J. Park, M.E.J. Newman, A network-based ranking system for US college football, J. Stat. Mech. (2005) P10014.
[14] X. Huang, L. Vodenska, F. Wang, S. Havlin, H.E. Stanley, Identifying influential directors in the united states corporate governance network, Phys. Rev. E 84 (2011) 046101.
[15] G. Sabidussi, The centrality index of a graph, Psychometrika 31 (1966) 581–603.
[16] L.C. Freeman, Centrality in social networks conceptual clarification, Social Networks 1 (1979) 215–239.
[17] P. Bonacich, Some unique properties of eigenvector centrality, Social Networks 29 (2007) 555–564.
[18] V.V. Kerrebroeck, E. Marinari, Ranking vertices or edges of a network by loops: a new approach, Phys. Rev. Lett. 101 (2008) 098701.
[19] M. Kitsak, L.K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H.E. Stanley, H.A. Makse, Identification of influential spreaders in complex networks, Nat. Phys. 6 (2010) 888–893.
[20] C.H. Comin, L.da F. Costa, Identifying the starting point of a spreading process in complex networks, Phys. Rev. E 84 (2011) 056105.
[21] B. Hou, Y. Yao, D. Liao, Identifying all-around nodes for spreading dynamics in complex networks, Physica A 391 (2012) 4012–4017.
[22] F. Bauer, J.T. Lizier, Identifying influential spreaders and estimating the number of infections in epidemic models: a walk counting approach, Europhys. Lett. EPL 99 (2012) 68007.
[23] D.-B. Chen, L. Lü, M.-S. Shang, Y.-C. Zhang, T. Zhou, Identifying influential nodes in complex networks, Physica A 391 (2012) 1777–1787.
[24] D.-B. Chen, H. Gao, L. Lü, T. Zhou, Identifying influential nodes in large-scale directed networks: the role of clustering, PLoS One 8 (2013) e77455.
[25] A. Zeng, C.-J. Zhang, Ranking spreaders by decomposing complex network, Phys. Lett. A 377 (2013) 1031–1035.
[26] J.-G. Liu, Z.-M. Ren, Q. Guo, Ranking the spreading influence in complex networks, Physica A 392 (2013) 4154–4159.
[27] L. Lü, Y.-C. Zhang, C.H. Yeung, T. Zhou, Leaders in social networks, the delicious case, PLoS One 6 (2011) e21202.
[28] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Comput. Netw. ISDN Syst. 30 (1998) 107–117.
[29] Q. Ou, Y.-D. Jin, T. Zhou, B.-H. Wang, B.-Q. Yin, Power-law strength-degree correlation from resource-allocation dynamics on weighted networks, Phys. Rev. E 75 (2007) 021102.
[30] T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J.R. Wakeling, Y.-C. Zhang, Solving the apparent diversity-accuracy dilemma of recommender systems, Proc. Natl. Acad. Sci. USA 107 (2010) 4511–4515.
[31] R. Guha, R. Kumar, P. Raghavan, A. Tomkins, Propagation of trust and distrust, in: Proceedings of the 13th International Conference on World Wide Web, ACM Press, New York, 2004, pp. 403–412.
[32] V. Gómez, A. Kaltenbrunner, V. López, Statistical analysis of the social network and discussion threads in Slashdot, in: Proceedings of the 17th International Conference on World Wide Web, ACM Press, New York, 2008, pp. 645–654.
[33] R.M. Anderson, R.M. May, B. Anderson, Infectious Diseases of Humans: Dynamics and Control, Oxford University Press, 1992.
[34] M. Kendall, A new measure of rank correlation, Biometrika 30 (1938) 81–89.
[35] L. Lü, T. Zhou, Link prediction in complex networks: a survey, Physica A 390 (2011) 1150–1170.
[36] H. Masum, Y.-C. Zhang, Manifesto for the reputation society, First Monday 9 (2004) 7.
[37] Y.-B. Zhou, T. Lei, T. Zhou, A robust ranking algorithm to spamming, Europhys. Lett. EPL 94 (2011) 48002.
[38] B.N. Levine, C. Shields, B.N. Margolin, A Survey of Solutions to the Sybil Attack, Technical Report of the University of Massachussets Amherst 2006-052, 2006.