Physica A 513 (2019) 297–306

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Identification of influential spreaders in bipartite networks: A singular value decomposition approach

Shuang Xu [a], Pei Wang [b,c,*], Chunxia Zhang [a]

[a] School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China
[b] School of Mathematics and Statistics, Henan University, Kaifeng 475004, China
[c] Institute of Applied Mathematics, Laboratory of Data Analysis Technology, Henan University, Kaifeng 475004, China

HIGHLIGHTS

- We proposed SVD-rank to identify influential spreaders in bipartite networks.
- Inspired by the LeaderRank, we also proposed SVDA-rank.
- Simulations on seven bipartite networks show effectiveness of the algorithms.
- The proposed algorithms are robust to network perturbations.

ARTICLE INFO

Article history:
Received 15 June 2018
Received in revised form 3 August 2018
Available online xxxx

Keywords:
Singular value decomposition
Complex network
Influential spreader
Bipartite network
Important node

ABSTRACT

A bipartite network is a graph that contains two disjoint sets of nodes, such that every edge connects the two node sets. The significance of identifying influential nodes in bipartite networks is highlighted from both theoretical and practical perspectives. By considering the unique feature of bipartite networks, namely, links between the same node set are forbidden, we propose two new algorithms, called SVD-rank and SVDA-rank respectively. In the two algorithms, singular value decomposition (SVD) is performed on the original bipartite network and augmented network (two ground nodes are added). Susceptible–Infected–Recovered (SIR) model is employed to evaluate the performance of the two algorithms. Simulations on seven real-world networks show that the proposed algorithms can well identify influential spreaders in bipartite networks, and the two algorithms are robust to network perturbations. The proposed algorithms may have potential applications in the control of bipartite networks.

1. Introduction

As a special type of undirected complex network, a bipartite graph consist of nodes who can be divided into two disjoint sets X and Y, where connections between the same node set are forbidden. Bipartite networks own a few common, but numerous different features in comparison with general undirected complex networks. The most representative example is the node's clustering coefficient [1], defined as the probability that neighbors of a considered node link to each others. Following this definition, nodes in bipartite networks have zero clustering coefficients, since connections between two nodes from the same node set are forbidden. Thus, researchers have to carefully revise clustering coefficient for bipartite networks in order to cope with this issue [2,3]. Besides, many systems can be modeled as bipartite networks, and increasing attention has been paid to such networks, such as the controllability [4,5], disease analysis [6], personal recommendation [7],

* Corresponding author at: School of Mathematics and Statistics, Henan University, Kaifeng 475004, China.
E-mail address: wangpei@henu.edu.cn (P. Wang).

https://doi.org/10.1016/j.physa.2018.09.005
0378-4371/

Table 1
Notations that are used in this paper.

$\mathbf{A}$	Adjacency matrix
$\overline{\mathbf{A}}$	Augmented network
$\mathbf{A}^x/\mathbf{A}^y$	Weight network of X/Y-class nodes
$\mathbf{B}$	Adjacency matrix of bipartite network
$\overline{\mathbf{B}}$	Augmented bipartite network
N	The number of nodes
N^x/N^y	The number of X/Y-class nodes
M	The number of edges
k	Degree centrality
h	h-index centrality
k_s	Coreness centrality
s^{PR}/s^{LR}	PageRank/LeaderRank centrality
$\mathbf{F}^{PR}/\mathbf{F}^{LR}$	Stochastic matrix of PageRank/LeaderRank
$\mathbf{x}/\mathbf{y}$	SVD(A)-rank centrality
$\mathbf{r}$	Propagation capability

community detection [8] and link prediction [9]. This paper aims to investigate the identification and ranking of highly influential nodes in bipartite networks.

Generally speaking, an immense amount of super infectious node identification methods have been proposed. The H-indices family are simple but effective, including degree [10], H-index [11] and coreness [12]. Furthermore, a plenty of variants were designed to improve accuracy or robustness [13,14] of existing algorithms. As well, through considering standard or biased random walk dynamics on complex networks, some algorithms were designed to identify important nodes, such as PageRank [15] and LeaderRank-family [16–18]. Recently, Maystre and Grossglauser proposed a ChoiceRank model, where a probability inference is made [19]. It shows that the ChoiceRank is better than PageRank. But this model must be trained by clickstream dataset.

Though there are extensive investigations on the identification of influential nodes, few of them have taken the features of bipartite networks into account. Generally speaking, it may make no sense if we rank two different types of nodes at the same time. As an example, the diseasome network consists of disease phenome and disease genome, where an edge represents a gene being related with a disease, but diseases and genes are totally different kinds of objects, and meanwhile, our interest is to identify the important genes rather than diseases [6]. Therefore, ideal methods are supposed to rank the two sets of nodes separately.

Motivated by the above mentioned problems, based on the singular value decomposition (SVD) [20], we introduce an iterative algorithm to identify influential nodes in bipartite networks. Accordingly, we call the new algorithm as SVD-rank. In order to evaluate the effectiveness of SVD-rank, the SIR model [21] is applied to some representative bipartite networks. Compared with traditional methods, the proposed SVD-rank has its advantages. The proposed algorithms provide some new insights to explore the bipartite networks, which may be promised to cope with the coarse-graining [22] and control of bipartite networks. The remainder of the paper is organized as follows. Section 2 gives the descriptions of identification of influential nodes and some existing algorithms. The new algorithms are proposed in Section 3. In Section 4, simulations of the SIR model on seven real-world bipartite networks are performed to evaluate the performance of the proposed algorithms. Conclusions and discussions will be presented in the last Section 5.

2. Problem formulation and existing algorithms

To begin with, we summarize the main notations that will be used in the following sections, which are listed in Table 1.

2.1. Identification of influential spreaders

In some circumstance, influential spreaders are also called important nodes, important spreaders and so on [10–12,23,24]. Node influence can be predicted by well designed algorithms. Given a complex network, according to a proposed algorithm, each node can be assigned a score to evaluate their influence. A node with higher score is more influential. Therefore, identification of influential spreaders is equivalent to rank nodes according to their influential scores. Generally speaking, the ground truth influence of a node is evaluated by simulation of spreading dynamics on such node, such as the SIR model [21]. If the node can infect a large amount of other nodes (spread scope) in the complex network during the spreading, then it is defined as an actually influential spreader. To evaluate the predictive power of a ranking algorithm, some well defined correlation coefficients between the spread scope vector and the algorithm score vector for all nodes of the networks can be employed. If there are high correlation between the spread scope vector and the algorithm score vector, then we say that the algorithm has high prediction power in ranking the actually influence of nodes.

2.2. Traditional centrality measures

For an undirected unweighted network with N nodes and M edges (a bidirectional edge is only counted once), its wiring diagram $\mathbf{A} = (a_{ij})_{N\times N}$ is symmetric, where $a_{ij} = 1$ if node i connects with node j and 0 otherwise. Most of the traditional node ranking algorithms are based on the adjacency matrix $\mathbf{A}$, where some of them are described as follows.

Degree is the simplest centrality measure, which is defined as the number of direct connections,

$$k_i = \sum_{j=1}^{N} a_{ij}. \tag{1}$$

The computation complexity of degree is $O(N)$. Recently, Lü et al. applied *H-index* [11] to rank nodes of complex networks. The H-index of node i equals h_i, the maximum integer such that there are at least h_i neighbors, all of which have degree no less than h_i. If let $(k_1, \ldots, k_{k_i})$ be the degree of neighbors of node i, and without loss of generality, it is ranked in decreasing order, thus, H-index can be computed by

$$h_i = \mathcal{H}\left(k_1, \ldots, k_{k_i}\right) = \max_j \left\{\min(k_j, j)\right\}. \tag{2}$$

For example, given the neighbors' degree (10, 8, 5, 4, 3), the H-index of this node is 3, because $\max\{1, 2, 3, 3, 3\} = 3$. The computation complexity of H-index is $O(N + M)$.

Kitsak et al. [12] suggested that the *coreness* k_s can better reflect node's propagation capability or spreading influence in comparison with node degree and betweenness [12]. The coreness is a score identified by k-core decomposition analysis [25]. At first, we remove all nodes whose degree is less than 1. Now the degree of remained nodes may reduce to 1 and we should continue to remove them until remained nodes are with degrees greater than 1. In step two, remove nodes with degree 2 until remained nodes are with degrees greater than 2. Repeat this operation until all nodes are removed. The nodes removed in step i are assigned with coreness score i. Generally speaking, coreness indicates the node position in the whole network, i.e. at the core (high coreness) or periphery (low coreness). The computation complexity of coreness is $O(N + M)$.

PageRank [15] and *LeaderRank* [16–18] are Markov chain based iterative methods. PageRank mimics the dynamics of surfing the Internet (a biased random walk), and LeaderRank mimics the dynamics of standard random walk. First, set initial score as 1 for all nodes, $s_i(0) = 1 (i = 1, 2, \ldots, N)$. Then, the updating rule of PageRank follows:

$$s_i(t+1)^{PR} = \sum_{j=1}^{N} \frac{a_{ji}}{k_j^{out}} s_j(t)^{PR}, \tag{3}$$

k_j^{out} is the out-degree of node j and t represents iteration step. Clearly, k^{out} equals k in undirected networks. The convergence state $s(t_c)$ is the final PageRank score. But, in this manner, there are m kinds of PageRank scores if this network is not strongly connected and can be partitioned into m strongly connected subnetworks. The above problem can be avoided by the following updating rule

$$s_i(t+1)^{PR} = \frac{1-c}{N} + c\sum_{j=1}^{N} \frac{a_{ji}}{k_j^{out}} s_j(t)^{PR}. \tag{4}$$

Here, c is a damping factor, usually set as 0.85. In fact, Eq. (4) models the behavior of the Internet surfer. When the surfer is browsing page (node) i, he/she randomly clicks a hyperlink on page i with probability c or opens a new page with probability $1-c$. In summary, the probability transition matrix of this process is $\mathbf{F}^{PR}$, where $f_{ij}^{PR} = ca_{ij}/k_j^{out} + (1-c)/N$. Equivalently, in matrix form, $\mathbf{F}^{PR} = c\,\mathrm{diag}(\mathbf{A}\mathbf{1}_N)^{-1}\mathbf{A} + (1-c)\mathbf{1}_N\mathbf{1}_N^T/N$, where $\mathbf{1}_N$ denotes the N-dimensional column vector with all elements being 1. Markov theory shows that the PageRank score of a page is the probability of opening that page after a large number of clicks. In other words, the PageRank score is the stationary distribution of Markov chain $\mathbf{F}^{PR}$, i.e. the principal left eigenvector of $\mathbf{F}^{PR}$.

Recently, Nikolakopoulos et al. [26] revised PageRank for multipartite graphs. In what follows, we call this algorithm as MPR. Let $\mathcal{C}_i$ denotes the class of node i and $|\mathcal{C}_i|$ denotes its cardinality. They constructed the following stochastic matrix

$$\mathbf{F}^{MPR} = c\ \mathrm{diag}(\mathbf{A}\mathbf{1}_N)^{-1}\mathbf{A} + (1-c)\mathbf{M}. \tag{5}$$

Here, $\mathbf{M}$'s (i, j) entry

$$m_{ij} = \begin{cases} |\mathcal{C}_i|^{-1} & , v_j \in \mathcal{C}_i; \\ 0 & , \text{otherwise}. \end{cases} \tag{6}$$

Clearly, different from the PageRank, the surfer goes to a node of the other partite set with probability c (choosing one of his neighbors) and goes to a node of the same partite set with probability $1-c$ (choosing uniformly at random a node of the same partite set). For an example, if webs are organized as a bipartite graph and the surfer is at a Y-class page (node), he/she clicks a hyperlink on this page with probability c and goes to a new Y-class page (node) randomly with probability $1-c$ (although there is no hyperlink from the old Y-class page to the new one). In experimental parts, we set $c = 0.85$, which is a canonical value that is suggested by [26].

Different from the PageRank, the LeaderRank excludes the damping factor and adds a ground node that connects with all other nodes via bidirectional edges, resulting in a strongly connected network with $N+1$ nodes and $M+N$ edges. The new network is called augmented network, whose adjacency matrix is called augmented matrix with the following form

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{1}_N \\ \mathbf{1}_N^T & 0 \end{bmatrix}. \tag{7}$$

The LeaderRank models the standard random walk on this augmented network. The initial score is $s_i(0) = 1 (i = 1, 2, \ldots, N)$ for the ordinary node, and $s_{N+1}(0) = s_g(0) = 0$ for the ground node. The updating rule of the LeaderRank follows:

$$s_i(t+1)^{LR} = \sum_{j=1}^{N+1} \frac{a_{ji}}{k_j^{out}} s_j(t)^{LR}. \tag{8}$$

The LeaderRank score is unique, since this algorithm corresponds to a stochastic process on the strongly connected graph with transition matrix $\mathbf{F}^{LR}$, where $f_{ij}^{LR} = a_{ji}/k_j^{out}$ (Note that $a_{ji} = a_{ij}$ in undirected networks). And the LeaderRank score is the stationary distribution of $\mathbf{F}^{LR}$. Generally speaking, the LeaderRank is applied to assess the node propagation capability when pestilence breaks out. It is shown that the LeaderRank is better than the PageRank. The computation complexity of the LeaderRank, PageRank and MPR is $O(N+M)$.

Morone et al. proposed Collective-Influence (CI) algorithm to identify the minimal set of influencers in networks via optimal percolation [27,28]. Morone et al. stated that the most important nodes in a complex network consist of the minimal set whose removal divides the network into many disconnected and non-extensive components. They casted the issue of influence maximization in complex networks into an optimization problem and showed that the solution is the largest eigenvalue of the non-backtracking matrix (NB) of the network. Then they introduced CI algorithm, where collective influence of nodes is calculated by minimizing the largest eigenvalue of the NB matrix. Recently, Morone et al. proposed the variants of the CI algorithm to further reduce the computational complexity [29].

3. The new algorithms: SVD-rank and SVDA-rank

3.1. Singular value decomposition and SVD-rank

Given a bipartite graph $\mathbf{B} = (b_{ij})_{N_y \times N_x}$, where $b_{ij} = 1$ if nodes y_i and x_j directly connect with each other; $b_{ij} = 0$ otherwise. Since we want to separate X-class from Y-class, at first, we construct two weighted undirected graph $\mathbf{A}^x = \mathbf{B}^T\mathbf{B}$ and $\mathbf{A}^y = \mathbf{B}\mathbf{B}^T$, whose nodes are from the same class. Actually, a_{ij}^x (or a_{ij}^y) counts the number of ways with length 2 from x_i (or y_i) to x_j (or y_j). $\mathbf{A}^x$ and $\mathbf{A}^y$ reflect the connection strength between nodes. For an example, when disease breaks out, node x_i is more likely to infect x_j if a_{ij}^x very high, because there are many paths from x_i to x_j.

As many literatures stated [30–34], the principal eigenvector of an adjacency matrix is a good network centrality measure. Eigenvector centrality takes not only direct connections but indirect connections of all lengths into account. It can naturally adapt to weighted networks [35]. Furthermore, very recently, Xu et al. proofed that the principal eigenvector was just the optimal solution under a proposed probabilistic framework [36]. Therefore, we can also use the principal eigenvector $\mathbf{x}$ and $\mathbf{y}$ of $\mathbf{A}^x$ and $\mathbf{A}^y$ to be the node importance scores respectively. Hereinafter, we call $\mathbf{x}$ and $\mathbf{y}$ as SVD-rank. The following theorem describes the relationship between bipartite graph $\mathbf{B}$ and the SVD-rank.

Theorem 1. *Given a bipartite graph $\mathbf{B} \in \mathbb{R}^{N_y \times N_x}$, the SVD-rank of the two classes of nodes are exactly the left and the right principal singular vectors.*

Proof. Applying the SVD to the bipartite graph $\mathbf{B}$, we have $\mathbf{B} = \mathbf{Y}\mathbf{\Sigma}\mathbf{X}^T$, where $\mathbf{Y} \in \mathbb{R}^{N_y \times N_x}$ is an orthogonal matrix, $\mathbf{\Sigma} \in \mathbb{R}^{N_x \times N_x}$ is diagonal matrix with non-negative real numbers and $\mathbf{X} \in \mathbb{R}^{N_x \times N_x}$ is an orthogonal matrix. The diagonal elements $\sigma_i (i = 1, 2, \ldots, N_x)$ of $\mathbf{\Sigma}$ are the singular values of $\mathbf{B}$. The i'th column of $\mathbf{Y}$ and $\mathbf{X}$ are called the left and the right singular vectors that correspond to σ_i respectively. Without loss of generality, we assume $\sigma_i \geq \sigma_j (i \geq j)$. It is easy to obtain that

$$\mathbf{A}^x = \mathbf{B}^T\mathbf{B} = \mathbf{X}\mathbf{\Sigma}\mathbf{Y}^T\mathbf{Y}\mathbf{\Sigma}\mathbf{X}^T = \mathbf{X}\mathbf{\Sigma}^2\mathbf{X}^T. \tag{9}$$

$$\mathbf{A}^y = \mathbf{B}\mathbf{B}^T = \mathbf{Y}\mathbf{\Sigma}\mathbf{X}^T\mathbf{X}\mathbf{\Sigma}\mathbf{Y}^T = \mathbf{Y}\mathbf{\Sigma}^2\mathbf{Y}^T. \tag{10}$$

Therefore, singular vectors are the eigenvectors of $\mathbf{A}^x$ and $\mathbf{A}^y$, of which the SVD-rank are the 1st columns. This completes the proof. □

Fig. 1(a) shows how to use the SVD to compute the SVD-rank scores. Applying the SVD to $\mathbf{B} = \mathbf{Y}\mathbf{\Sigma}\mathbf{X}^T$, the 1st column of $\mathbf{Y}$ and the 1st row of $\mathbf{X}^T$ are just the SVD-rank.

Inspired by the LeaderRank, we now propose SVDA-rank algorithm. In a network, the ground node is defined as the fictitious node which connects all others via bidirectional edges [16]. In earlier study, though the mathematical mechanisms

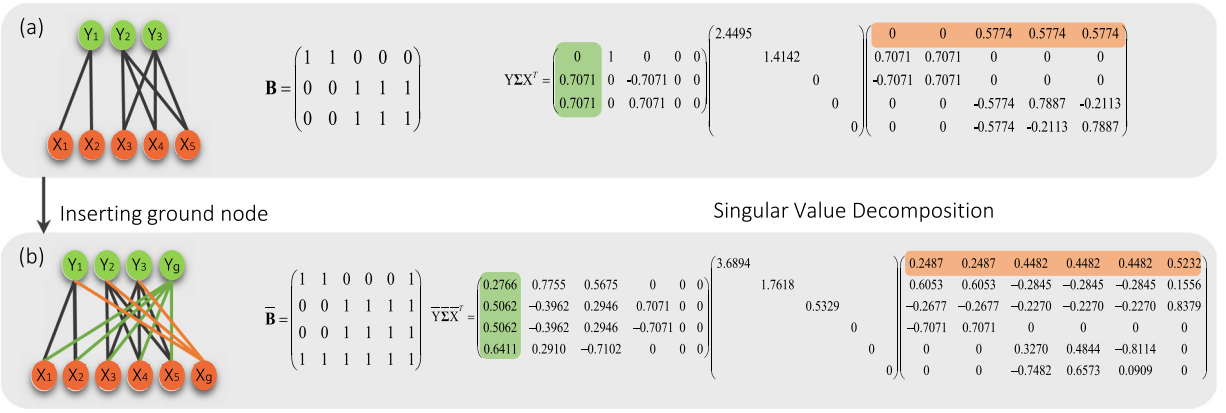

Fig. 1. Illustrations of SVD-rank and SVDA-rank in a toy bipartite network. (a) The original network $\mathbf{B}$ and SVD-rank. (b) The augmented network $\bar{\mathbf{B}}$ and SVDA-rank. The singular vectors corresponding to the principal singular value are shown in orange and green shadows for the X-class nodes and the Y-class nodes respectively. The singular values are shown in gray shadows . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

of ground node are still not clear, plenty of empirical results showed that the ground node is able to enhance the performance of a ranking algorithm [16–18]. Very recently, Xu et al. theoretically illustrated that the existence of ground node could indeed improve node ranking algorithm [36]. Specifically, we add two ground nodes to the bipartite network, node X_g connects with all Y-class nodes and node Y_g connects with all X-class nodes. Consequently, the augmented matrix can be written as

$$\bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} & \mathbf{1}_{N_y} \\ \mathbf{1}_{N_x}^T & 1 \end{bmatrix}. \tag{11}$$

The SVDA-rank applies the SVD to the matrix $\bar{\mathbf{B}}$ instead of the original graph matrix $\mathbf{B}$. Fig. 1(b) depicts how to obtain the SVDA-rank score for a toy network.

3.2. Algorithm

Theorem 1 points out that we are able to obtain SVD-rank/SVDA-rank scores by directly computing the left and the right singular vectors of graph $\mathbf{B}/\bar{\mathbf{B}}$. Albeit they can be defined by matrix factorization, there is no need to perform the SVD to obtain all singular vectors. As a substitution, the principal singular vectors can be computed by the *power iterative method*, which is a fast approximation technique. The algorithm is described as follows:

Algorithm- 1

Input:
The graph $\mathbf{B}$ and convergence tolerance ϵ.
Output:
Importance score $\mathbf{x}$, $\mathbf{y}$.
(1) Initialize importance score $\mathbf{x}(0)$, $\mathbf{y}(0)$; $t \leftarrow 0$; $e \leftarrow 2\epsilon$
(2) Update singular vectors:
while $e > \epsilon$ **do**
(2a) Compute:

$$\begin{cases} \hat{\mathbf{y}}(t+1) \leftarrow \mathbf{B}\mathbf{x}(t), \\ \mathbf{y}(t+1) \leftarrow \dfrac{\hat{\mathbf{y}}(t+1)}{\max \hat{\mathbf{y}}(t+1)}, \\ \hat{\mathbf{x}}(t+1) \leftarrow \mathbf{y}^T(t+1)\mathbf{B}, \\ \mathbf{x}(t+1) \leftarrow \dfrac{\hat{\mathbf{x}}(t+1)}{\max \hat{\mathbf{x}}(t+1)}. \end{cases} \tag{12}$$

(2b) Compute $e = ||\mathbf{y}(t+1) - \mathbf{y}(t)||_2 + ||\mathbf{x}(t+1) - \mathbf{x}(t)||_2$; $t \leftarrow t+1$
end while
(3) $\mathbf{x} \leftarrow \mathbf{x}(t)$, $\mathbf{y} \leftarrow \mathbf{y}(t)$.

Table 2
Basic topological features of the seven real-world networks. N and M denote the number of nodes and edges. N_x and N_y denote the numbers of X-class nodes and Y-class nodes respectively. $\langle k\rangle$ is the average degree. β is the spreading rate and will be used in the simulation of the SIR model. We set $\beta = 1.5\beta_c$, where $\beta_c = \langle k\rangle / \left(\langle k^2\rangle - \langle k\rangle\right)$ is the approximate threshold of epidemic outbreak in undirected networks [21].

Network	N	N_x	N_y	M	$\langle k\rangle$	β
AmeRev [38]	141	5	136	160	2.27	0.068
Leadership [39]	44	24	20	99	4.50	0.294
WikiBooks [40]	30 616	27 732	2 884	67 613	4.42	0.025
WikiNews [40]	26 447	25 039	1 408	68 703	5.20	0.035
SexEsc [41]	16 730	6 624	10 106	35 051	4.67	0.056
Ucforum [42]	1 421	522	899	7 089	4.99	0.081
Diseasome [6]	3 062	1 778	1 284	2 676	1.91	0.338

The SVDA-rank algorithm is the same to Algorithm 1 after replacing $\mathbf{B}$ by $\overline{\mathbf{B}}$. The convergence of Algorithm 1 is guaranteed by the Perron–Frobenius theorem [37] for any initial $\mathbf{x}(0)$ and $\mathbf{y}(0)$ as long as $\mathbf{x}(0), \mathbf{y}(0) \neq \mathbf{0}$. And the convergence is geometric, with ratio $\sqrt{\sigma_2/\sigma_1}$. The computation complexity of Algorithm 1 is $O\left(\min(N_x + M, N_y + M)\right)$.

4. Algorithm performance

4.1. Datasets and statistical features

Seven real-world bipartite networks are considered to evaluate the performance of the proposed algorithms. Basic topological features for the seven networks are summarized in Table 2. The numbers of nodes in the seven networks range from 44 to 30 616 and the amounts of edges range from 99 to 68 703. In the following research, we divide the nodes into two classes (i.e. X and Y) for convenience.

4.2. Kendall τ correlation coefficient

We will use the Kendall τ correlation coefficient to evaluate the performance of an algorithm. Given two vectors $\mathbf{b} = (b_1, \ldots, b_N)^T$ and $\mathbf{a} = (a_1, \ldots, a_N)^T$, the i'th and j'th samples, (a_i, b_i) and (a_j, b_j), are concordant if $(a_i - a_j)(b_i - b_j) > 0$, discordant if $(a_i - a_j)(b_i - b_j) < 0$, and tied otherwise. The Kendall correlation coefficient is defined by

$$\tau = \frac{2(N_c - N_d)}{N(N-1)}, \tag{13}$$

where N_c and N_d denotes the concordant and discordant pairs of samples respectively. Different from the traditional Pearson correlation coefficient, the Kendall τ is popular when we need measure the nonlinear correlation relationships.

4.3. Identifying influential spreaders

As Refs. [10,12,13,17,18,21,34] suggested, the SIR model, an epidemiology model that have been paid extensive attention by both theoretical researchers and practitioners, is employed to evaluate the ground truth of node influence. Specifically, the spread of venereal disease in human sexual network is a typical example. As Fig. 2 illustrated, there are three states in the SIR model. That is, the susceptible, the infected and the removed states. The susceptible node may be infected by its neighbors with probability β. The infected node may be removed with probability γ. This epidemic process terminates when there is no infected node. See [21] for more details about the SIR model. For node i, we define propagation capability r_i as the number of removed nodes when the SIR model terminates, where node i is set as a single infection seed. Propagation capability r_i can reflect the influence of node i. The larger r_i is, the more influential of node i will be.

In the following simulations, we take fixed spreading rate for each network, as shown in Table 2, and we set the removed rates as 1. Considering the randomness, r_i is averaged over 100 independent simulation runs.

Generally speaking, the Kendall τ correlation coefficients between the propagation capability vector $\mathbf{r}$ for all nodes and the algorithm score vectors $\mathbf{x}$, $\mathbf{y}$ stand for the precision of the algorithm. At the same time, we call $\langle\tau\rangle$ (the average correlation coefficient over all networks) as generalized capability of an algorithm. The correlation coefficients for the seven networks are shown in Table 3. On one hand, traditional algorithms expose their defects. There are scarcely any cases being able to prove that traditional algorithms can adapt to bipartite networks. Except for Y-class of WikiBooks and Ucforum, the precision of SVD-rank or SVDA-rank is higher than other algorithms. The generalized capabilities ($\langle\tau\rangle$) of coreness, LeaderRank and PageRank are even inferior to the most simplest degree centrality. On the other hand, Table 3 reports that SVDA-rank significantly enhances the performance of SVD-rank in many cases. Therefore, we conclude that SVD-rank and SVDA-rank can well predict node influence and identify influential spreaders in bipartite networks.

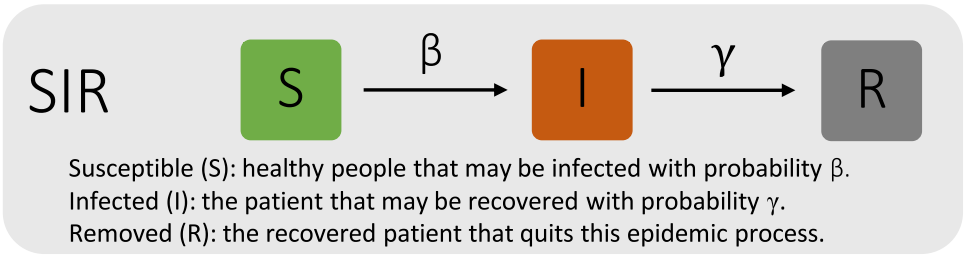

Fig. 2. Illustration of the SIR model.
Source: Figure is adapted from Pastor-Satorras et al. [21].

Table 3
Kendall τ correlation coefficients in the considered networks. k, h and k_s denote degree, H-index and coreness respectively. $\langle\tau\rangle$ denotes the average τ over all networks. The best prediction of each network is shown in bold.

Network (X-class)	k [10]	h [11]	k_s [12]	LeaderRank [16]	PageRank [15]	MPR [26]	SVD-rank	SVDA-rank
AmeRev	**1.0000**	0.2582	0.2582	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
Leadership	0.7700	0.8344	0.7198	0.5548	0.5329	0.5183	0.8175	**0.9197**
WikiBooks	0.6192	0.6252	0.6273	0.3803	0.4179	0.3324	0.8013	**0.8138**
WikiNews	0.6174	0.6207	0.6228	0.6501	0.4092	0.3586	0.7514	**0.7581**
SexEsc	0.6208	0.6316	0.6400	0.6172	0.2941	0.2891	**0.7265**	0.7230
Ucforum	0.8382	0.8857	0.8478	0.7964	0.7383	0.7473	0.8774	**0.9014**
Diseasome	0.3215	0.3516	0.3380	0.5865	−0.2623	−0.2795	**0.7721**	0.7224
Network (Y-class)	k	h	k_s	LeaderRank	PageRank	MPR	SVD-rank	SVDA-rank
AmeRev	0.4555	0.4555	0.4525	0.1183	0.3056	−0.2683	0.7922	**0.8429**
Leadership	0.6947	0.5804	0.6921	0.6947	0.5684	0.5789	0.7789	**0.8526**
WikiBooks	**0.8770**	0.8175	0.8170	0.7803	0.6153	0.5290	0.6205	0.7973
WikiNews	0.4822	0.5173	0.5186	0.3568	0.1484	0.0130	**0.6171**	0.5576
SexEsc	0.4718	0.5089	0.5217	0.2404	0.1314	0.1112	0.7677	**0.7885**
Ucforum	**0.9835**	0.9503	0.9275	0.9400	0.9115	0.9050	0.7774	0.8246
Diseasome	0.4627	0.4745	0.4046	0.3243	−0.0743	−0.0712	**0.7855**	0.7017
$\langle\tau\rangle$	0.6319	0.6349	0.6254	0.5415	0.3643	0.3403	0.7604	**0.7849**

Specifically, compared with the H-index family, our algorithms as well as other Markov chain based methods (i.e. LeaderRank, PageRank and MPR) could not achieve good performance on Y-class of WikiBooks and Ucforum. The reason may be that the local structure, instead of global topology, plays an important role in epidemic process in the two networks.

As to MPR, we find that the performance of MPR is even worse than the PageRank for most cases, although MPR is a modified version of the PageRank for bipartite networks. The reason may be that MPR's assumption does not account for spreading dynamics. MPR may be more suitable for web ranking instead of influential spreader ranking.

4.4. Robustness

A good algorithm should be robust to data noise. To test it, we compare the performance of the proposed algorithms in the original and noisy networks. We apply the algorithms to the original and noisy networks and obtain two ranking lists, denoted by $\mathbf{R}^o$ and $\mathbf{R}^n$ respectively. Here, $\mathbf{R}_i^o$ or $\mathbf{R}_i^n$ equals to m if node i has the m'th largest algorithm score in the original or the noisy networks. To measure the difference of the two ranking lists, an index is introduced as follows,

$$I_R = \sum_{i=1}^{N} |\mathbf{R}_i^o - \mathbf{R}_i^n|. \tag{14}$$

In the following simulations, noise is introduced to the original network by means of stochastically adding or deleting a certain proportion of edges. The number of added or deleted edges is denoted by fM, where f is the proportion and M denotes the number of edges in the original network. We consider cases of $f = \pm 0.1, \ldots, \pm 0.5$. Here, positive f represents edge addition, while negative f denotes edge deletion. f can be interpreted as the size of noise.

An algorithm is more robust if I_R is smaller. Figs. 3 and 4 show the simulation results for X- and Y- class nodes in some of the seven bipartite networks respectively (on account of high computational complexity, we do not consider coreness for WikiBooks, WikiNews, SexEsc). The line plots of I_R versus f reveal that I_R was greater if absolute value of f was larger. On the other hand, deletion of an edge is more likely to result in larger value of I_R. The PageRank and degree are very sensitive to noise since they tend to have greater I_R. While, MPR is more robust in comparison with PageRank. For each case, it shows that the lines for SVD-rank and SVDA-rank are always lower than the other algorithms. Thus, we declare that our algorithms are more robust against topological perturbations. In other words, the proposed algorithms are robust to data noise.

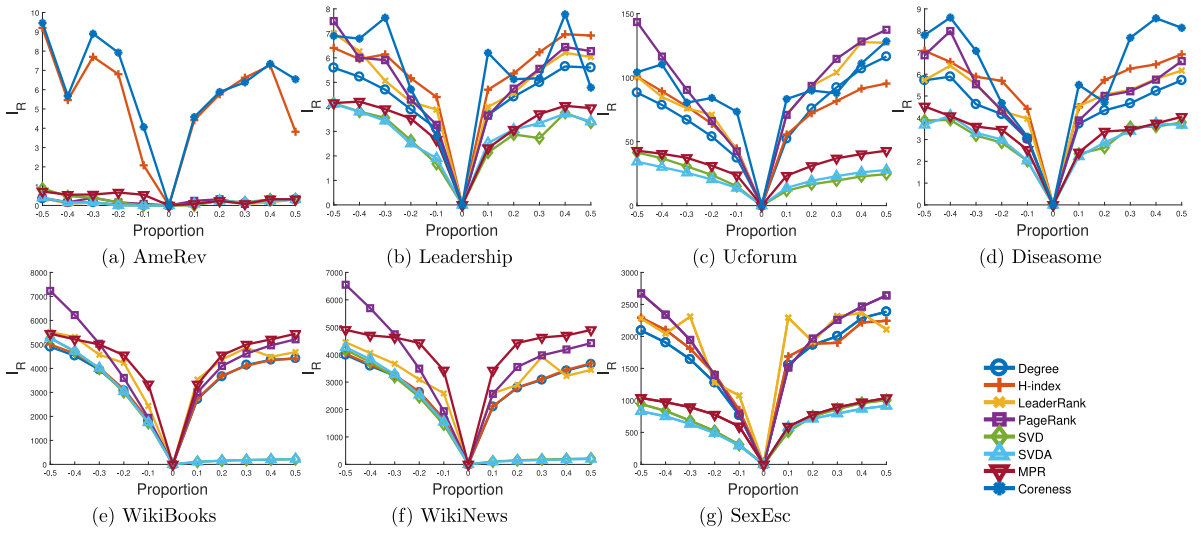

Fig. 3. Robustness of the proposed algorithms against random addition and deletion of edges for the X-class nodes. Results are averaged over 10 simulation runs.

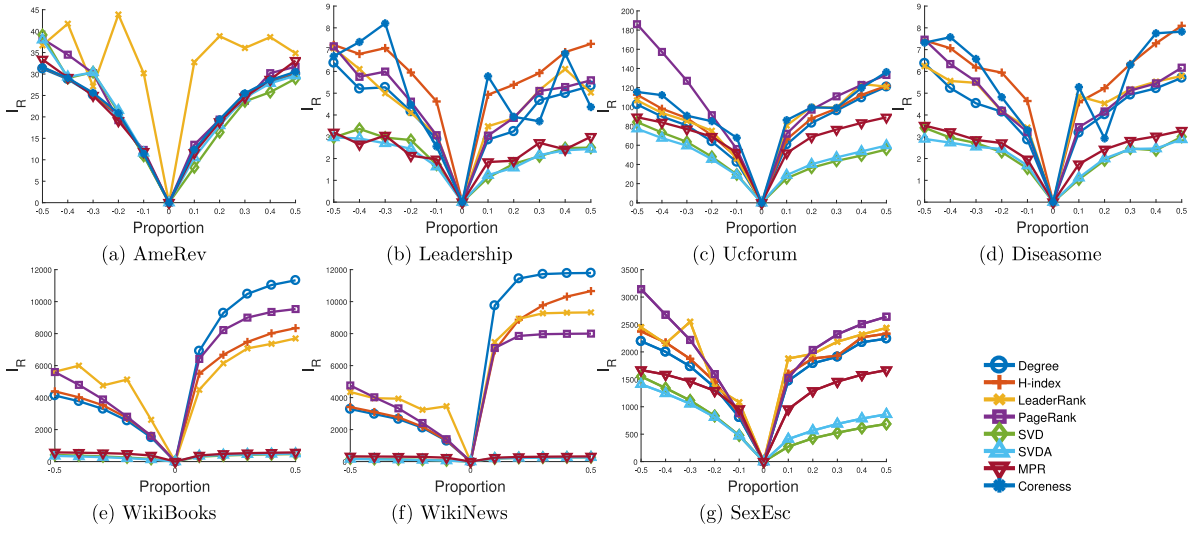

Fig. 4. Robustness of the proposed algorithms against random addition and deletion of edges for the Y-class nodes. Results are averaged over 10 simulation runs.

5. Discussions and conclusions

Identification of influential spreaders in complex networks is an attractive topic, which has attracted extensive investigations [10–26]. Abundant investigations have ignored the particular features of bipartite networks. Part of existing algorithms for undirected networks may lose efficacy in bipartite networks, while some of them did not take the particular features of the bipartite networks into consideration, and they need much computational time. It is therefore urgent to develop effective frameworks to cope with bipartite networks.

This paper establishes two algorithms to identify important nodes in bipartite networks. The SVD, a matrix factorization operation, is introduced to the bipartite matrix **B**, instead of emphasizing adjacency matrix **A** of a network. We prove that the left and right singular vectors correspond to the principal singular value of bipartite matrix **B** can exactly reflect the node importance. The proposed technique is called SVD-rank. This algorithm can be viewed as a special case of the Ing framework [34]. Moreover, two ground nodes, connecting all nodes of the other class, are added, leading to a strongly connected network $\bar{\mathbf{B}}$. One only needs to perform SVD on the augmented matrix $\bar{\mathbf{B}}$ and this operation can be called as SVDA-rank. Compared with traditional methods, including degree, H-index, coreness, PageRank and LeaderRank, the accuracy of

the new algorithms on identifying influential nodes in bipartite networks are quite good, where nodes' actually propagation capability is simulated by the SIR model.

Up to now, our algorithms are only designed for (one layer) bipartite networks. As an anonymous reviewer stated, the duplex or multilayer networks become a hot topic [43–45]. Recently, Wang et al. utilized tensor decomposition to rank nodes in multilayer networks [46]. Note that the proposed SVD-rank and SVDA-rank are actually based on the matrix decomposition technique. In the future, we will investigate how to adapt our algorithms for multilayer networks. Moreover, many problems in biological systems can be described by bipartite networks [47,48]. For example, by integrating differential expression genes in RNA-seq data and their GO annotations, one can construct bipartite networks and explore the identification of crucial responsive genes under certain treatments [48]. Thus, it is also interesting to apply the proposed algorithms to biological systems.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61773153, 61572393 and 11671317). The Key Scientific Research Projects in Colleges and Universities of Henan, China under Grants 17A120002. The Basal Research Fund of Henan University, China (yqpy20140049).

References

[1] D.J. Watts, S.H. Strogatz, Collective dynamics of small world networks, Nature 393 (6684) (1998) 440–442.
[2] P. Zhang, J. Wang, X. Li, M. Li, Z. Di, Y. Fan, Clustering coefficient and community structure of bipartite networks, Physica A 387 (27) (2008) 6869–6875.
[3] P.G. Lind, M.C. González, H.J. Herrmann, Cycles and clustering in bipartite networks, Phys. Rev. E 72 (5) (2005) 056127.
[4] J.C. Nacher, T. Akutsu, Structural controllability of unidirectional bipartite networks, Sci. Rep. 3 (2013) 1647.
[5] P. Wang, D. Wang, J. Lü, Controllability analysis of a gene network for *Arabidopsis thaliana* reveals characteristics of functional gene families, IEEE/ACM Trans. Comput. Biol. Bioinf. (2018) http://dx.doi.org/10.1109/TCBB.2018.2821145.
[6] K.-I. Goh, M.E. Cusick, D. Valle, B. Childs, M. Vidal, A.-L. Barabási, The human disease network, Proc. Natl. Acad. Sci. USA 104 (21) (2007) 8685–8690.
[7] T. Zhou, J. Ren, M. Medo, Y.-C. Zhang, Bipartite network projection and personal recommendation, Phys. Rev. E 76 (4) (2007) 046115.
[8] M.J. Barber, Modularity and community detection in bipartite networks, Phys. Rev. E 76 (6) (2007) 066102.
[9] X. Li, H. Chen, Recommendation as link prediction in bipartite graphs: A graph kernel-based machine learning approach, Decis. Support Syst. 54 (2) (2013) 880–890.
[10] P. Wang, C. Tian, J.-A. Lu, Identifying influential spreaders in artificial complex networks, J. Syst. Sci. Complex. 27 (4) (2014) 650–665.
[11] L. Lü, T. Zhou, Q.-M. Zhang, H.E. Stanley, The h-index of a network node and its relation to degree and coreness, Nature Commun. 7 (2016) 10168.
[12] 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 (11) (2010) 888–893.
[13] A. Namtirtha, A. Dutta, B. Dutta, Identifying influential spreaders in complex networks based on kshell hybrid method, Physica A 499 (2018) 310–324.
[14] F.D. Malliaros, M.E. Rossi, M. Vazirgiannis, Locating influential nodes in complex networks, Sci. Rep. 6 (2016) 19307.
[15] S. Brin, L. Page, Reprint of: The anatomy of a large-scale hypertextual web search engine, Comput. Netw. 56 (18) (2012) 3825–3833.
[16] L. Lü, Y.-C. Zhang, C.H. Yeung, T. Zhou, Leaders in social networks, the delicious case, PLoS One 6 (6) (2011) e21202.
[17] Q. Li, T. Zhou, L. Lü, D. Chen, Identifying influential spreaders by weighted leaderrank, Physica A 404 (2014) 47–55.
[18] S. Xu, P. Wang, Identifying important nodes by adaptive leaderrank, Physica A 469 (2017) 654–664.
[19] L. Maystre, M. Grossglauser, Choicerank: Identifying preferences from node traffic in networks, in: International Conference on Machine Learning(ICML), Sydney, NSW, Australia, 6-11 August 2017, 2017, pp. 2354–2362.
[20] G.W. Stewart, On the early history of the singular value decomposition, SIAM Rev. 35 (4) (1993) 551–566.
[21] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, A. Vespignani, Epidemic processes in complex networks, Rev. Modern Phys. 87 (3) (2015) 925.
[22] S. Xu, P. Wang, Coarse graining of complex networks: A k-means clustering approach, in: 2016 Chinese Control and Decision Conference (CCDC), 2016, pp. 4113–4118.
[23] P. Wang, J. Lü, X. Yu, Identification of important nodes in directed biological networks: a network motif approach, PLoS One 9 (2014) e106132.
[24] P. Wang, X. Yu, J. Lü, Identification and evolution of structurally dominant nodes in protein-protein interaction networks, IEEE Trans. Biomed. Circuits Syst. 8 (1) (2014) 87–97.
[25] S.B. Seidman, Network structure and minimum degree, Social Networks 5 (3) (1983) 269–287.
[26] A.N. Nikolakopoulos, A. Korba, J.D. Garofalakis, Random surfing on multipartite graphs, in: IEEE Inter Conf on Big Data, Washington, DC, USA, 5-8 Dec 2016, 2016, pp. 736–745.
[27] F. Morone, H.A. Makse, Influence maximization in complex networks through optimal percolation, Nature 527 (7579) (2015) 544.
[28] S. Pei, F. Morone, H.A. Makse, Theories for influencer identification in complex networks, in: Complex Spreading Phenomena in Social Systems, Springer, Cham, 2018, pp. 125–148.
[29] F. Morone, B. Min, L. Bo, R. Mari, H.A. Makse, Collective Influence Algorithm to find influencers via optimal percolation in massively large social media, Sci. Rep. 6 (2016) 30062.
[30] P. Bonacich, Factoring and weighting approaches to clique identification, J. Math. Sociol. 2 (1) (1972) 113–120.
[31] P. Bonacich, Power and centrality: A family of measures, Am. J. Sociol. 92 (5) (1987) 1170–1182.
[32] P. Bonacich, Some unique properties of eigenvector centrality, Social Networks 29 (4) (2007) 555–564.
[33] R. Poulin, M.C. Boily, B.R. Msse, Dynamical systems to define centrality in social networks, Social Networks 22 (3) (2000) 187–220.
[34] S. Xu, P. Wang, J. Lü, Iterative neighbour-information gathering for ranking nodes in complex networks, Sci. Rep. 7 (2017) 41321.
[35] X. Qi, E. Fuller, Q. Wu, Y. Wu, C.-Q. Zhang, Laplacian centrality: A new centrality measure for weighted networks, Inf. Sci. 194 (2012) 240–253.
[36] S. Xu, P. Wang, C. Zhang, J. Lü, Spectral learning algorithm reveals propagation capability of complex networks, IEEE Trans. Cybern. (2018) http://dx.doi.org/10.1109/TCYB.2018.2861568, (in press).
[37] J.P. Keener, The Perron–Frobenius theorem and the ranking of football teams, SIAM Rev. 35 (1) (1993) 80–93.
[38] J. Kunegis, American revolution network dataset–KONECT, http://konect.uni-koblenz.de/networks/brunson_revolution (December 2016).
[39] R. Barnes, T. Burkett, International network for social network analysis, Connections 30 (2).
[40] Wikimedia Foundation, Wikimedia downloads, http://dumps.wikimedia.org/ (January 2010).
[41] L.E.C. Rocha, F. Liljeros, P. Holme, Information dynamics shape the sexual networks of Internet-mediated prostitution, Proc. Natl. Acad. Sci. USA 107 (13) (2010) 5706–5711.

[42] T. Opsahl, Triadic closure in two-mode networks: Redefining the global and local clustering coefficients, Social Networks 35 (2) (2013) 159–167.
[43] Y. Li, X. Wu, J.-A. Lu, J. Lü, Synchronizability of duplex networks, IEEE Trans. Circuits Syst. Express Briefs 63 (2) (2017) 206–210.
[44] G. Mei, X. Wu, Y. Wang, M. Hu, J.-A. Lu, G. Chen, Compressive-sensing-based structure identification for multilayer networks, IEEE Trans. Cybern. 48 (2) (2018) 754–764.
[45] X. Wei, X. Wu, S. Chen, J.-A. Lu, G. Chen, Cooperative epidemic spreading on a two-layered interconnected network, SIAM J. Appl. Dyn. Syst. 17 (2) (2018) 1503–1520.
[46] D. Wang, H. Wang, X. Zou, Identifying key nodes in multilayer networks based on tensor decomposition, Chaos 27 (6) (2017) 063108.
[47] Z. Wang, et al., Multi-gene co-transformation can improve comprehensive resistance to abiotic stresses in *B napus L*, Plant Sci. 274 (2018) 410–419.
[48] P. Wang, et al., Transcriptomic basis for drought-resistance in *Brassica napus L*, Sci. Rep. 7 (2017) 40532.