An algorithm based on positive and negative links for community detection in signed networks

Yansen Su1, Bangju Wang1, Fan Cheng1, Lei Zhang1, Xingyi Zhang1 & Linqiang Pan2,3

Community detection problem in networks has received a great deal of attention during the past decade. Most of community detection algorithms took into account only positive links, but they are not suitable for signed networks. In our work, we propose an algorithm based on random walks for community detection in signed networks. Firstly, the local maximum degree node which has a larger degree compared with its neighbors is identified, and the initial communities are detected based on local maximum degree nodes. Then, we calculate a probability for the node to be attracted into a community by positive links based on random walks, as well as a probability for the node to be away from the community on the basis of negative links. If the former probability is larger than the latter, then it is added into a community; otherwise, the node could not be added into any current communities, and a new initial community may be identified. Finally, we use the community optimization method to merge similar communities. The proposed algorithm makes full use of both positive and negative links to enhance its performance. Experimental results on both synthetic and real-world signed networks demonstrate the effectiveness of the proposed algorithm.

Many complex systems in the real world can be modeled as networks1. The networks that include only positive links are called unsigned networks, and the networks with both positive and negative links are called signed networks. Compared with unsigned networks, the links in a signed networks bring more information. Specifically, a positive link in a unsigned network just means a ‘relationship’, while a positive link in signed networks denotes a ‘positive relationship’, and a negative one denotes a ‘negative relationship’. For example, in a signed social network, the relationships between parties may be political alliances and oppositions2. Ferligoj and Kramberger has established the positive links and the negative links to represent the political arrangements with positive and negative ties, respectively2. Besides, there are positive relationships–friendship, trust and like, as well as negative relationships–hostility, mistrust and dislike. In the field of biological science, a gene may be enhanced or repressed by another gene, and the enhanced or repressed relationships could be reflected by the positive or negative links3–6. A protein is likely to be expressed in a subtype of lung cancer, while it is unexpressed in another subtype of lung cancer. The relationships between proteins and the subtypes of lung cancer could also be reflected by positive and negative links7. Recently, Kunegis et al. showed that taking the positive and negative links into consideration could help to find more useful information compared with the only analysis of positive links8.

Community detection problem has attracted increasing attention since it was first proposed by Girvan and Newman9. Most of these community detection methods can only handle the networks without negative links, i.e. unsigned networks8–13. In an unsigned network, communities are defined as the groups of nodes in which links are dense, while between which are less dense. Unlike the definition above, the communities in signed networks are defined as the groups of nodes in which positive links are dense and between which negative links are also dense. That is, community detection methods in unsigned networks focus merely on link density but not the signs of links as their clustering attributes. However, the communities in signed networks depend on not only the density of links but also the signs of links. Thus, previous community detection algorithms in unsigned networks are not suitable for the community detection problem in signed networks. In view of the importance of

1Key Lab of Intelligent Computing and Signal Processing of Ministry of Education, School of Computer Science and Technology, Anhui University, Hefei, 230039, China. 2Key Laboratory of Image Processing and Intelligent Control, School of Automation, Huazhong University of Science and Technology, Wuhan, 430074, China. 3School of Electric and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou, 450002, Henan, China. Correspondence and requests for materials should be addressed to X.Z. (email: xyzhanghust@gmail.com) or L.P. (email: lqpan@mail.hust.edu.cn)
signed networks, community detection methods in signed networks need to be developed. The challenge of the community detection problem in signed networks is that the community structure is ambiguous since there are some negative links within communities and some positive links between communities. In the face of the challenge, researchers have put forward lots of community structure detection algorithms to get the best partition of signed networks.

Several algorithms have been extended from the community detection algorithm in unsigned networks to solve community detection problem in signed networks20. Yang et al. first proposed the FEC algorithm to detect communities from signed networks based on random walk. Subsequently, several two-stage clustering algorithms have been proposed21–23. For instance, the community modularity values are respectively calculated by positive and negative links, and the communities are evaluated by the combination of these two community modularity values21. The GN-H algorithm is the combination of GN and hierarchical clustering algorithm to detect communities in signed networks22. Specifically, it uses the GN algorithm to detect communities based on the positive links, and then combine the negative links to get the final hierarchical clustering results. However, in these two-stage clustering algorithms, the latter stage is always affected by the previous stage, which may limit the performance of the algorithms. Liu et al. first proposed the community detection problem as a multiobject problem (MOP), but the proposed objective functions still need further optimization and improvement to enhance its performance24. Majority of previous researches mainly use positive links for community detection, and negative links are only used for adjustment. In fact, positive links attract a node to be in a community, while the node is rejected outside by negative links. Negative links have no less information than positive links. Thus, further study is needed to make full use of both positive and negative links for community detection in signed networks.

In our work, we propose a random walk-based algorithm named SRWA for community detection in signed networks based on positive and negative links. The overall framework of SRWA is to detect initial communities in a signed network, and then expand these initial communities by application of random walks. Firstly, a dense subgraph is detected based on the nodes, whose degree is larger than that of its neighbours. Then, the initial community is growing by adding the node which is more likely to be attracted into the community than to be rejected from the community step by step. Specifically, a node which is not in a current community has a positive probability to be in the community and a negative probability to be away from the initial community. The positive probability is compared with the negative one to judge whether the node should be added into the community. If a node could not be added into current communities, then a new initial community may be developed. Experimental results on both synthetic and real-world signed networks show the feasibility and effectiveness of the proposed algorithm.

Results
In this section, we present the comparative results of the proposed algorithm and the representative algorithms, i.e., FEC20, MEAs-SN24 and a method to optimize the modularity based on Tabu search which is implemented by Radatool (Tabu search for short)21, 25, on both real-world and synthetic signed networks.

Real-world and synthetic signed networks. Real-world signed networks. The first real social network is the U.S. supreme court justices network, which describes the voting behavior of nine justices in the supreme court of the United States during the period of 2006–200726. The positive line means that one justice supports the other one, and the negative line indicates the opposite meaning. Its community structure is shown in Fig. 1. We can see that the U.S. supreme court justices network is divided into two communities.

The Slovene parliamentary party network represents the relationships among ten parties of the Slovene parliament in 19942. Positive links mean that the parliament activities of two parities are similar, while negative links mean that their activities are dissimilar. Figure 2 shows the topological structure of the Slovene parliamentary party network and its community structure.

The Gahuku-Gama subtribes network reflects the political alliances and oppositions among 16 Gahuku-Gama subtribes, which are distributed in a particular area and are involved in warfare with each other27. Positive and negative links represent the political arrangements with positive and negative ties, respectively. Its community structure can be seen in Fig. 3.
The Sampson monastery network represents the social relationships between 18 monks in the monastery of New England\(^28\). Sampson collected four kinds of social relationships among a group of monks, i.e., friendship, esteem, influence and sanction. Each type of relationship has both positive and negative aspects. Six variants of the Sampson monastery network can be obtained from UCINET IV datasets and each variant consists of 18 nodes, however, the numbers of positive and negative links are different in these variants. The information about the six variants of the Sampson monastery network are described in Table 1\(^29\). All these variants have three communities due to the fact that 18 monks were divided into three groups, i.e., Young Turks, Outcasts, and Loyal Opposition\(^29\).

The microarray expression data for the construction of a gene network used in the study originated from the Gene Expression Omnibus (GEO) with the accession number GSE23400 (http://www.ncbi.nlm.nih.gov/). There are 52 samples and each sample contains expression data of 54,675 probes, which are associated to genes according to the information of GPL570 (a microarray chip). According to the number of genes that a probe detects, probes can be classified into three categories: probes detecting a single gene, probes detecting more than one gene, and probes detecting no genes. We performed the removal of probes which could not detect any genes in each sample, and calculated the expression value of each gene which could be detected by more than one probe. In addition, we calculated the Pearson correlation coefficients of two genes based on their expression data. If the
Table 2. Information of LFR benchmark signed networks. ‘N’ represents the number of nodes in a network; ‘k’ denotes the average degree of nodes; μ is the fraction of links that each node shares with nodes in other communities; P− is the fraction of negative links within communities, while P+ is the fraction of positive links between communities.

| Group name | N   | k   | μ  | P−  | P+  |
|------------|-----|-----|----|-----|-----|
| Group 1    | 128 | 16  | 0.1–0.5 | 0.0–0.8 | 0.0–0.6 |
| Group 2    | 500 | 10  | 0.3–0.5 | 0.1–0.5 | 0.1–0.3 |
| Group 3    | 1000 | 10 | 0.3–0.5 | 0.1–0.5 | 0.1–0.3 |

Table 3. The values of NMI and Qsigned on real-world networks. ‘SCJ,’ ‘SPP,’ and ‘GGS’ represent the U.S. supreme court justices network, the Slovene parliamentary party network, the Gahuku-Gama subtribes network, respectively. SAM-AFF4 and SAM-INFL are two variants of the Sampson monastery network. ‘GIN’ denotes the gene-gene interaction network. ‘Tabu’ represents Tabu search. ‘–’ means a null value.

| Networks | NMI | Qsigned | NMI | Qsigned |
|----------|-----|---------|-----|---------|
| SCJ      | 1   | 1       | 0.4050 | 0.4050 |
| SPP      | 0.8572 | 0.8483 | 0.4086 | 0.4022 |
| GGS      | 0.9143 | 0.9022 | 0.3870 | 0.3779 |
| SAM-AFF4 | 0.7007 | 0.7492 | 0.5296 | 0.7448 |
| SAM-INFL | 0.5296 | 0.7448 | 0.1460 | 0.1839 |
| GIN      | —   | —       | 0.2220 | 0.1827 |

Pearson correlation coefficient between gene1 and gene2 is larger than 0.8 or smaller than −0.8, then a positive link or a negative link is considered between gene1 and gene2. A positive link between gene1 and gene2 denotes that gene1 and gene2 are positively related, and a negative link means that they have a negative correlation. Then, a gene-gene interaction network (GIN, for short) is constructed, including 658 nodes and 3338 links, where 2774 are positive links and 564 are negative links.

Synthetic signed networks. In this work, we extended the Lancichinetti-Fortunato-Radicchi (LFR, for short) benchmark to signed networks[30]. A signed network generator is designed with an unsigned network generator and a program to control the type of links in an unsigned network[31]. The signed network generator is denoted as SRN(n, k, maxk, t1, t2, minc, maxc, om, on, μ, P−, P+). Here, N is the number of nodes in a network; k and maxk are the average and maximum degree of nodes; t1 and t2 are the exponents for the degree and community size distribution; minc and maxc are the minimum and maximum community size; on and om are the number of overlapping nodes and the number of memberships of overlapping nodes. More importantly, μ is the fraction of links that each node shares with nodes in other communities, which controls the cohesiveness of the communities in the generated SRNs. The higher the value of μ is, the more ambiguous the community structure is. P− is the fraction of negative links within communities, while P+ is the fraction of positive links between communities. Ideally, negative links should be between communities and positive links should be within communities. Thus, P− and P+ are two parameters to adjust the noise level. When the value of μ is fixed, the larger the values of P− and P+ are, the more ambiguous the community structure is. That is, given a fixed μ, we can control the noise level by adjusting both P− and P+.

Comparison with other algorithms. We verify the performance of the proposed algorithm (SRWA) by comparing it with three representative algorithms (FEC, MEAs-SN, and Tabu search) on both real-world and synthetic signed networks.

Comparison on real-world signed networks. As can be seen in Table 3, the proposed algorithm could generate the true partition results on the networks (e.g., the U.S. supreme court justices network, the Slovene parliamentary party network, the Gahuku-Gama subtribes network, and two variants (i.e., SAM-AFF4 and SAM-INFL) of the Sampson monastery networks). Besides, the obtained NMI and Qsigned values were almost larger than those of other algorithms.

We also examined the performance of the proposed algorithm on the gene-gene interaction network, the truth partition of which is unknown. Although the Qsigned value of the proposed SRWA (i.e., 0.2901) was smaller than that of Tabu search (i.e., 0.4577) on the gene-gene interaction network, the communities achieved by SRWA seem...
to be more reasonable than those obtained by Tabu search and other compared algorithms. To be specific, on the
gene-gene interaction network, SRWA detected 41 communities, among which 11 communities were confirmed to be related to certain biological processes by the database for annotation, visualization and integrated discovery (DAVID for short, https://david.ncifcrf.gov/summary.jsp) (see Table 4). For example, a community detected by SRWA contains seven nodes, which represent the genes ANKH, RP4-758J24.5, MIR6741, DNAJC30, NEIL2, NSMAF and XRN2, respectively. Interestingly, above seven genes are all phosphoproteins, which are bound to phosphoric acid. In addition, the other ten communities detected by SRWA are corresponding to the following biological functions: membrane, alternative splicing, splice variant, protein binding, signal peptide, sequence variant, splice variant and cytoplasm. Here, we refer to a community which is confirmed to be related to a biological process by DAVID as an effective community. The ratio of the effective communities to all communities detected by SRWA is 0.268. However, the ratios of the effective communities to all communities detected by the compared algorithms (FEC, MEAs-SN and Tabu search) are respectively 0.017, 0.004 and 0.022, which are smaller than that by SRWA. Therefore, the SRWA performed better than other compared algorithms on the gene-gene interaction network.

**Comparison on synthetic signed networks.** All algorithms are tested on three groups of synthetic signed networks. A total of 30 independent runs are conducted for each algorithm and the average results are shown.

| Algorithms | FEC | MEAs-SN | Tabu | SRWA |
|------------|-----|---------|------|------|
| $N_e$ | 227 | 416 | 217 | 41 |
| $N_v$ | 4 | 2 | 6 | 11 |
| $N_s$ | 0.017 | 0.004 | 0.022 | 0.268 |

Table 4. The effective communities on the gene-gene interaction networks. ‘$N_e$’ represents the total number of detected communities. ‘$N_v$’ denotes the number of effective communities among all detected communities. ‘Tabu’ represents Tabu search.
Figure 4. Comparison between SRWA and other algorithms on synthetic signed networks with 128 nodes.

Figure 5. Comparison between SRWA and other algorithms on synthetic signed networks with 500 and 1000 nodes.
the node and the members of a community. We have tested the performance of SRWA, and compared it with other representative algorithms (FEC, MEAs-SN and Tabu search) on both real-world and synthetic signed networks. The experimental results have demonstrated the feasibility and effectiveness of SRWA. The feature of the proposed algorithm could be summarized as follows. (1) SRWA has a good ability to detect communities on signed networks. Several other algorithms have good performances on small-scale networks with clear community structures, however, their detection results are far from the expectation on large-scale networks with unclear community structure. The proposed SRWA shows its superiority over the competing approaches for detecting communities in signed networks with unclear community structures in terms of the quality of found communities. (2) SRWA is not sensitive to the initial nodes and it needs not any prior knowledge on the community structure.

In our future work, we will focus on how we can use the SRWA approach to further address problems in other related domains such as disease module mining. So far, the work about disease module mining considers a biological network as a large graph including only positive links. However, the relations among the entities of the biological network are complex, which could not be modeled only by positive links. From such a signed biological network, we may discover some previously unknown information. In addition, it is also interesting to investigate bio-inspired computing models for community detection in complex networks, such as probe machine and spiking neural P system.

**Methods**

A signed network can be abstracted as a graph \( SN = (V_{SN}, E^p, E^n) \), where \( V_{SN} = \{ v_1, v_2, \ldots, v_n \} \) is the set of nodes in the network, \( E^p \) is the set of positive links and \( E^n \) is the set of negative links. The graph could be expressed as an adjacency matrix \( A \), where the element \( a(i,j) \) represents the type of the link between the nodes \( v_i \) and \( v_j \) (i.e., \( < v_i, v_j > \)). Specifically, if the link between the nodes \( v_i \) and \( v_j \) is positive, then \( a_{ij} = 1 \); if the link between \( v_i \) and \( v_j \) is negative, then \( a_{ij} = -1 \); if there is no relationships, then \( a_{ij} = 0 \).

The community detection in signed networks is to detect the communities in which the links are positive and between which the links are negative. Let \( C = \{ C_1, C_2, \ldots, C_m \} \) be a set of communities in a signed network. The community detection problem in the signed network can be described as: \( a_{ij} = 1, (v_i \in C_l) \land (v_j \in C_m) ; a_{ij} = -1, (v_i \in C_l) \land (v_j \in C_m) \land (l \neq k) \).

The proposed algorithm aims to make full use of both positive and negative links to detect communities in a signed network. The overall framework of SRWA is presented in Table 5, which consists of three main steps: (1) the initial communities are detected; (2) the initial communities are expanded based on random walks; (3) a procedure for community optimization is performed. In what follows, we introduce the details of SRWA.

**Detecting initial communities in signed networks.** The node with a large impact in a network always has a large number of neighbours. The importance of a node could be reflected by the node degree, which is the sum of the positive degree and the absolute value of the negative degree (Eq. 1).

\[
deg(v) = deg_p(v) + |deg_n(v)|, 
\]

where \( deg(v) \) represents the node degree, and \( deg_p(v) \) and \( deg_n(v) \) are the positive degree and the negative degree of the node, respectively. Specifically, if the degree of a node is larger than those of its neighbours, then the node is more likely to be a center of a community than its neighbours. The local maximum degree node is defined as the node which has a larger degree compared with its neighbours. The way to discover the local maximum degree nodes was referred to the previous work. In this work, we identify the local maximum degree nodes from all nodes in a signed network based on node degrees.

Here, a initial community in a signed network is defined as a dense subgraph, which includes a local maximum degree node, as well as its close neighbours. Given a local maximum degree node \( node_1 \), we identify its neighbour nodes \( (node_2) \) with the largest positive degree. The reason why the positive degree of \( node_2 \) is used to identify \( node_2 \) is that, as members of initial communities, \( node_1 \) and \( node_2 \) should be linked closely by positive links. \( node_1 \) and \( node_2 \) may have a common neighbour node \( (node_3) \), which is also detected based on positive links. A initial community is comprised by the nodes \( node_1 \), \( node_2 \) and \( node_3 \), together with the links among them.

**Expanding communities.** Let \( Y = \{ Y_k | k = 1, \ldots, q \} \) be the set of all communities, where \( q \) is the number of the communities, \( Y_k = (V_k, E^p_k, E^n_k) \) is the \( k^{th} \) community, \( V_k \) is the set of nodes in the community, \( E^p_k \) and \( E^n_k \) are respectively the set of positive and negative links in the community. Specifically, in the initial situation, \( Y_k (k = 1, \ldots, q) \) is an initial community.

Let the walker start from a node \( u \), which is not belong to any current communities. Then, the node \( u \) could teleport to current communities with the basis of the connections of nodes. The total probability theorem and conditioning probability model are used to calculate the positive probability of the node \( u \) teleporting to a community based on positive links (i.e., \( p^+ (u \rightarrow Y), (k = 1, \ldots, q) \)), as well as the negative probability which represents the node is away from the community based on negative links (i.e., \( p^- (u \rightarrow Y), (k = 1, \ldots, q) \)). If the positive probability of the node \( u \) teleporting to a community is larger than the negative probability of \( u \) being away from the community, then \( u \) may be added into the community; otherwise, it is in current communities, which implies that a new initial community should be formed.

There are \( q \) initial communities, so we perform \( q \) runs of random walks to calculate \( p^+ (u \rightarrow Y) \) and \( p^- (u \rightarrow Y) \). At the \( k^{th} \) run of random walks, it is supposed that \( u \) belongs to the \( k^{th} \) community. The graph of the \( k^{th} \) random walk process is...
\[ G_k = (V_k', E_k', E_k^N), \]  

where \( V_k' = \bigcup_{i=1}^{l} V_i \cup \{u\}, E_k' = (\bigcup_{i=1}^{l} E_i') \cup \{(u, v) \mid v \in V_k' \} \) and \( E_k^N = E_k^N \).

First, we calculate the positive and negative probability of the walker teleporting from \( u \) to the node \( v \) by Eq. 3.

\[ \text{Similar}^+(v_i, v_j) = \frac{|\Gamma^+_{v_i} \cap \Gamma^+_{v_j}|}{|\Gamma^+_{v_i}| \cup |\Gamma^+_{v_j}|}, \]  

where \( \Gamma^+_{v_i} (\Gamma^+_{v_j}) \) is the positive neighborhood of \( v_i (v_j) \), the member of which is connected with \( v_i (v_j) \) by a positive link, and \(|x|\) indicates the cardinality (i.e., number of elements) in the set \( x \). Let \( v \) be \( u \) in Eq. 3. \( \text{Similar}^+(v_i, u) \) represents the positive similarity between \( u \) and \( v_i \) \((1 \leq i \leq m)\), and it is also denoted as \( \text{Similar}^+(v_i) \) for short.

Let the matrix \( M^+ \) be the normalization of the positive similarity between nodes in the \( k^{th} \) community. That is, \( M^+(i, j) = \frac{\text{Similar}^+(v_i, v_j)}{\sum_j \text{Similar}^+(v_i, v_j)} \). Here, \( M^+ \) could be considered as the transition matrix of a random walker. Suppose

| Input                                      | Signed network \( SN \) |
|--------------------------------------------|-------------------------|
| Output                                     | Community set \( Y \) |

| Step 1                                      | Calculate the node degree of each node in \( SN \) by Eq. 1; |
|                                            | Find the node which has a larger degree compared with its neighbors, and put it in set \( H \); |
|                                            | \( \forall \) Each node in \( H \) is a local maximum degree node. |
| Step 2                                      | For each node \( v \), in set \( H \), discover initial communities \( Y^v \); |
|                                            | Put the elements of \( Y^v \) in set \( Y \); |
|                                            | \( \forall \) \( Y \) is the set of initial communities. |
| Step 3                                      | Merge the initial communities which are identical in \( Y \); |
|                                            | Return \( Y \) as the set of initial communities; |
|                                            | Put all nodes in initial communities in \( V \); |
|                                            | Put the rest nodes which are not in initial communities in \( U \); |
|                                            | \( \forall \) \( Y \) is the set of nodes which are in initial communities. |
| Step 4                                      | For each node \( u \) in \( U \) and an initial community \( Y_k \) in \( Y \), calculate \( P^+(u \rightarrow Y_k) \) and \( P^-(u \rightarrow Y_k) \) by Algorithm 1. |
|                                            | \( \forall \) \( Y_k \) is the \( k^{th} \) community in \( Y \). |
|                                            | \( \forall \) \( P^+(u \rightarrow Y_k) \) means the positive probability of \( u \) belonging to \( Y_k \). |
|                                            | \( \forall \) \( P^-(u \rightarrow Y_k) \) means the negative probability which represents \( u \) is away from \( Y_k \). |
| Step 5                                      | Compare \( P^+(u \rightarrow Y_k) \) with \( P^-(u \rightarrow Y_k) \); |
|                                            | If \( P^+(u \rightarrow Y_k) > P^-(u \rightarrow Y_k) \), then put \( u \) in \( \text{Temp} \) number. |
|                                            | \( \forall \) \( \text{Temp} \) number is used to storage the number of initial community which is likely to contain \( u \). |
| Step 6                                      | If \( \text{Temp-number} = 0 \), then add \( u \) to the community \( Y_{\text{same}} \) that results in the largest positive probability; |
|                                            | \( \forall \) \( P^+(u \rightarrow Y_{\text{same}}) = \max \{P^+(u \rightarrow Y) \mid |Y| \in \text{Temp-number}\} \). |
|                                            | Delete \( u \), form \( U \). |
| Step 7                                      | If \( \text{Temp-number} = 0 \), for \( u \), discover initial communities \( Y^u \); |
|                                            | If \( \text{Temp-number} = 0 \), \( Y = Y \cup Y^u \); |
|                                            | Put all nodes included in \( Y^u \) in \( V \); |
|                                            | Put the rest nodes which are not included in \( Y^u \) into \( U \). |
| Step 8                                      | Repeat step 4–7 until there is no node left in \( U \). |
| Step 9                                      | Merge the communities which are identical or similar in \( Y \) by Eq. 14; |
|                                            | Return \( Y \) as the set of communities. |

Table 5. The overall framework of SRWA.
the positive probability of the walker teleporting from \( u \) to \( v_i \) is \( s^+_t(i) \) at the time \( t \). Particularly, in the initial situation, the positive probability of the walker teleporting from \( u \) to \( v_i \) is the normalization of the positive similarity between \( u \) and \( v_i \), i.e., \( s^+_0(i) = \frac{\text{Similar}(u,v_i)}{\sum_v \text{Similar}(u,v)} \). At the time \( t+1 \), the positive probability \( s^+_{t+1} \) is calculated as follows.

\[
s^+_{t+1} = (1 - \alpha) \cdot (M^+)^T \cdot s^+_t + \alpha \cdot d,
\]

where \((M^+)^T\) is the transpose of the normalization of the positive similarity matrix \( M^+ \), and the \( i \)-th entry \( s^+_{t+1}(i) \) captures the positive probability of the walker teleporting from \( u \) to \( v_i \) at the time \( t+1 \).

Iterate the Eq. 4 until \( s^+ \) is convergent. Suppose when the iteration has been completed, the stable state is \( \pi^+ = (\pi^+_1, \ldots, \pi^+_m) \), then \( \pi^+ \) satisfies \( \pi^+ = (1 - \alpha) \cdot (M^+)^T \cdot \pi^+ + \alpha \cdot d \). In this situation, the \( i \)-th entry of \( \pi^+ \) denotes the conditional positive probability that the node \( u \) teleports to \( v_i \) when \( u \) belongs to the \( k \)-th community.

Similarly, we calculate the negative similarity based on negative links by Eq. 5.

\[
\text{Similar}^-(v_i,v_j) = \frac{|\Gamma^-_{v_i} \cap \Gamma^-_{v_j}|}{|\Gamma^-_{v_i}| + |\Gamma^-_{v_j}|},
\]

where \( \Gamma^-_{v_i} (\Gamma^-_{v_j}) \) is the negative neighborhood of \( v_i (v_j) \), the member of which is connected with \( v_i (v_j) \) by a negative link.

The negative similarities between nodes are normalized to get the transition matrix \( \Gamma^- \). Suppose \( s^- \) represents the conditional probability that \( u \) is away from \( v_i \) when \( u \) belongs to the \( k \)-th community at the time \( t \). We also calculate the initial negative probability vector \( s^-_0 \), the \( i \)-th entry of which is the normalization of the negative similarity between \( v_i \) and \( u \), i.e., \( s^-_0(i) = \frac{\text{Similar}^-(u,v_i)}{\sum_v \text{Similar}^-(u,v)} \). Then, \( s^-_{t+1} \) could be calculated by Eq. 6.

\[
s^-_{t+1} = (1 - \alpha) \cdot (M^-)^T \cdot s^-_t + \alpha \cdot d.
\]

Iterate the Eq. 6. When the iteration has been completed, \( \pi^- = (\pi^-_1, \ldots, \pi^-_m) \) denotes the stable state, where \( \pi^-_i \) represents the conditional negative probability that \( u \) is away from \( v_i \) when \( u \) belongs to the \( k \)-th community.

Next, the node \( u \) has an average conditional positive probability \( p^+(u \to Y_j|u \in G_k) \) to teleport to a community \( Y_j \) when \( u \) is connected to the nodes in the \( k \)-th community. Specifically, \( p^+(u \to Y_j|u \in G_k) \) is the mean value of the conditional probabilities and represents \( u \) teleports to all nodes in \( Y_j \) in the graph \( G_k \) (Eq. 7).

\[
p^+(u \to Y_j|u \in G_k) = \text{mean}[\pi^+_i | v_i \in Y_j],
\]

where \( V_j \) is the node set of the community \( Y_j \).

Similarly, \( u \) also has an average conditional negative probability \( p^-(u \to Y_j|u \in G_k) \) to be away from \( Y_j \) when \( u \) is connected to the nodes in the \( k \)-th community (Eq. 8).

\[
p^-(u \to Y_j|u \in G_k) = \text{mean}[\pi^-_i | v_i \in V_j],
\]

The probability that \( u \) belongs to the \( k \)-th community is based on the positive similarity between \( u \) and a node in the \( k \)-th community, which is calculated as Eq. 9. We also calculate the probability that \( u \) does not belong to the \( k \)-th community as Eq. 10.

\[
p^+(u \in G_k) = \text{avg}[\text{Similar}^+(u,v)| \forall v_i \in V_k],
\]

\[
p^-(u \in G_k) = \text{avg}[\text{Similar}^-(u,v)| \forall v_i \in V_k].
\]

Finally, the positive probability for the node \( u \) to teleport to or the negative probability for \( u \) to be away from a community \( Y_j \) is calculated based on the theorem of total probability by Eqs 11 and 12.

\[
p^+(u \to Y_j) = \sum_{k=1}^{K} [p^+(u \to Y_j|u \in G_k) \times p^+(u \in G_k)],
\]

\[
p^-(u \to Y_j) = \sum_{k=1}^{K} [p^-(u \to Y_j|u \in G_k) \times p^-(u \in G_k)],
\]

where \( p^+(u \to Y_j|u \in G_k) \) is the average conditional positive probability for \( u \) teleporting to the community \( Y_j \) when \( u \) is connected to the nodes in the \( k \)-th community, while \( p^-(u \to Y_j|u \in G_k) \) is the average conditional negative probability for \( u \) being away from to \( Y_j \) on the same condition.

The algorithm to calculate the positive and negative probability of a node belonging to each community is described in Table 6. If a node is more likely to be in a community than to be away from the community, then it will be added into the community. Otherwise, it could not be added into any current communities. In this situation, the node could be considered as a new important node, and a new initial community which includes the new important node as well as its close neighbours may be detected. If a new initial community has been detected, then the number of the current communities plus one, and the above procedures are repeated to add nodes into
Table 6. Algorithm 1.

| Input | Node-set V; node u; community Yc. |
|-------|----------------------------------|
| Output | P^+ (u \rightarrow Y_c) means the positive probability of u belonging to Y_c; P^- (u \rightarrow Y_c) means the negative probability which represents u is away from Y_c. |
| Step 1 | Construct the graph G_k by Eq. 2. |
|       | *G_k represents the graph of the k^th random walk process.* |
| Step 2 | Calculate the positive similarity matrix Similar^+ and the negative similarity matrix Similar^- by Eqs 3 and 5; Normalize both Similar^+ and Similar^- to obtain M^+ and M^-.* |
|       | *M^+ and M^- are respectively the transition matrix based on positive and negative links.* |
| Step 3 | Calculate s_p and s_q. |
|       | *s_p and s_q represent the normalization of the positive and negative similarity between v_i \in V and u.* |
| Step 4 | Calculate d = \begin{pmatrix} \frac{1}{d_1}, \frac{1}{d_2}, \ldots, \frac{1}{d_n} \end{pmatrix}. Let \alpha = 0.15. |
|       | *d is the transition probability vector and \alpha is the teleporting probability.* |
| Step 5 | Iterate the Eq. 4 until s_i^\alpha is convergent, and let \pi^\alpha to be the convergent s_i^\alpha.* |
|       | Iterate the Eq. 6 until s_i^\alpha is convergent, and let \pi^\alpha to be the convergent s_i^\alpha.* |
|       | *s_i^\alpha means the positive probability of the walker teleporting from u to v at the time t.* |
|       | *s_i^\alpha means the negative probability that u is away from v_i at the time t.* |
|       | *\pi^\alpha means the positive probability of the walker teleporting from u to v_i.* |
|       | *\pi^\alpha means the negative probability that u is away from v_i.* |
| Step 6 | Calculate p^+(u \rightarrow Y_i | u \in G_k) by Eq. 7; |
|       | Calculate p^-(u \rightarrow Y_i | u \in G_k) by Eq. 8. |
|       | *p^+(u \rightarrow Y_i | u \in G_k) and p^-(u \rightarrow Y_i | u \in G_k) denote an average conditional positive and negative probability that u teleports to a community Y_i when u is connected to the nodes in the k^th community.* |
| Step 7 | Calculate p^+(u \in G_k) by Eq. 9; |
|       | Calculate p^-(u \in G_k) by Eq. 10. |
|       | *p^+(u \in G_k) means the positive probability that u is connected to the nodes in the k^th community.* |
|       | *p^-(u \in G_k) means the negative probability that u is connected to the nodes in the k^th community.* |
| Step 8 | Calculate p^+(u \rightarrow Y_j) = p^+(u \rightarrow Y_j | u \in G_k \times p^+(u \in G_k) and p^-(u \rightarrow Y_j) = p^-(u \rightarrow Y_j | u \in G_k \times p^-(u \in G_k). |

**Community optimization.** Two or more communities may have a large number of common nodes. That is, these communities may be identical or similar. In this case, the expanded communities should be merged into one community. If communities C_i and C_j satisfy the following formula, then they can be merged into a larger community C_i^\& j:

\[
\frac{|C_i \cap C_j|}{\min(|C_i|, |C_j|)} > \xi,
\]

where \(\xi\) is a threshold. Let \(\xi = 0.5\), meaning that most members of the small community are in the large community, the two communities can be merged into one.

**Time complexity.** The proposed algorithm takes a time complexity of \(O(dN)\) to find local maximum degree nodes in a network, where \(d\) is the average degree of nodes, and \(N\) is the number of nodes in the network. At the stage of detecting initial communities, the time used to detect initial communities based on local maximum degree nodes is \(O(d^2 p)\), where \(p\) is the number of local maximum degree nodes, and \(d^2\) is the average positive degree of nodes. In initial situation, there are \(p\) initial communities at most. At the stage of expanding communities, it needs to calculate the probability that a node teleports to each node in communities based on an iterative formula. It takes a time complexity of \(O(logn)\) in each iteration as stated in ref. 39, where \(n\) is the number of nodes in the communities. The worst-case complexity for iteration is \(O(logN)\). A small number of nodes (i.e., \(h\)) which are not in any communities is either in a new community, or to be added to a community based on the tightness. It takes a time complexity of \(O(p + d)\) to judge whether a node is in a new community. If a node is in a...
new community, then the number of initial communities plus one. In the worst case, there are \( p + h \) communities in the stage. Otherwise, it takes the time complexity of \( O((p + h)h) \) to calculate the tightness between a node and a community. The time complexity of the stage after \( p + h \) iterations is \( O((p + h) \log N) + O((p + h)h) \). At the stage of community optimization, it takes a time complexity of \( O((p + h)h) \) to judge whether two communities should be merged. Therefore, the time complexity of the entire algorithm is \( O((d + p + h)N) \), since \( O(d^2 p) < O(dN) \), \( O(p \log N) < O(pN) \), \( O(h \log N) < O(hN) \) \( O((p + h)h) < O(pN) + O(hN) \) and \( O((p + h)^2) < O((p + h)N) \).

### Evaluation measures.

Normalized Mutual Information \( NMI \)\(^{14} \) and the extended modularity \( Q \) \( Q_{\text{signed}} \)\(^{21} \) are widely used indexes for measuring the performance of community detection algorithms in signed networks. Both of them reflect the detection results from different points of view. Thus, both \( NMI \) and \( Q_{\text{signed}} \) are employed here as indexes to test the detection results.

\[
NMI(P_R, P_F) = \frac{-2\sum_{XX} X_{ij} \log \left( \frac{X_{N}X_{N}}{X_{ij}X_{ij}} \right)}{\sum_{XX} X_{ij} \log \left( \frac{X_{N}}{X_{ij}} \right) + \sum_{XX} X_{ij} \log \left( \frac{X_{N}}{X_{ij}} \right)}
\]

(15)

where \( P_R \) and \( P_F \) respectively represent the community partition result obtained by an algorithm and the real community partition; \( N \) is the number of nodes; \( X \) is a \( 2 \times 2 \) matrix, and \( X_{ij} \) is the number of nodes from the real community \( i \) that also belong to the found community \( j \); \( X_{ij} = X_{ij}^+ + X_{ij}^-; X_{ij}^+ = X_{ij}^+ + X_{ij}^- \). If the partitioning result \( P_F \) is the same as \( P_R \), then \( NMI(P_R, P_F) = 1 \); if they are completely opposite, then \( NMI(P_R, P_F) = 0 \).

\[
Q_{\text{signed}} = \frac{1}{2w^+ + 2w^-} \sum_{i,j} w_{ij} \left( \frac{w_{ij}^+ w_{ij}^-}{2w^+ - 2w^-} \right) \delta(C_i, C_j)
\]

(16)

where \( w_{ij} \) is the weight of adjacency matrix, \( w_{ij}^+ (w_{ij}^-) \) denotes the sum of all positive weights of node \( v_i(v_j) \), and \( w_{ij}^+ (w_{ij}^-) \) denotes the sum of all negative weights of node \( v_i(v_j) \). \( w^+ (w^-) \) represents the total positive (negative) strength of the SN, and \( C_i (C_j) \) represents the community which node \( v_i (v_j) \) belongs to, and \( \delta(C_i, C_j) \) is 1 if nodes \( v_i \) and \( v_j \) are in same community; otherwise \( \delta(C_i, C_j) \) is 0.

### References

1. Liu, X., Stanley, H. E. & Gao, J. Breakdown of interdependent directed networks. *Proceedings of the National Academy of Sciences of the United States of America* 113 (2016).
2. Ferligoj, A. & Kramberger, A. An analysis of the slovene parliamentary parties network. *Developments in Statistics and Methodology* 209–216 (1996).
3. Bowers, P. M., Cokus, S. J., Eisenberg, D. & Yeates, T. O. Use of logic relationships to decipher protein network organization. *Briefings in Bioinformatics* 16, 6399–6420 (2015).
4. Friggeri, A., Chelius, G. & Fleury, E. Egomunities exploring socially cohesive person-based communities. Tech. Rep., INRIA, Reading, United Kingdom (2011).
5. Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008, P10008 (2008).
6. Zeng, X., Zhang, X., Liao, Y. & Pan, L. Prediction and validation of association between micrornas and diseases by multipath methods. *Biochimica Et Biophysica Acta* 1860, 2735–2739 (2016).
7. Su, Y. & Pan, L. Identification of logic relationships between genes and subtypes of non-small cell lung cancer. *PLoS One* 9, e94644 (2014).
8. Kunegis, J., Preusse, J. & Schwagerfeit, F. What is the added value of negative links in online social networks? In *Proceeding of the 22nd International Conference on World Wide Web*, 727–736 (2013).
9. Yang, B., Cheung, W. K. & Liu, J. Community mining from signed social networks. *IEEE Transactions on Cybernetics*, 1–14 (2017).

23. Larusso, N., Bogdanov, P. & Singh, A. Identifying communities with coherent and opposing views. In Proceeding of the 15th Annual Graduate Student Workshop in Computing, Santa Barbara: UCSB, 31–32 (2010).
24. Liu, C., Liu, J. & Jiang, Z. A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks. Cybernetics, IEEE Transactions on 44, 2274–2287 (2014).
25. Arenas, A., Fernandez, A. & Gomez, S. Analysis of the structure of complex networks at different resolution levels. New Journal of Physics 10, 053039 (2008).
26. Doreian, P. & Mrvar, A. Partitioning signed social networks. Social Networks 31, 1–11 (2009).
27. Read, K. E. Cultures of the central highlands, new guinea. Journal of Anthropological Research 10, 1–43 (1954).
28. Sampson, F. S. A novitiate in a period of change: an experimental and case study of social relationships. Ph.D. thesis, Cornell University (1968).
29. Yang, B. & Liu, D. Y. A heuristic clustering algorithm for mining communities in signed networks. J. Comput. Sci. and Technol. 22, 320–328 (2007).
30. Esmailian, P. & Jalili, M. Community detection in signed networks: the role of negative ties in different scales. Scientific Reports 5, 14339 (2015).
31. Liu, C., Liu, J. & Jiang, Z. A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks. Cybernetics IEEE Transactions on 44, 2274–2287 (2014).
32. Xu, J. Probe machine. IEEE Transactions on Neural Networks and Learning Systems 27, 1405–1416 (2016).
33. Zhang, X., Pan, L. & Paun, A. On the universality of axon p systems. IEEE Transactions on Neural Networks and Learning Systems 26, 2816–2829 (2017).
34. Zhang, L., Wu, J., Zhuang, Y., Zhang, Y. & Yang, C. Review-oriented metadata enrichment: A case study. In Proceedings of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries, 173–182 (ACM, New York, 2009).
35. Jaccard, P. Distribution de la flore alpine dans le bassin des dranses et dans quelques regions voisines. Bulletin Societe Vaudoise Sciences Naturelle 4 (1901).
36. Leicht, E. A., Holme, P. & Newman, M. E. Vertex similarity in networks. Physical Review E 73, 026120 (2006).
37. Gomez, S., Fernandez, A., Granell, C. & Arenas, A. Structural patterns in complex systems using multidendrograms. Entropy 15, 5464–5474 (2013).
38. Chen, Q., Wu, T. & Fang, M. Detecting local community structures in complex networks based on local degree central nodes. Physica A: Statistical Mechanics and Its Applications 392, 529–537 (2013).
39. Krishnan, V. & Lakshmivarahan, S. Probability and random processes. Journal of the Royal Statistical Society 40, 164–165 (2001).

Acknowledgements

This work was supported by National Natural Science Foundation of China (61502004, 61672033, 91530320, 61272152, 61320106005, 61472002), the Innovation Scientists and Technicians Troop Construction Projects of Henan Province (154200510012), MOE Key Laboratory of Image Processing and Intelligence Control (IPIC2014-03), Natural Science Foundation of Anhui Higher Education Institutions of China (KJ2012A010) and Key Program of Natural Science Project of Educational Commission of Anhui Province (KJ2017A013).

Author Contributions

X.Z., B.W. and Y.S. designed the study; X.Z., B.W. and Y.S. performed the experiments, analyzed the data and prepared the figures; B.W. and Y.S. wrote the paper. X.Z., B.W., Y.S., F.C., L.Z., and L.P. reviewed the manuscript.

Additional Information

Supplementary information accompanies this paper at doi:10.1038/s41598-017-11463-y

Competing Interests: The authors declare that they have no competing interests.

Publisher’s note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.