

Finding communities in directed networks by PageRank random walk induced network embedding

Darong Lai¹, Hongtao Lu¹ and Christine Nardini²

1. Department of computer science and engineering, Shanghai Jiao Tong University, 800 Dong Chuan Road, 200240, Shanghai, China
2. CAS-MPG Partner Institute for Computational Biology, 320 Yue Yang Road, 200031, Shanghai, China

Abstract

Community structure has been found to exist ubiquitously in many different kinds of real world complex networks. Most of the previous literature ignores edges directions and applies methods designed for community finding in undirected networks to find communities. Here, we address the problem of finding communities in directed networks. Our proposed method uses PageRank random walk induced network embedding to transform a directed network into an undirected one, where the information on edges directions is effectively incorporated into the edges weights. Starting from this new undirected weighted network, previously developed methods for undirected network community finding can be used without any modification. Moreover, our method improves on recent work in terms of community definition and meaning. We provide two simulated examples, a real social network and different sets of power law benchmark networks to illustrate how our method can correctly detect communities in directed networks.

PACS: 89.75.Hc

Keywords: directed network; community; random walk; network embedding; modularity.

1. Introduction

Many complex systems, including physical, biological and social systems as well as many man-made technical systems can be modeled by networks. Networks consist of vertices (or nodes) and edges (or links) representing system units and relations between these units, respectively. When the edges have a direction, the network is called directed and, otherwise, undirected. In the past few years, several common properties have been discovered in different kinds of network systems [1]. Among these ubiquitous properties, community structure has recently attracted much attention from various scientific fields.

A network has a community structure if there exist knit groups of vertices characterized by many more edges between vertices in the same group and far less edges among different groups. The community structure phenomenon indicates the heterogeneity of distribution of edges in the networks, and it is important in many aspects. Communities in worldwide web usually correspond to sets of web pages with common topics [2]; communities in biological systems are related to functional

modules, while in social systems they represent sectors of different organizations [3]. Finding community structure in networks facilitates our understanding of the network components, their relationships and dynamical behaviors in the systems they belong to.

A large amount of methods has been proposed to detect communities in networks [4, 5, 6], most of which are especially designed for undirected networks. In the real world, however, many networks of interest are directed, including the World Wide Web, food webs and many other biological networks. Previous and most common methods designed to unveil the communities in directed networks simply discard the direction of edges and treat them as undirected networks. However, due to the informative content of the direction, recent works have turned their attention to the identification of community structures in directed networks [7, 8, 9, 10]. This analysis is frequently performed through the computation of modularity, a useful quality measure of the partitioning of a network into communities, often based on the comparison of the actual network with its configuration model (or null model) [11], which is a network with the same number of nodes and edges per node, but randomly rewired.

Leicht and Newman designed a computationally efficient method using an eigenvector based modularity optimization method for undirected networks [12] and, based on this work, a generalized modularity for directed ones [7]. However, this latter approach shows two main limitations: (i) it cannot effectively distinguish the direction of edges and (ii) it cannot recognize modularity in networks with edges representing patterns of movement among vertices. Kim et al. [13] proposed a new definition of modularity for directed networks to tackle the problem more efficiently. Although their approach successfully deals with directed networks having edges representing patterns of movement among vertices, in substance it transforms a directed network into a new directed one to incorporate the information contained in the edges directions of the original network. This transformation can certainly benefit from newly designed community detection methods for directed networks (such as [7]), however, it evokes two new limitations. First, due to the special choice of the configuration model used in the modularity definition, this approach cannot directly use other useful methods originally designed for undirected networks [4, 5, 6]. Second, it cannot guarantee that the configuration model network is connected and thus cannot guarantee the existence of the stationary distribution of the random walk (i.e. the probability of being on one node only depends on the number of edges leaving the node). This makes the interpretation of a community in term of trapped time of a random walker problematic¹.

In this paper, we propose a new transformation of a directed network into an undirected one, which is essentially different from the previous naïve strategies ignoring edges directions in the directed networks. In fact, it incorporates the

¹ Recently, we have noticed that in the 2nd version of the paper by Kim et al. [13], the authors have changed their original configuration model to the one here proposed. We have also noticed that the work of their 2nd version paper was done 18 days after we submitted our manuscript. The two proposed new definitions of modularity happen to be the same. However, our definition is strictly derived from network embedding and thus gives an additional natural explanation of the new modularity definition.

information of edges directions by weighting the resultant edges in the transformed undirected network, which warrants the possibility to take advantage of previously developed algorithms for undirected networks. Moreover, compared to the modularity defined in Ref.13, our new transformation uses the configuration model defined in Ref.11 for which connectedness is guaranteed, and this makes the interpretation of a community in term of trapped time more intuitive. In fact, our method defines a community as a group of vertices sharing some common relationship, i.e. they are more similar to each other in the group than to vertices outside the group, an interpretation closer to the intuitive original meaning of community structure. It must be pointed out that the modularity has recently been found to have a resolution limit [14]. Nevertheless, we still adopt modularity optimization as a basic algorithm to find communities. However, since the key strategy in our method is to transform a directed network into an undirected one, it is also possible to adopt other types of modularity-independent community detection methods to detect communities.

Eliminato: 15

Eliminato: in practice, modularity is still useful and efficient to detect communities in networks. Thus, in this paper,

In Section 2, we review some basic concepts to introduce our approach. Section 3 describes the proposed method. The relationship with other methods and examples will be given in Section 4. Finally, we discuss the results and draw some conclusive consideration.

2. Basic concepts

A weighted network $G=(V,E)$ has n vertices in a definite vertex set V , as well as an edge set $E \subseteq (V,V)$ containing vertex pairs. In a directed network an edge is an ordered vertex pair (i,j) , while in an undirected one both the vertex pair (i,j) and (j,i) represent the same edge. A network can be represented by an *adjacency* matrix A whose elements are nonnegative, i.e. A_{ij} is positive if there is an edge between vertex i and vertex j , and 0 otherwise. The out-degree k_i^{out} of a vertex i in a directed network is the total weight $\sum_j A_{ij}$ of edges starting from it, while its in-degree k_i^{in} is the total weight $\sum_j A_{ji}$ of edges ending up to it. In undirected networks, $k_i^{out} \equiv k_i^{in}$ and is simply called degree k_i .

Now consider a discrete random walk process on an undirected network [15]: starting from a vertex, a walker randomly and uniformly selects among one of the neighbors of that vertex the one he will move to at the next time step, and repeats the process. The sequence of visited vertices defines a Markov chain, the states of which are the vertices on the network at each step. This process can be specified by a matrix

Eliminato: 16

P called *transition matrix*, whose elements are $P_{ij} = \frac{A_{ij}}{k_i}$ denoting the probability of

transiting from vertex i to vertex j at a given time step. If we define the diagonal matrix D having the degree of each vertex of a connected network on its diagonal, i.e. $D_{ii} = k_i, \forall i$, then $P = D^{-1}A$, where $diag(D)$ is also named *degree sequence*. If

at time step $t \rightarrow \infty$, the probability of the walker to stay on vertex i (π_i) only depends on the vertex degree (k_i), the random walk is said to have stationary

distribution and $\pi_i = \frac{k_i}{2M}$, where M is the sum of weights of the edges in the network. The connectedness of an undirected network guarantees that the random walk on it is irreducible, meaning that the stationary state of the random walk exists.

Similarly, random walk process can also exist on a directed network [16]. The

Eliminato: 17

probability of transiting from vertex i to vertex j can be $P_{ij} = \frac{A_{ij}}{k_i^{out}}$ if $k_i^{out} \neq 0$. If

the Markov chain of a random walk on a directed network is irreducible, the stationary distribution π_i of staying at each vertex i exists and satisfies $\sum_i \pi_i = 1$.

However, for a general directed network, the existence of the stationary distribution is not warranted. To tackle this problem, Page and Brin introduced a special transition matrix that ensures the random walk on a general directed network is irreducible, and successfully used it as an important tool to rank web pages [17].

Eliminato: 18

For the sake of clarity, from now on, in this paper, we will call the random walk associated with this modified transition matrix as *PageRank random walk*. If we denote the out-degree matrix of a directed network as $D_{out} = diag(k_1^{out}, \dots, k_n^{out})$ and a a vector of length n with all zeros but $a_i = 1$ when $k_i^{out} = 0$, along with a n -dimensional vector $e = (1, 1, \dots, 1)^T$ of all ones, the modified transition matrix P is defined as:

$$P = \alpha(D_{out}^+ A + \frac{1}{n} a e^T) + (1 - \alpha) \frac{1}{n} e e^T \quad (2.1)$$

where $1 - \alpha$ is the probability of randomly and uniformly jumping to any vertex of the network and D_{out}^+ is the Moore-Penrose pseudoinverse of D_{out} . Matrix P

defined in this way will be stochastic, irreducible and primitive [16]. The values of

Eliminato: 17

this transition matrix can be interpreted as the probability $\alpha \frac{A_{ij}}{k_i^{out}} + (1 - \alpha) \frac{1}{n}$, when

$k_i^{out} \neq 0$ to jump to a direct neighbor of vertex i , and as the probability $(1-\alpha)\frac{1}{n}$ to jump uniformly randomly to any vertex on the network when i is a dangling node ($k_i^{out} = 0$). The higher the value of α , the more accurately the topology will be preserved. The stationary distribution is then specified by $\pi = P^T \pi$, where $\pi = (\pi_1, \dots, \pi_n)$ and $\sum_i \pi_i = 1$. Notably, π , also called PageRank vector, is used as the metric of importance of web pages in PageRank context.

To qualify a community partition, a useful metric called modularity is widely used [12, 18]. For undirected weighted networks, it is defined by:

Eliminato: 19

$$Q^{ud} = \sum_g e_{gg} - a_g^2 = \frac{1}{2M} \sum_{i,j} (A_{ij} - \frac{k_i k_j}{2M}) \delta(c_i, c_j) \quad (2.2)$$

Here, e_{gg} is the fraction of weight of edges inside the community g , and $a_g = \sum_c e_{gc}$ indicates the fraction of weight of edges incident to g . M is the sum of weights of the edges in the network. The community membership of vertex i is denoted as c_i , and the Kronecker function $\delta(\cdot, \cdot)$ ensures that the summation is being performed over all pairs of vertices in the same community. The modularity measures how much the network deviates from its random counterpart (the configuration model) with the same degree sequence (k_1, \dots, k_n) . In general, the higher the value of the modularity, the better the partition of the network into communities, and the general values of modularity for real networks lie within the range 0.3 to 0.7 [18]. In this perspective, finding community structure in complex networks corresponds to finding the maximal value of the modularity. Since modularity maximization is proven to be a NP-hard problem [19], several heuristic methods have been proposed to approach the goal [4, 5, 6].

Eliminato: 19

Eliminato: 20

Recently, the formula of modularity has been generalized by Arenas et al. [8] to measure the quality of community partitions in directed networks. The generalized modularity for directed network decomposition is defined as:

$$Q^d = \frac{1}{M} \sum_{i,j} (A_{ij} - \frac{k_i^{out} k_j^{in}}{M}) \delta(c_i, c_j) \quad (2.3)$$

This modularity is used as a benefit function to vote for a statistically surprising configuration: if a vertex i has high out-degree but low in-degree while vertex j is in the reverse situation, there is more likely a directed edge from i to j than vice versa. As pointed by Arenas et al. [8], there is a relation between Q^d and Q^{ud} :

$$Q^d = Q^{ud} + \frac{1}{16M^2} \sum_{ij} (k_i^{out} - k_i^{in})(k_j^{out} - k_j^{in}) \quad (2.4)$$

where Q^{ud} is obtained by ignoring the direction of edges in the original directed network. An intuitive interpretation of this mathematical relation is given by the case of an undirected network with no obvious modularity ($Q^{ud}=0$, for example a clique).

In this case, the directed network can still show modularity ($Q^d > 0$), for example 2 communities: one characterized by nodes with larger out-degree, and one characterized by nodes with larger in-degree [7].

3. Proposed method

The topology of a network is specified once its adjacency matrix is given, as the adjacency matrix is a network representation in the vertex space. In our method, we aim to represent the vertices of a directed network in a Euclidean vector space while preserving as much as possible the local properties of each vertex i to all its neighbors. This strategy is depicted in Fig. 1. A network community is a group of vertices sharing one (or more) common properties, i.e. vertices in the same community share a certain type of similarity with each other. The purpose of network embedding is to represent each vertex of a network as a low dimensional vector that preserves these similarities between the vertex pairs, usually measured by the edges weights. Related works have already been done in the last few years, and focus on embedding vertices of networks into a Euclidean space [20, 21, 22, 23]. Among those embeddings for undirected networks, the Laplacian $L^{ud} = D - A$ of an undirected network plays an important role in the success of the algorithms. Motivated by these works, we begin our analysis on directed networks basing our approach on a particular Laplacian for directed networks, originally defined in spectral graph theory by Fan Chung [24] and called combinatorial Laplacian there.

We here use PageRank random walk [17] to define the combinatorial Laplacian for directed networks and call it *directed PageRank combinatorial Laplacian* (L^{PRd}). Given the PageRank random walk transition matrix P of a directed network, together with a diagonal matrix Π consisting of probability of staying on each vertex in stationary state on its diagonal, i.e. $\Pi = \text{diag}(\pi_1, \dots, \pi_n)$, the directed PageRank combinatorial Laplacian is:

$$L^{PRd} = \Pi - \frac{\Pi P + P^T \Pi}{2} \quad (3.1)$$

Knowing $Pe = e$ and $\Pi e = (\pi_1, \dots, \pi_n)$, it is easy to show that

$\Pi = \text{diag}\left(\frac{\Pi P + P^T \Pi}{2} e\right)$. Comparing the directed PageRank combinatorial Laplacian

Eliminato: 21

Eliminato: 22

Eliminato: 23

Eliminato: 24

Eliminato: 25

Eliminato: 18

L^{PRd} defined in equation (3.1) to the one for undirected networks L^{ud} , we can interpret $W = \frac{\Pi P + P^T \Pi}{2}$ as the adjacency matrix of a new network. Since W is a

symmetric matrix ($W = W^T$), Π can also be regarded as the degree matrix of that network. As a result, we obtain an undirected network from the directed PageRank combinatorial Laplacian of a directed network. To the best of our knowledge this interpretation is innovative and, in the following, we will show that the induced undirected network has strong relationship with the original directed network, as it is in fact a transformation of a directed network into an undirected one effectively incorporating the information of the direction of edges. Similarly to the previously

adopted strategies [20, 21, 22, 23], we map the network to a R -dimensional

Euclidean space. Each vertex of the network is then a R -dimensional vector point

$\mathbf{x}_i = (x_{i1}, x_{iR})^T$ in this space. It is then possible to design an objective function to satisfy the condition of preserving local properties among vertices to get optimal embedding of each vertex:

$$\sum_{ij} W_{ij} \|x_i - x_j\|^2 \quad (3.2)$$

where, $\|\cdot\|$ is the vector 2-norm. This objective function is called *smooth function* in

semi-supervising learning literature [25], and produces heavy penalty if neighboring vertex i and vertex j are mapped far apart. Optimal embedding is obtained by minimizing the above-designed objective function.

Proposition 1 Let $X = (x_1, \dots, x_n)^T$, then $\sum_{ij} W_{ij} \|x_i - x_j\|^2 = 2tr(X^T L X)$.

Proof: At first,

$$\begin{aligned} & 2tr(X^T L X) \\ &= tr(X^T (2\Pi - (\Pi P + P^T \Pi)) X) \\ &= 2tr(X^T \Pi X) - tr(X^T \Pi P X) - tr(X^T P^T \Pi X) \\ &= 2\sum_i \pi_i x_i^T x_i - \sum_i \pi_i x_i^T \sum_j p(i, j) x_j - \sum_j p(i, j) x_j^T \sum_i \pi_i x_i \\ &= 2\sum_i \pi_i x_i^T x_i - \sum_i \pi_i \sum_j p(i, j) x_i^T x_j - \sum_j \pi_j \sum_i p(j, i) x_j^T x_i \\ &= \sum_i \pi_i x_i^T x_i + \sum_j \pi_j x_j^T x_j - 2\sum_i \pi_i \sum_j p(i, j) x_i^T x_j \\ &= \sum_i \pi_i x_i^T x_i \sum_j p(i, j) + \sum_j (\sum_i \pi_i p(i, j)) x_j^T x_j - 2\sum_i \pi_i \sum_j p(i, j) x_i^T x_j \quad (3.3) \\ &= \sum_i \pi_i \sum_j p(i, j) \|x_i - x_j\|^2 \end{aligned}$$

Here, we make use of $\sum_j p(i, j) = 1$ for any vertex i and $\pi_j = \sum_i \pi_i p(i, j)$

Eliminato: 21

Eliminato: 22

Eliminato: 23

Eliminato: 24

Eliminato: 26

when random walk reaches its stationary state for the second to the last equality in the above induction. We further simplify the above equation:

$$\begin{aligned}
& \sum_i \pi_i \sum_j p(i, j) \|x_i - x_j\|^2 \\
&= \sum_{ij} \pi_i p(i, j) \|x_i - x_j\|^2 \\
&= \frac{1}{2} (\sum_{ij} \pi_i p(i, j) \|x_i - x_j\|^2 + \sum_{ij} \pi_j p(j, i) \|x_j - x_i\|^2) \quad (3.4) \\
&= \frac{1}{2} (\sum_{ij} (\pi_i p(i, j) + \pi_j p(j, i)) \|x_i - x_j\|^2) \\
&= \sum_{ij} W_{ij} \|x_i - x_j\|^2
\end{aligned}$$

this completes the proof.

Since a semi-positive definite quadratic form has only one symmetric definite square matrix associated with it, we obtain the weighted undirected network representation of a directed network. In contrast to previous literature, which deals with directed network by simply ignoring edges directions, our new method incorporates them into the weights of the newly induced undirected network. In summary, in our approach we propose to start from the symmetric matrix W , treating it as an adjacency matrix of an undirected network, and use any among the many available community detection algorithms to decompose the new undirected network into communities. In this paper, we call the induced undirected network that reweighs pairwise relations between all vertices of the original directed network *similarity network*.

Among many community detection methods designed for undirected weighted networks, we are especially interested in the one proposed by Newman [12] -which reformulates the problem of community detection as an eigenvalue decomposition problem- as this approach is computationally efficient and highly effective in practical applications. This modularity matrix based method, combined with a fine tuning strategy, iteratively bisects the network until there is no sub-network to be further divided to increase the value of modularity. The largest eigenvalue of the modularity matrix plays a key role in this process, as it acts as an indicator of whether a subnetwork can be further divided or not. In the following, we use this method to find communities.

By PageRank induced network embedding, we obtain a new undirected weighted network representation of the original directed network by incorporating the information of edges directions into weights. Starting from this new network with weighted adjacency matrix $W = (W_{ij})$, we rewrite the modularity formula for it:

$$\begin{aligned}
& Q^{d2ud} \\
&= \sum_{ij} (W_{ij} - (\sum_l W_{il})(\sum_k W_{jk})) \delta(c_i, c_j) \\
&= \sum_{ij} \left(\frac{(\Pi P + P^T \Pi)_{ij}}{2} - (\sum_l \frac{(\Pi P + P^T \Pi)_{il}}{2}) (\sum_k \frac{(\Pi P + P^T \Pi)_{jk}}{2}) \right) \delta(c_i, c_j) \quad (3.5) \\
&= \sum_{ij} \left(\frac{(\Pi P + P^T \Pi)_{ij}}{2} - \pi_i \pi_j \right) \delta(c_i, c_j)
\end{aligned}$$

where we make use of the relation $\Pi e = (\pi_1, \dots, \pi_n)$, $P e = e$ and $\sum_{ij} W_{ij} = 1$.

In summary, our algorithm can be summarized as follows:

(1) Input the adjacency matrix A of a directed network and the value of factor α ;

(2) Compute the transition matrix P by equation (2.1);

Eliminato: (2.1)

(3) Solve the eigenvalue problem $\pi = P^T \pi$, subjected to the probability constraint

$$\sum_i \pi_i = 1;$$

(4) Construct the symmetric matrix $W = \frac{\Pi P + P^T \Pi}{2}$, where $\Pi = \text{diag}(\pi_1, \dots, \pi_n)$;

(5) Apply a community detection algorithm [4, 5, 6] to find communities of the weighted undirected network induced by W . These newly found communities are also communities in the original directed networks.

Some notes to the computational complexity of the proposed method should be addressed here. Traditionally, the eigenvalue problem $\pi = P^T \pi$ in step (3) can be

computed by power method [16]. From a starting vector \mathbf{x}_0^T , the k th iteration

Eliminato: 17

equation can be expressed by: $\mathbf{x}_k^T = \alpha \mathbf{x}_{k-1}^T \bar{P} + (\alpha \mathbf{x}_{k-1}^T a + (1-\alpha)) \mathbf{v}^T$, where $\bar{P} = D_{out}^+ A$

and $\mathbf{v} = \frac{1}{n} \mathbf{e}$. Consequently, the eigenvalue problem can be implemented with

vector-matrix multiplications on the extremely sparse \bar{P} . Since generally the average number of non-zeros per row in \bar{P} is small, the time complexity for each step is approximately $O(n)$. It is reported that the power method converges quickly for

about several tens of iterations [17]. Thus, the total complexity of the proposed procedure is mainly dominated by the cost of the community detection algorithm that we choose.

Eliminato: 18

It is interesting to note that if the network is originally an undirected network, our new definition includes the original modularity definition. The stationary distribution π_i^{ud} of a random walker visiting vertex i is $\pi_i^{ud} = k_i / 2M$. Only if the undirected

network is connected, a stationary distribution of random walk on it exists. In this case, we do not need to use the modified transition matrix P for general directed networks, and P becomes $D^{-1}A = \text{diag}^{-1}(\pi_1, L, \dots, \pi_n)A$ and $\Pi = \frac{1}{2M} \text{diag}(k_1, L, \dots, k_n)$.

Combining these equations with equation (3.1), we obtain:

$$W_{ij} = \frac{(\Pi P + P^T \Pi)_{ij}}{2} = \frac{1}{2M} A_{ij} \quad (3.6)$$

By substitution of equation (3.6) into (3.5), we recover the original definition of modularity for undirected networks:

$$\begin{aligned} Q^{d2ud} &= \sum_{ij} (W_{ij} - (\sum_l W_{il})(\sum_k W_{jk})) \delta(c_i, c_j) \\ &= \sum_{ij} (\frac{A_{ij}}{2M} - \frac{k_i}{2M} \frac{k_j}{2M}) \delta(c_i, c_j) = Q^{ud} \end{aligned} \quad (3.7)$$

Thus the new definition unifies the framework for both directed and undirected networks embedding. Benefiting from the abundant optimization algorithms developed to maximize modularity for undirected networks, our method can be effectively and easily used as a tool for communities finding in undirected and directed networks.

As a final note, we briefly mention here the possibility to alternatively analyze the communities of directed networks using techniques from the machine learning community. In fact, PageRank induced network embedding treats vertices as vector points in the Euclidean space. Having embedded the vertices of a directed network into Euclidean space, and represented them by vectors, it is possible to take advantage of the abundant techniques developed in machine learning literature, and in particular clustering approaches, to group these vector points into different clusters corresponding to communities of the original directed network. It is worth noting that in this case extra conditions should be imposed to achieve a correct embedding. First, $x_i^T \Pi x_i = 1$ is needed to remove arbitrary scale of the embedding vector. Second, condition $x_i^T \Pi e = 0$ should be added to get the exact solution, since $Le = 0$ and P is primitive by construction, which means e is the only eigenvector associated to the eigenvalue 0.

Formatted: Motivo: Trasparente

Formatted: Motivo: Trasparente

4. Relation with other methods and Applications

There are several previous works that map networks into Euclidean space and then apply classical machine learning methods to networks analysis [22, 23]. The main idea of this mapping is to preserve as much as possible local properties among vertices on the networks, to benefit from the abundance of well-established techniques produced by the machine learning community. An example is the work that directly transforms vertices of a directed network to vector points and treats these vector points as training instances for machine learning algorithms [22]. We employ a similar

Eliminato: 23

Eliminato: 24

Eliminato: 23

mapping of vertices of the network to vector points. Instead of processing these vector points directly, we obtain a new representation of the original network which can then be processed with all the techniques available in complex network analysis domain. Kim et al. [13] recently proposed a method called Link Rank to detect communities in directed networks. Link Rank uses PageRank induced random walk to rank the edges and then is applied to community detection. The key idea is that two vertices are more likely to be in the same community if the edge connecting them has higher rank. They then defined a new modularity as:

$$Q^{new} = \sum_{ij} (\pi_i P_{ij} - \frac{k_i^{in} k_j^{in}}{M M}) \delta(c_i, c_j) \quad (4.1)$$

with a choice of configuration model different from ours. Denoting $L_{ij} = \pi_i P_{ij}$, Link Rank in essence attempts to represent the original directed network as another directed network. In order to find the maximal Q^{new} , they symmetrized it to obtain:

$$\begin{aligned} Q^{new} &= \frac{1}{2} (\sum_{ij} (\pi_i P_{ij} - \frac{k_i^{in} k_j^{in}}{M M}) \delta(c_i, c_j) + \sum_{ji} (\pi_j P_{ji} - \frac{k_j^{in} k_i^{in}}{M M}) \delta(c_j, c_i)) \\ &= \sum_{ij} (\frac{1}{2} (\pi_i P_{ij} + \pi_j P_{ji}) - \frac{k_i^{in} k_j^{in}}{M M}) \delta(c_i, c_j) \end{aligned} \quad (4.2)$$

This equation is similar to our Q^{d2ud} but different in the second term (different configuration model) within the summation. As explained by Kim et al., their configuration model reflected the expected time a random walker was trapped in communities. They also deemed their configuration model is connected, where edge

weight between any vertex pair i and j was given by $\frac{k_i^{in} k_j^{in}}{M M}$. We point out here

that configuration models constructed in this way cannot guarantee connectedness if there exists a vertex i satisfying $k_i^{in} = 0$. For this reason, the stationary distribution of random walk on this configuration model network may not exist. In contrast, our definition based on network embedding from machine learning field has a natural explanation for the configuration model. Since we transform the original directed network into an undirected weighted one containing information on edges directions, the configuration model used in our modularity definition is the classical one defined in Ref.11, that is a network with the same degree sequence as the network considered but where the edges are rewired randomly. Similarly, in term of random walk our configuration model can be more suitably interpreted as the expected time a random walker is trapped in communities due to the fact that the configuration model is connected.

A flow-based method for community detection in directed networks [10] is also directly related to our work. In this method, Rosvall and Bergstrom employed a

coding scheme borrowed from information theory to detect communities in directed networks. Their method views finding community structure in networks as minimizing the description length of a random walk across the network (also induced by PageRank). The total description length consists of the length for coding community transitions and the length for coding movements with-in communities. When comparing different partitions of the network, the shorter the description for the trajectory of random walk, the more reasonable the community partition is. And the best partition is the one that gives the shortest description length.

Both Link rank and flow-based methods are based on PageRank random walk and try to give the community definition as a group of vertices where a random walker is more likely to be trapped in if the walk starts from one vertex in this group rather than out of the group. Our method adds another interpretation: a community is a group of vertices sharing a common relationship, i.e. they are more similar to each other in the group than to vertices outside the group.

4.1 Simulated examples

Our method is based on PageRank random walk and modularity optimization, but it presents some differences with direct maximization modularity over possible partitions of a network. Here we test two simple directed networks originally used by Rosvall and Bergstrom in their paper [10]. Fig.2 and Fig.3 show these two 16-vertices networks for which different methods give different partitions (the whole discussion can be easily extended to a larger network). Fig.2 shows a network with the weight of bold edges twice the weight of the other edges, which introduces a structure pattern generating persistent movement within and limited movement between four clusters as highlighted in Fig.2 (a). As shown in the figure, our method and flow-based method as well as Link Rank produce identical results (Fig.2 (a)), which all capture the structural regularities of this network. However, the generalized modularity optimization method, which only counts weights of in-edges and out-edges in modules, produces different result (Fig2 (b)) and fails to find this type of regularity. As a byproduct, Fig.4 (a) shows the network embedding result of this approach: the vertices in the same community partitioned by our method are closer than the ones in different communities.

We also applied our method to another directed network where each vertex is either a source or a sink. The network depicted in Fig.3 (a) shows no movement along the edges but still presents some structural similarities between vertices. This network shows no obvious community structure and is likely to be grouped into one cluster. Not surprisingly, all modularity based methods partition the network into four communities as shown in Fig.3 (b) but the values of the respective modularity are different. Both our method and Link Rank obtain very small values (0.1901 and 0.1831, respectively) indicating that there is no obvious community structure, while generalized modularity based method produces a very high value (0.5600) indicating this network has statistically surprising community configuration. In contrast, the flow-based method tends to find that the network has no community structure. Fig.4(b) shows the embedding result which conveys information of the structural equivalence between some vertices, also captured by our method as a result of partitioning into

Formattato: Non
Evidenziato

four groups, though statistically not significant. From Fig. 4 (b), there appear to be four pairs of vertices (sink vertices in this case) superposing together, indicating that every pair of vertices is equivalent. The difference between values obtained by our method and Link Rank is due to the choice of different configuration model in the definition of modularity.

4.2 Friendship social network

We now consider the application of our method to a social network of friendships among school children. This network is a directed network taken from the U.S. National Longitudinal study of Adolescent Health (AddHealth) and it was recently used to test a mixture model based algorithm to find communities [26]. The data were gathered through questionnaires handed out to students who were asked to identify their friends within the school. In contrast to our common view of friendship as a reciprocal relationship, student X identifying Y as a friend in this study does not necessarily mean Y will also identify X as a friend. The friendships in this network are thus represented as directed edges.

We first used PageRank random walk induced network embedding to transform this directed network into an undirected one. Applying eigenvector based modularity optimization to bisect this induced network produces the result shown in Fig.5, together with the division given by the mixture model based method for comparison. The network was plotted in a similar way as done in the application of mixture model based method [26], where the shapes and the colors of the vertices indicate the student ethnicity. As shown in Fig.5, the solid line indicates the result obtained by our method, while the dashed one reflects the one produced by the mixture model based method. Both methods group most of the black students into one group (blue square) and most of the white students into another (red circle), corresponding to the finding that the groupings correlate strongly with student ethnicity as many other networks in the AddHealth data set [27, 28]. However, our method places nearly all white students except two (vertex 85 and 86) into one group, in comparison with mixture model based method placing additional eight white students into another group consisting of most of the black students. In addition, it also distributes more evenly other ethnic students among the two groups (11 and 12 vs. 9 and 14). If this directed network is directly fed into the eigenvector based modularity optimization algorithm, nearly 20% percent of the white students will appear to be grouped with most of the black students. Optimization using equation (4.1) produces results that are not as good as ours in capturing the actual modularity of the network, due to the existence of some dangling vertices in this network. The result of embedding this directed network into a 2-dimensional vector space is also shown in Fig.6 where we can clearly visualize that there are two distinct groups.

4.3 Computer generated power law benchmark networks

In order to test our method more extensively, we use a recently introduced class of benchmark networks with power law distributions of degree and community size [29]. The vertex degrees of a network are controlled by a power law distribution with exponent t_1 , and the community size also distributes according to power law with

Formattato: Non Evidenziato

Eliminato: 27

Eliminato: 27

Eliminato: 28

Eliminato: 29

Formattato: Non Evidenziato

Eliminato: 30

exponent t_2 . The ratio between the external degree of each vertex with respect to its community and the total degree of the vertex is determined by a common *mixing parameter* μ . Thus the larger the value μ of a network is, the harder to detect communities in it. To evaluate the performance of our method on these benchmark networks, we adopt a criterion, *normalized mutual information*, originally used by Danon et al. [4] in the test of community detection algorithms, to measure the similarity of the partition B found by algorithm to the real community partition A . The normalized mutual information is:

$$NMI_{AB} = \frac{-2 \sum_{i=1}^{C_A} \sum_{j=1}^{C_B} N_{ij} \log(N_{ij} N / N_{i\bullet} N_{\bullet j})}{\sum_{i=1}^{C_A} N_{i\bullet} \log(N_{i\bullet} / N) + \sum_{j=1}^{C_B} N_{\bullet j} \log(N_{\bullet j} / N)} \quad (4.3)$$

This definition is based on a *confusion matrix* \mathbf{N} , whose rows are real communities and columns are communities obtained by algorithm. The element N_{ij} represents the number of vertices in real community i that appear in the obtained community j . $N_{i\bullet}$ is the sum over row i and $N_{\bullet j}$ is the sum over column j . The real number of communities is denoted C_A and the number of communities obtained by algorithm is C_B . If the two partitions are identical, NMI_{AB} takes 1 as its maximal value. NMI_{AB} has its minimal value of 0 if the two partitions are independent.

We generate three sets of benchmark networks, each of which is a 1000-vertices network. The average degree of each network is 20 and the maximal degree is 50. For the exponent of the degree distribution and that of the community size the default values provided by the algorithm were used: $t_1 = -2$, $t_2 = -1$. The mixing parameter μ is varying from 0.05 to 0.8. The first set consists of networks with community size ranging from 50 (minCS) to 100 (maxCS), and the second set from 70 to 100. The last set of networks consists of networks with equal sized communities. The performance of our method is shown in Fig. 7. As we can see from the figure, our method correctly found communities in these three set of networks up until mixing parameter $\mu = 0.6$.

According to our natural explanation of new definition of the modularity $Q^{d^{2ud}}$, we can learn from the figure that the vertices on these networks with μ less than or equal to 0.6 are similar if they are in the same community.

5. Discussion and conclusion

In this paper, we have proposed a new method for finding communities in directed networks. Our method uses PageRank induced random walk to perform the embedding of networks into a vector space, preserving as much as possible the local topology of the original directed networks. Differently from previous methods used for detecting communities in directed networks by simply discarding the edges directions, our method effectively incorporates edges directions information into the weights of new edges via network embedding. Network embedding treats vertices of a directed network as vector points, using directed PageRank combinatorial Laplacian. Although community detection in directed networks can be alternatively solved with clustering methods where clusters correspond to communities, we instead start from the directed PageRank combinatorial Laplacian to extract the similarity matrix associated with similarity network. We advantageously interpret the similarity matrix induced by network embedding as a weighted adjacency matrix of a new undirected network. As a result, directed networks community detection can be achieved in the same way as it is performed for undirected networks. Moreover, the new modularity for directed networks includes the one of the original undirected networks as a special case. Our method is similar to Link Rank [13] but with a different choice of configuration model in the modularity definition. Our configuration model ensures the existence of stationary distribution of random walk on it, which makes the trapped time of a random walker on the network having reasonable meaning. In addition, we give another explanation for the definition of new modularity, i.e. communities induced by PageRank random walk are composed of vertices sharing common properties. We successfully apply our method to networks with structure indicating pattern of vertices movement, a real friendship social network of U.S. high school students and different sets of benchmark networks with power law degree distribution and power law community size distribution. Although we use modularity optimization algorithm as the basic algorithm of the proposed procedure, it is not a modularity dependent method.

In summary, based on PageRank random walk induced network embedding, our method effectively transforms a directed network into an undirected one that incorporates edges' direction information into the weights of edges. The advantage of this strategy is that we can directly use previously developed methods for undirected networks to find communities in directed ones. We expect our method to be applied to analyze number of directed networks, including social and biological ones, as well as networks where edges represent patterns of movement among vertices that share some common properties or traits such as webpage networks, citation networks and so on.

Acknowledgement

The authors thank M.E.J. Newman, E. A. Leicht and Petter Holme for their kind help. This work is supported by NSFC under grant No. 60873133.

References

- [1]. S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D.-U. Hwang, Physics Reports 424 (2006), pp175-308

[2]. G. W. Flake, S. R. Lawrence, C. L. Giles, and F. M. Coetzee, IEEE Computer 35 (2002), pp66-71

[3] M. Girvan and M. E. J. Newman, Proc. Natl. Acad. Sci. USA 99 (2002), pp7821–7826

[4] L. Danon, A. Díaz-Guilera, J. Duch, A. Arenas, J. Stat. Mech. (2005), P09008

[5] M. E. J. Newman, Eur. Phys. J. B 38(2004), pp321–330

[6] S. Fortunato, arXiv: 0906.0612

[7] E. A. Leicht and M. E. J. Newman, Phys. Rev. Lett. 100, 118703 (2008)

[8] A. Arenas, J. Duch, A. Fernandez, S. Gomez, New Journal of Physics, 9, 176 (2007)

[9] R. Guimerà, M. Sales-Pardo, LAN Amaral, Phys. Rev. E. 76, 036102(2007)

[10] Martin Rosvall and Carl T. Bergstrom, Proc. Natl. Acad. Sci. USA 105, 1118 (2008)

[11] M. E. J. Newman, Phys. Rev. E. 74, 036104(2006)

[12] M. E. J. Newman, Proc. Natl. Acad. Sci. USA 103, 8577 (2006)

[13] Kim, Youngdo, Son, Seung-Woo, Jeong, Hawoong, eprint arXiv:0902.3728 (2009).

[14] S. Fortunato and M. Barthelemy, Proc. Natl. Acad. Sci. U.S.A. 104, 36 (2007).

[15] Jae Dong Noh, Heiko Rieger, Phys. Rev. Lett. 92,118701(2004)

[16] Amy N. Langville and Carl D. Meyer, Internet Mathematics, 1(3) (2004)

[17] S. Brin, L. Page, [Computer Networks and ISDN Systems](#) 30 (1998), pp 107–117

[18] M.E.J. Newman, M. Girvan, Phys. Rev. E, 69 026113, 2004

[19] U Brandes, D Delling, M Gaertler, R Goerke, Arxiv preprint physics/0608255, 2006

[20] A. Capocci, V D P Servedio, G. Caldarelli, F. Colaiori, Lecture notes in computer science(2004), pp181-187

[21] Donetti L, Munoz M A., Stat. Mech. Theor. Exp. (2004), P10012,

[22] Mo Chen, Qiong Yang, Xiaoou Tang, International Joint Conferences on Artificial Intelligence, 2007

[23] Deng Cai, Xiaofei He, Jiawei Han, Proceedings of the 15th international conference on Multimedia, 2007, pp403-412

[24] F. Chung, Annals of Combinatorics 9 (2005), pp1-19

[25] Xiao Jun Zhu, TR. 1530, University of Wisconsin-Madison: Department of Computer Sciences, 2006

[26] M. E. J. Newman and E. A. Leicht, Proc. Natl. Acad. Sci. U.S.A. 104, 9564 (2007)

[27] James Moody, American Journal of Sociology 107(3) 679:716 (2001)

[28] <http://www.cpc.unc.edu/projects/addhealth>

[29] A. Lancichinetti, S. Fortunato, Phys. Rev. E. 80, 016118 (2009)

Figures and captions

- Eliminato: v1
- Eliminato: [14] Kim, Youngdo, Son, Seung-Woo, Jeong, Hawoong, eprint arXiv:0902.3728v2 (2009).
- Eliminato: 15
- Eliminato: 16
- Eliminato: 17
- Eliminato: 18
- Eliminato: ergey
- Eliminato: awrence
- Eliminato: 7th International World Wide Web Conference,1998
- Formattato: Tipo di carattere: (Predefinito) Times New Roman, 五号
- Eliminato: 19
- Eliminato: 20
- Eliminato: 21
- Eliminato: 22
- Eliminato: 23
- Eliminato: 24
- Eliminato: 25
- Eliminato: 26
- Eliminato: 27
- Eliminato: 28
- Eliminato: 29
- Formattato: Inglese (U. S. A.)
- Eliminato: 30

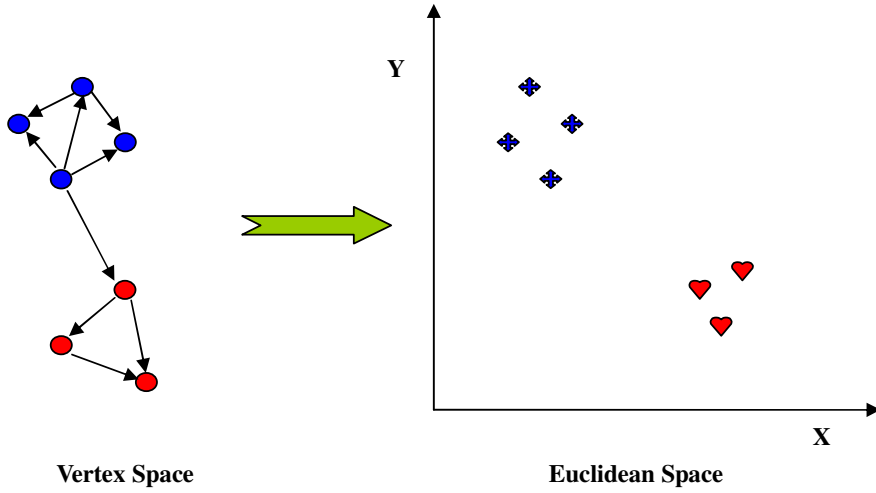


Fig. 1 Schematic illustration of the transformation of a directed network (in vertex space) into the Euclidean vector space. In this embedding, the locality property of a vertex to all its neighbors is preserved as much as possible.

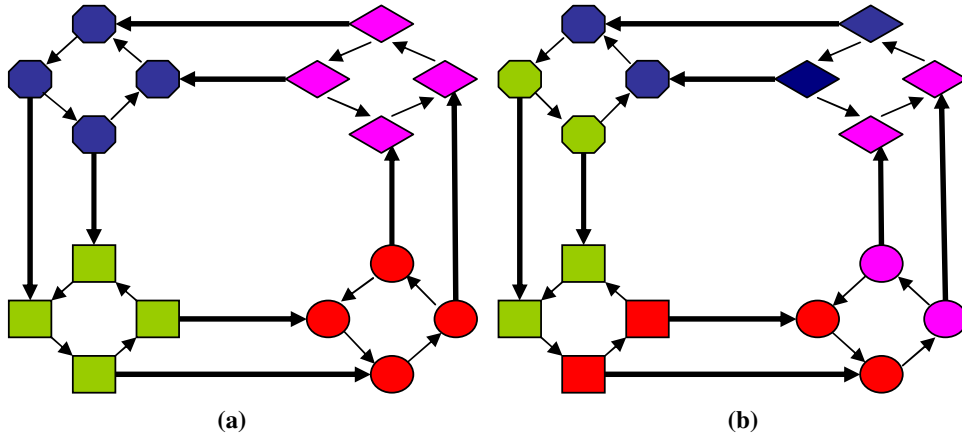


Fig. 2 Community partition by optimizing our modularity Q^{d2ud} in comparison with flow-based method [9], Link Rank method [11] and generalized modularity optimization [6]. The weights of the bold edges are twice those of other normal edges, and the color together with the shape of a node indicates the community membership of that node. (a) Community partition obtained by optimizing Q^{d2ud} , which is identical to that given by the flow-based method (average code length is 2.67bits/step) and that by Link Rank with modularity value $Q^{new} = 0.4133$. The value of our modularity is $Q^{d2ud} = 0.4133$, while the modularity used by Leicht and Newman is $Q^d = 0.2500$. (b) Community partition by optimizing Q^d . For this partition, $Q^d = 0.5000$, $Q^{d2ud} = 0.3304$, $Q^{new} = 0.3304$ and the average length of flow-based method is 4.13bits/step.

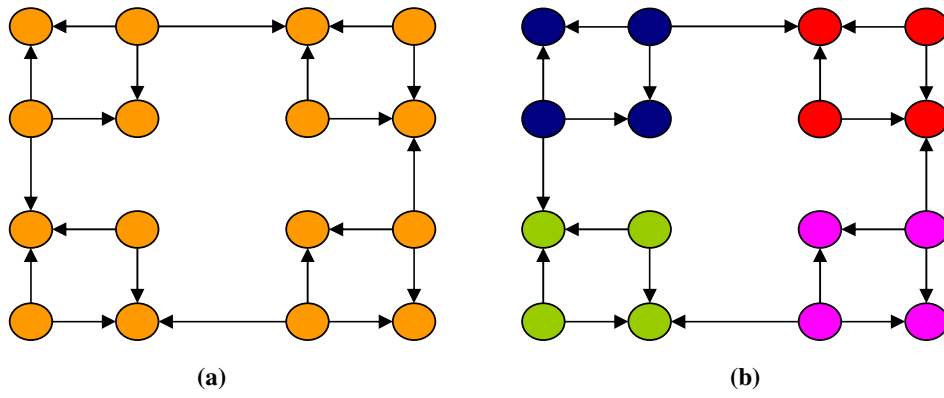
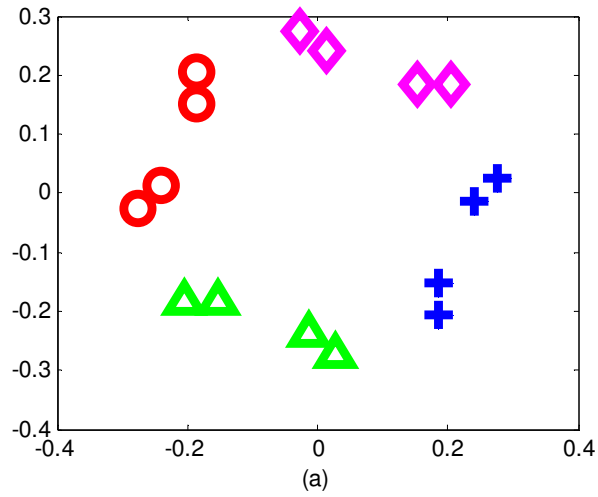


Fig. 3 The network consists of 16 vertices with each node being either a source or a sink. (a) Partition by flow-based method with all vertices in one community. (b) Community partition by optimizing differently defined modularities. $Q^d (=0.5600)$ is significantly higher than $Q^{d2nd} (=0.1901)$ and $Q^{new} (=0.1831)$. The color of a node indicates the community membership of that node.



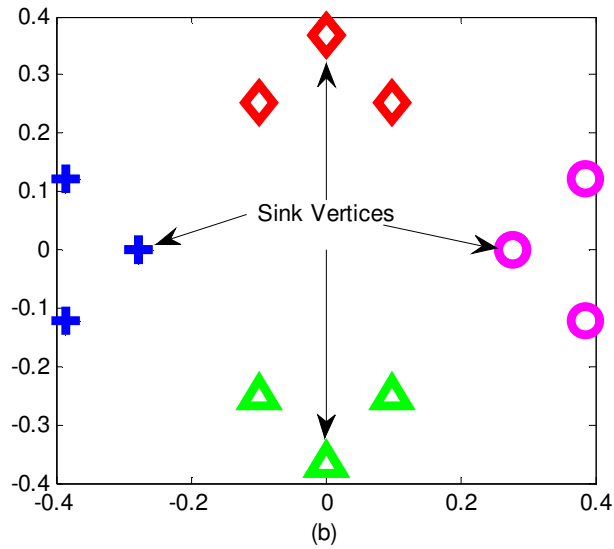


Fig. 4 Results of network embedding. (a) Mapping of the network in Fig.2 into 2-D vector space. (b) Mapping of the network in Fig.3 into 2-D vector space. Four pairs of sink vertices superposition indicate structural equivalence between each pair. The value of α used for mapping is 0.99.

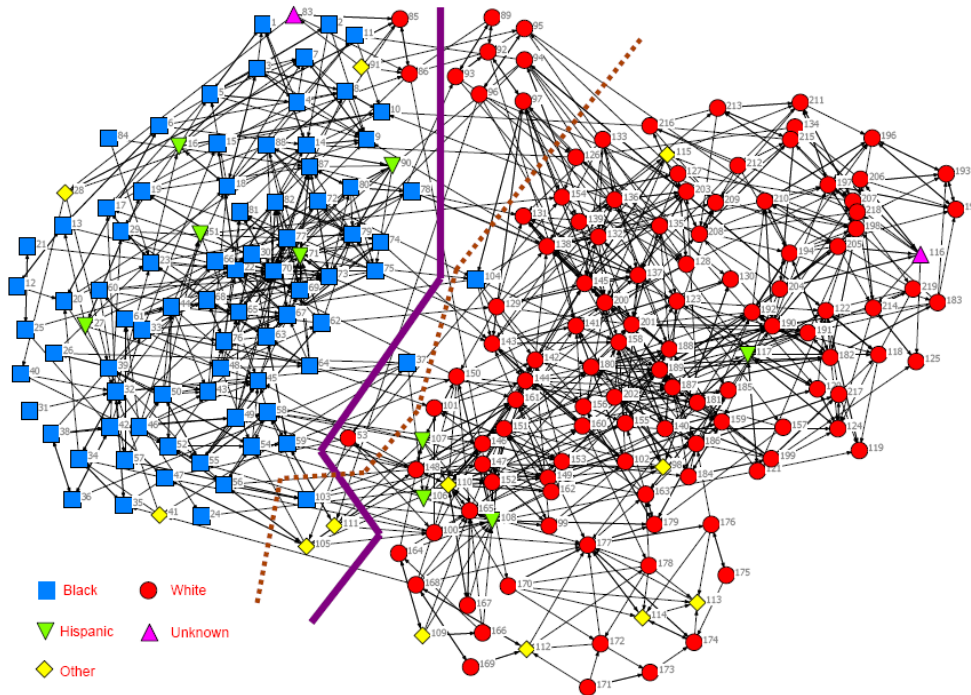


Fig.5 A directed friendship social network of U.S. high school students and the partitions into two groups produced by mixture model based method (dashed line) and our proposed method (solid line). Vertex shapes and colors indicate the ethnicity of the students.

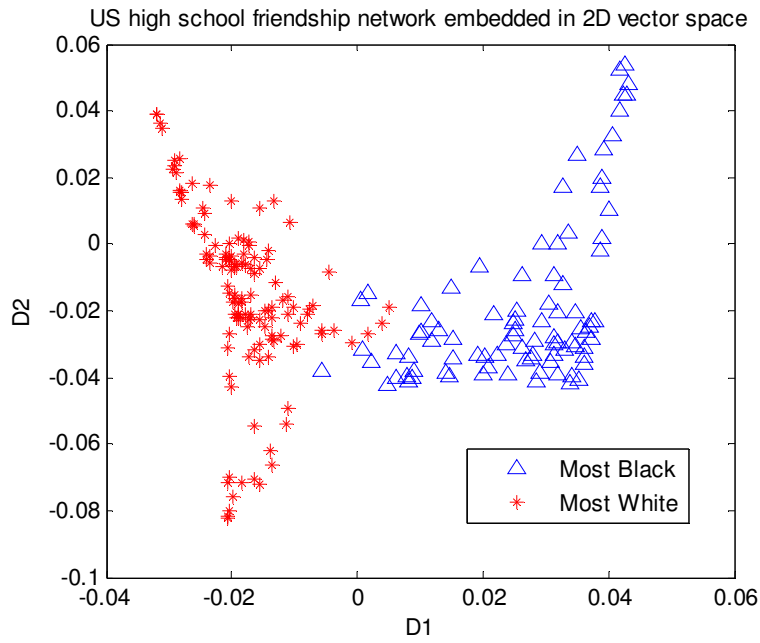


Fig. 6 Result of embedding the directed friendship social network of U.S. high school students into 2-D vector space. The value of α used for mapping is 0.99.

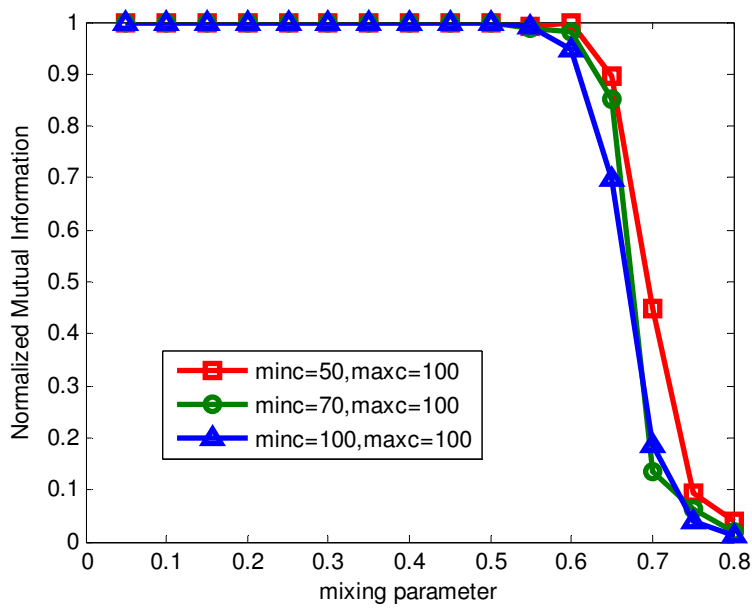


Fig. 7 Results of testing against three sets of benchmark directed and unweighted networks.