Vertex centrality of complex networks based on joint nonnegative matrix factorization and graph embedding

Pengli Lu(卢鹏丽)[†] and Wei Chen(陈玮)[‡]

School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, *China*

(Received 3 March 2022; revised manuscript received 1 April 2022; accepted manuscript online 20 April 2022)

Finding crucial vertices is a key problem for improving the reliability and ensuring the effective operation of networks, solved by approaches based on multiple attribute decision that suffer from ignoring the correlation among each attribute or the heterogeneity between attribute and structure. To overcome these problems, a novel vertex centrality approach, called VCJG, is proposed based on joint nonnegative matrix factorization and graph embedding. The potential attributes with linearly independent and the structure information are captured automatically in light of nonnegative matrix factorization for factorizing the weighted adjacent matrix and the structure matrix, which is generated by graph embedding. And the smoothness strategy is applied to eliminate the heterogeneity between attributes and structure by joint nonnegative matrix factorization. Then VCJG integrates the above steps to formulate an overall objective function, and obtain the ultimately potential attributes fused the structure information of network through optimizing the objective function. Finally, the attributes are combined with neighborhood rules to evaluate vertex's importance. Through comparative analyses with experiments on nine real-world networks, we demonstrate that the proposed approach outperforms nine state-of-the-art algorithms for identification of vital vertices with respect to correlation, monotonicity and accuracy of top-10 vertices ranking.

Keywords: complex networks, centrality, joint nonnegative matrix factorization, graph embedding, smoothness strategy

PACS: 89.75.–k **DOI:** 10.1088/1674-1056/ac6867

1. Introduction

Complex systems from nature and society is modeled and represented by network composed of vertices, that denote entities, and edges, that correspond to the interactions between entity pairs, such as brain networks,[1] message networks,[2,3] human lives,[4] and social systems.[5] Scale-free characteristic of network demonstrates that the impact of different vertices on the function and structure may vary greatly and a few important vertices will affect the cascading effect[6,7] and propagation effect[8] of the network. Where the term "important vertices" means that certain vertices have a greater impact on the network structure and function to a greater extent than other vertices. Therefore, a key problem with important theoretical and practical significance is that of evaluation of vertex importance.

Many experts have made great efforts to evaluate the crucial vertices from different angles and presented many centrality algorithms. From the perspective of topological structure of network, they can generally be divided into two categories in general, which are based on local or global structure to determine vertex's significance. Indicators based on local topology, for example, degree centrality (DC)[9] takes the number of edges connected to neighboring vertices as a metric. Deng *et al*.[10] proposed a method based on the inverse-square law, named I, which defined intensity of vertex by the sum of mutual attraction between each vertex and other vertices. Zareie *et al*.[11] incorporated the notions of Shannon entropy and Jensen–Shannon Divergence to determine vertex centrality. Extend cluster coefficient ranking measure (ECRM)[12] is an improved cluster rank approach combining local clustering coefficient and similarity of connections between neighboring vertices. Node local centrality (NLC)[13] employs network embedding method and network neighborhood topology information to obtain the spreading capability of vertex.

The algorithms based on global structure tend to have relatively higher computational complexity than those based on local topology. For example, betweenness centrality (BC)[14] and closeness centrality (CC)[15] are based on the global path. K-shell (KS)[16] adopts the topological location of vertices and closeness to the graph core to sort them. In view of the shortcomings of KS, hierarchical K-shell (HKS)[17] introduces the *pred*, *succ*, and *sibl* sets to standardize the vertex's topological position and estimates propagation capacity of vertex via its position and neighbors. Deng *et al*.[18] proposed a novel method which takes into account the propagation characteristics of itself and all vertices. Besides, they provided a quantitative model (GIN) to measure the global importance of each vertex. Ullah *et al*.[19] have presented a global structure model (GSM) for influential vertices identification, which not only considers self-propagation characteristics, but also emphasizes

[†]Corresponding author. E-mail: lupengli88@163.com
[‡]Corresponding author. E-mail: chenwei_9711@163.com

© 2023 Chinese Physical Society and IOP Publishing Ltd

vertex's global influence in the network. In addition, the algorithms GLS and MCCD on the basis of the global and local structure of the network were proposed by Sheng *et al.*[20] and Zhang *et al.*[21]

When using vertex centrality indicators to predict which vertices are among the top influencers or super communicators, comparative analyses have demonstrated that no single centrality algorithm is consistent in performance across realistic case studies. At the core of the algorithms to solve this problem is the assessment of important vertices with combining different attributes of vertex. Usually, multiple attributes describing a vertex characteristics are represented by a vector data. That is, for the vertices v_i $(i = 1,2,\ldots,n)$ in network, $\boldsymbol{U}_i = (u_{i1}, u_{i2}, \ldots, u_{i5})$ means that the vertex v_i can be characterized by vector data with five attributes (for instance, the degree of vertex, the Laplacian energy of vertex, the central location of vertex, the propagation characteristics of vertex, the diversity of neighbors). When k attributes are used to characterize the vertex, the vector data $\boldsymbol{U}_i$ is k-dimensional, and the matrix $\boldsymbol{U} \in \mathbb{R}_+^{n\times k}$ is the attribute of the network.

For integrating more attributes of the vertex, many algorithms incorporate multiple attribute decision making into the criterion to evaluate the importance of vertices, however, there are some unsolved problems. For example, indexes with different attributes are subjectively adopted as vertex attributes in the algorithm, which render it difficult to fully characterize the network and ignore the selected attributes whether linearly related. In the next place, the attributes of vertex and structure of the network are heterogeneous. Directly integration attribute and structure cannot effectively deal with the heterogeneity, which hinders the full utilization of them. In addition, for the increasingly large network data, how to more effectively represent attributes and structure information of network are also one of the issues we need to consider.

To overcome these problems, a new vertex centrality approach via joint nonnegative matrix factorization and graph embedding is proposed, referred to as VCJG centrality. VCJG is mainly composed of two major components: joint nonnegative matrix factorization obtains the attributes of vertex which fuse structure information, and the importance of vertices are evaluated applied attrubute and neighborhood rules. For automatically obtaining potential attributes, nonnegative matrix factorization (NMF)[22] is employed to decompose the weighted adjacency matrix. To capture the structure information of network, the global and local topology structure matrix generated by graph embedding are jointly decomposed via NMF. And we apply the smoothness strategy to integrate attributes, local and global structure information for overcoming the heterogeneity of them. Next, VCJG integrates the above steps to formulate an overall objective function, and generates an attribute matrix fused the structure information of network through optimizing the objective function. The above operation can ensure that each attribute obtained is linearly independent. Then we integrate the vertex attribute with neighborhood rules to evaluate vertex's importance finally. To verify the effectiveness of the VCJG criterion, four traditional algorithms and five state-of-the-art centrality approaches are used as measures for comparison. And through extensive experiments on nine real-world networks, the results assert the proposed approach is superior to contrastive algorithms.

The rest paper is structured as follows. Section 2 briefly reviews the related works. In Section 3, a new centrality algorithm is proposed. We analyze the complexity of the proposed algorithm in Section 4. After that, we describe the real-world network datasets and experimental results in Section 5. Finally, Section 6 is the conclusion.

2. Related works

Let $G = \{V(G), E(G)\}$ be an undirected and unweighted network, with vertex set $V(G) = \{v_1, v_2, \ldots, v_n\}$ and edge set $E(G) = \{e_1, e_2, \ldots, e_m\}$, where $|V(G)| = n$ and $|E(G)| = m$. The adjacency matrix of G, denoted by $\boldsymbol{A}(G) = \boldsymbol{A}$, is the $n \times n$ matrix whose (i, j)-entry is 1 if vertices v_i and v_j are adjacent in G and 0 otherwise. Let d_i be the degree of vertex v_i, and $\boldsymbol{D}(G) = \boldsymbol{D}$ the degree diagonal matrix with diagonal entries $d_1, d_2, \ldots, d_n$. The set of all neighbors of v_i is denoted by N_i, where $|N_i| = d_i$. The length of shortest path between vertices v_i and v_j is represented as $dist_{ij}$. The maximum value of $dist_{ij}$ in network G is called the diameter of G, which is represented by $Dist(G) = Dist$.

2.1. Presented approaches

In order to assess the impact of vertex more comprehensively, many scholars have designed centrality algorithms based on multi-attribute decision making. Du *et al.*[23] considered several different approaches as the multiple attributes and employed TOPSIS to aggregate the multiple attributes for measuring the vertex's importance. Yang *et al.*[24] proposed a centrality method that apples VIKOR to integrate multiple centrality criteria from different angles and obtain comprehensive and reasonable ranking results. Yan *et al.*[25] comprehensively considered the multiple attributes of vertex and adopted the entropy weight method to get the corresponding weight of each criterion for overcoming the influence of subjective factors.

The above algorithms have been devoted to measure the importance of vertices from multiple attributes. However, these attributes are generated by multiple vertex centrality algorithms, resulting in correlations between them. For example, in view of the idea of the multi-attribute decision making, Yang *et al.*[26] regard DC, BC, and CC as vertex's attributes, which have correlation because both BC and CC are based on

global structure. In this paper, we will consider using the NMF algorithm to extract vertex attributes that are uncorrelated of each other.

2.2. Network embedding

It is not easy to take advantage of the structural information of the network (graph) because it is discrete by itself. As an example, individuals (vertex) cannot be added to each other, nor can an individual be subtracted from the relationship between individuals (edge). Quite a few researchers have begun to explore a kind of low-dimensional and dense vector representation method to represent high-dimensional network, namely network embedding.

Node2vec[27] learns continuous attribute representations for vertices in networks, which is an algorithmic framework. It has a flexible concept of a vertex's network neighborhood and a biased random walk procedure, which can effectively explore diverse neighborhoods. Therefore, the mapping of vertices to a low-dimensional attribute space can be obtained to preserve the network neighborhood of vertices as much as possible.

2.3. Preliminaries

Lemma 1[28] A walk of length l in G, from vertex v_i to v_j, is a finite sequence of vertices of G,

$$v_i = u_0, u_1, \ldots, u_x, \ldots, u_l = v_j,$$

such that u_{x-1} and u_x are adjacent for $1 \le x \le l$, where u_x represents the $(x+1)$-th vertex in the walk. The number of walks of length l, from v_i to v_j, is the entry in position (i, j) of the matrix $\boldsymbol{A}^l$.

2.3.1. NMF algorithm

The ubiquity of high-dimensional data not only leads to an exponential amount of computation which makes it difficult to ensure real-time processing, but also makes it difficult to mine the intrinsic structure and relevance of the data. Dimensionality Reduction is a data processing method that mines the essential law of data and discovers the internal relationship. Dimensionality reduction is divided into attribute selection and attribute extraction. Attribute selection is a subset of attributes that is extracted directly from the original data, and attribute extraction is a series of mathematical transformations that synthesize new attributes in place of the original data set. In this paper, non-negative matrix factorization (NMF) is used to extract attributes from the original network.

NMF is a method in matrix factorization algorithms that represents a matrix as the multiplication of simple and low-rank matrices. Since network can be expresses as adjacency (laplacian, signless laplacian) matrix, then a clear advantage for matrix decomposition in graph theory lies in the allowance of understanding the fundamental properties of graph and doing straight forward calculation.

Given a nonnegative matrix $\boldsymbol{T} \in \mathbb{R}_+^{m\times n}$, nonnegative matrix factorization (NMF) aims to find two nonnegative matrices $\boldsymbol{B} \in \mathbb{R}_+^{m\times k}$ and $\boldsymbol{F} \in \mathbb{R}_+^{k\times n}$ such that the product of them can approximate the matrix $\boldsymbol{T}$, *i.e.*,

$$\min_{\boldsymbol{B}\ge 0,\, \boldsymbol{F}\ge 0} \sum_{i=1}^{m}\sum_{j=1}^{n} [\boldsymbol{T}_{ij} - (\boldsymbol{BF})_{ij}]^2.$$

Where k is the rank of the matrix, usually $(m+n)k < mn$.[29]

Usually, we can understand that the column vector of the original matrix $\boldsymbol{T}$ is the weighted sum of all column vectors of the matrix $\boldsymbol{B}$, and the weight coefficient is the element of the corresponding column vector of the matrix $\boldsymbol{F}$, as shown in the following equation.

For example Support non-negative matrixs $\boldsymbol{T} \in \mathbb{R}_+^{3\times 3}$, $\boldsymbol{B} \in \mathbb{R}_+^{3\times 2}$, and $\boldsymbol{F} \in \mathbb{R}_+^{2\times 3}$ satisfy $\boldsymbol{F} \approx \boldsymbol{BF}$, *i.e.*,

$$\begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{33} & t_{33} \end{bmatrix} \approx \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix} \times \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \end{bmatrix},$$

where

$$\begin{bmatrix} t_{11} \\ t_{21} \\ t_{31} \end{bmatrix} \approx \begin{bmatrix} b_{11}f_{11} + b_{12}f_{21} \\ b_{21}f_{11} + b_{22}f_{21} \\ b_{31}f_{11} + b_{32}f_{21} \end{bmatrix}.$$

Therefore, it is called matrix $\boldsymbol{B}$ as the base matrix and matrix $\boldsymbol{F}$ as the attribute matrix.

3. Measuring the vertex centrality

VCJG is mainly composed of two major components: joint nonnegative matrix factorization obtains the attribute with linearly independent and fusing structure information, and the vertex's importance are evaluated with attribute and neighborhood rules. In this section, they are described in detail.

3.1. Constructing objective function

The joint matrix factorization in VCJG involves three procedures, where nonnegative matrix factorization is adopted to generate the attribute of networks, the topological information is obtained by joint decomposing the topological matrix, the smoothness procedure connects the attribute and topology information of network. Therefore, the objective function of VCJG is composed of the costs of three programs.

The adjacent matrix $\boldsymbol{A}$ of the network characterizes whether there is interaction between the vertices, but it is not enough to fully describe the strength of it. Then, the weighted adjacent matrix $\boldsymbol{W}$ (Ref. [30]) of the network is defined as follows:

$$w_{ij} = \begin{cases} \dfrac{1}{2}\left(\dfrac{d_i}{d_j} + \dfrac{d_j}{d_i}\right), & \text{if } v_i \text{ and } v_j \text{ are adjacent,} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Obviously it is a symmetric nonnegative matrix. The most widely used strategy for obtaining network attributes is matrix factorization, we intuitively choose to use NMF, *i.e.*,

$$\min_{\boldsymbol{B}\geq 0,\ \boldsymbol{F}\geq 0} \sum_{i=1}^{n}\sum_{j=1}^{n}[\boldsymbol{W}_{ij}-(\boldsymbol{BF})_{ij}]^2, \tag{2}$$

where the $\boldsymbol{B}$ is the basis matrix and $\boldsymbol{F}$ is the attribute matrix corresponding to matrix $\boldsymbol{B}$, and k is the number of attributes. As can be seen from Section 2, Subsection 2.2.1, the k of factor matrix decomposed by NMF satisfies $(m+n)k < mn$. In this algorithm, the number of rows and columns of the original matrix $\boldsymbol{W}$ is n, then we have $(n+n)k < n^2 \Rightarrow k < n/2$.

The preceding equation can be solved by minimizing the squared error, that is, the l_2 norm between two matrices,

$$\min_{\boldsymbol{B}\geq 0,\ \boldsymbol{F}\geq 0} \|\boldsymbol{W}-\boldsymbol{BF}\|^2. \tag{3}$$

As is well known that social networks contain two different dimensions, where the structural dimension describes the interaction and composition dimension indicates vertex attributes. Therefore, only through Eq. (3) to obtain the attributes of the network is not enough to characterize the network. Considering that the topological structure is generally divided into global and local structures, we can obtain the topology information of the network through joint nonnegative matrix factorization. The adjacency matrix can only characterize the 1-order structure. Then for global structure, according to Lemma 1 we define the global topology matrix $\boldsymbol{Tg}$ via the matrix $\boldsymbol{A}^l$ as follows:

$$\boldsymbol{Tg} = \frac{\sum_{l=1}^{Dist}\theta^{l-1}\boldsymbol{A}^l}{\sum_{l=1}^{Dist}\theta^{l-1}}, \tag{4}$$

where *Dist* is the diameter of G and parameter $\theta \in [0.1, 1]$ controls the relative weight of walks. $(\boldsymbol{Tg})_{ij}$ is the weighted average of the number of walk from vertex v_i to v_j. For local structure, since node2vec maps vertices to low-dimensional attribute spaces and maintains the network neighborhood of vertices as much as possible, we apply it to get the local structure matrix $\boldsymbol{Tl}$ of G.

Here, we also use the same strategy by joint decomposing $\boldsymbol{Tg}$ and $\boldsymbol{Tl}$ under the same basis matrix $\boldsymbol{B}$ to obtain the structure information of the network. The function is written as

$$\min_{\boldsymbol{B}\geq 0,\ \boldsymbol{Sg}\geq 0,\ \boldsymbol{Sl}\geq 0} \|\boldsymbol{Tg}-\boldsymbol{BSg}\|^2+\|\boldsymbol{Tl}-\boldsymbol{BSl}\|^2, \tag{5}$$

where the matrices $\boldsymbol{Sg}$ and $\boldsymbol{Sl}$ are local and global topology information, respectively. The attributes and structure of the network are heterogeneous, so the strategy we adopted is to smooth the attribute matrix $\boldsymbol{F}$ using the topology information $\boldsymbol{Sg}$ and $\boldsymbol{Sl}$, *i.e.*,

$$\min_{\boldsymbol{F}\geq 0} \quad \alpha\|\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl}-\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}\|^2+(1-\alpha)\|\boldsymbol{Sg}^{\mathrm{T}}\boldsymbol{Sg}-\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}\|^2, \tag{6}$$

where $\alpha \in [0,1]$ is a parameter.

Considering that the decomposition process mentioned above is independent, we apply joint strategy to construct the objective function. By combining Eqs. (3), (5), (6), the objective function of VCJG is formulated as

$$\begin{aligned}\min f \ =\ & \min \|\boldsymbol{W}-\boldsymbol{BF}\|^2+\|\boldsymbol{Tg}-\boldsymbol{BSg}\|^2+\|\boldsymbol{Tl}-\boldsymbol{BSl}\|^2\\ & +\alpha\|\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl}-\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}\|+(1-\alpha)\|\boldsymbol{Sg}^{\mathrm{T}}\boldsymbol{Sg}-\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}\|,\\ & s.t. \quad \boldsymbol{B}\geq 0,\boldsymbol{F}\geq 0,\boldsymbol{Sl}\geq 0,\boldsymbol{Sg}\geq 0,\end{aligned} \tag{7}$$

where parameter α controls the direction in which the attribute is smoothed to the topology structure.

3.2. Optimal solution

From Eq. (7), we can obtain the multiplicative update solutions of the matrices $\boldsymbol{B}\in\mathbb{R}_+^{n\times k}$, $\boldsymbol{Sl}\in\mathbb{R}_+^{k\times n}$, $\boldsymbol{Sg}\in\mathbb{R}_+^{k\times n}$, and $\boldsymbol{F}\in\mathbb{R}_+^{k\times n}$ by

$$\begin{aligned}\boldsymbol{B} &= \boldsymbol{B}\odot\frac{\boldsymbol{WF}^{\mathrm{T}}+\boldsymbol{TlSl}^{\mathrm{T}}+\boldsymbol{TgSg}^{\mathrm{T}}}{\boldsymbol{B}(\boldsymbol{FF}^{\mathrm{T}}+\boldsymbol{SlSl}^{\mathrm{T}}+\boldsymbol{SgSg}^{\mathrm{T}})},\\ \boldsymbol{Sl} &= \boldsymbol{Sl}\odot\frac{\boldsymbol{B}^{\mathrm{T}}\boldsymbol{Tl}+2\alpha\boldsymbol{F}^{\mathrm{T}}\boldsymbol{FSl}}{(2\alpha\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl}+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})\boldsymbol{Sl}},\\ \boldsymbol{Sg} &= \boldsymbol{Sg}\odot\frac{\boldsymbol{B}^{\mathrm{T}}\boldsymbol{Tg}+2(1-\alpha)\boldsymbol{F}^{\mathrm{T}}\boldsymbol{FSg}}{[2(1-\alpha)\boldsymbol{Sg}^{\mathrm{T}}\boldsymbol{Sg}+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B}]\boldsymbol{Sg}},\\ \boldsymbol{F} &= \boldsymbol{F}\odot\frac{\boldsymbol{B}^{\mathrm{T}}\boldsymbol{W}+2\alpha\boldsymbol{FSl}^{\mathrm{T}}\boldsymbol{Sl}+2(1-\alpha)\boldsymbol{FSg}^{\mathrm{T}}\mathbf{Sg}}{(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B}+2\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F})\boldsymbol{F}},\end{aligned} \tag{8}$$

where symbol $\odot$ is Hadamard product.[31] It is worth noting that the matrices $\boldsymbol{B}$, $\boldsymbol{Sl}$, $\boldsymbol{Sg}$, and $\boldsymbol{F}$ need to be updated at the same time. According to the nature of NMF, the matrix $\boldsymbol{F}$ is a row non-singular matrix. Next, we give the proof of Eq. (8).

Proof Before that, we first give the relationship between the Euclidan norm and the trace of the matrix $\boldsymbol{A}$, *i.e.*, $\|\boldsymbol{A}\|^2 = \mathrm{Tr}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})$.

For example Given a matrix $\boldsymbol{A}=\begin{bmatrix}a_{11} & a_{12}\\ a_{21} & a_{22}\end{bmatrix}\in\mathbb{R}^{2\times 2}$, then its Euclidan norm is

$$\|\boldsymbol{A}\| = (a_{11}^2+a_{12}^2+a_{21}^2+a_{22}^2)^{1/2}.$$

For $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}=\begin{bmatrix}a_{11}^2+a_{21}^2 & a_{11}a_{12}+a_{21}a_{22}\\ a_{12}a_{11}+a_{22}a_{21} & a_{12}^2+a_{22}^2\end{bmatrix}$, we have

$$\mathrm{Tr}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}) = a_{11}^2+a_{21}^2+a_{12}^2+a_{22}^2 = \|\boldsymbol{A}\|^2.$$

Return the above function f, removing irrelevant items, equation (7) is reformulated as

$$\begin{aligned}\min f \ =\ & \min \|\boldsymbol{BF}\|^2+\|\boldsymbol{BSl}\|^2+\|\boldsymbol{BSg}\|^2\\ & +\|\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}\|^2+\alpha\|\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl}\|^2\\ & +(1-\alpha)\|\boldsymbol{Sg}^{\mathrm{T}}\boldsymbol{Sg}\|^2-2[\mathrm{Tr}(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{BF})\\ & +\mathrm{Tr}(\boldsymbol{Tl}^{\mathrm{T}}\boldsymbol{BSl})+\mathrm{Tr}(\boldsymbol{Tg}^{\mathrm{T}}\boldsymbol{BSg})\end{aligned}$$

$$+\alpha\,\mathrm{Tr}(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{SlF}^{\mathrm{T}}\boldsymbol{F})+(1-\alpha)\,\mathrm{Tr}(\boldsymbol{Sg}^{\mathrm{T}}\boldsymbol{SgF}^{\mathrm{T}}\boldsymbol{F})],$$
$$s.t.\quad \boldsymbol{B}\geq 0,\boldsymbol{F}\geq 0,\boldsymbol{Sl}\geq 0,\boldsymbol{Sg}\geq 0. \tag{9}$$

The alternative iteration based strategy is adopted to solve the above formula, where one variable is updated by fixing the others. We first solve the update rule of the matrix $\boldsymbol{B}$. The matrices $\boldsymbol{F}$, $\boldsymbol{Sl}$, $\boldsymbol{Sg}$ are fixed, and irrelevant items are removed. The equation (9) can be rewritten as

$$\min_{\boldsymbol{B}\geq 0}\|\boldsymbol{BF}\|^2+\|\boldsymbol{BSl}\|^2+\|\boldsymbol{BSg}\|^2-2[\mathrm{Tr}(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{BF})$$
$$+\mathrm{Tr}(\boldsymbol{Tl}^{\mathrm{T}}\boldsymbol{BSl})+\mathrm{Tr}(\boldsymbol{Tg}^{\mathrm{T}}\boldsymbol{BSg})]. \tag{10}$$

We use lagrangian multiplier method to deal with the nonnegative constraint of matrix $\boldsymbol{B}$. Let be $\Phi=[\phi_{ij}]$, the lagrange $\mathcal{L}$ of Eq. (10) is

$$\mathcal{L}=\|\boldsymbol{BF}\|^2+\|\boldsymbol{BSl}\|^2+\|\boldsymbol{BSg}\|^2-2[\mathrm{Tr}(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{BF})+\mathrm{Tr}(\boldsymbol{Tl}^{\mathrm{T}}\boldsymbol{BSl})$$
$$+\mathrm{Tr}(\boldsymbol{Tg}^{\mathrm{T}}\boldsymbol{BSg})]+\mathrm{Tr}(\boldsymbol{\Phi B}).$$

The partial derivative of matrix $\boldsymbol{B}$ is derived as follows:

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{B}}=\frac{\partial(\|\boldsymbol{BF}\|^2)}{\partial\boldsymbol{B}}+\frac{\partial(\|\boldsymbol{BSl}\|^2)}{\partial\boldsymbol{B}}+\frac{\partial(\|\boldsymbol{BSg}\|^2)}{\partial\boldsymbol{B}}$$
$$-\frac{2\partial[\mathrm{Tr}(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{BF})+\mathrm{Tr}(\boldsymbol{Tl}^{\mathrm{T}}\boldsymbol{BSl})+\mathrm{Tr}(\boldsymbol{Tg}^{\mathrm{T}}\boldsymbol{BSg})]}{\partial\boldsymbol{B}}$$
$$+\frac{\partial[\mathrm{Tr}(\boldsymbol{\Phi B})]}{\partial\boldsymbol{B}}.$$

Respectively, according to the formulas

$$\frac{\partial[\mathrm{Tr}(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})]}{\partial\boldsymbol{X}^{\mathrm{T}}}=2\boldsymbol{X},\quad\frac{\partial(\boldsymbol{XA})^{\mathrm{T}}}{\partial\boldsymbol{X}}=\boldsymbol{A}^{\mathrm{T}},$$

we have

$$\frac{\partial(\|\boldsymbol{BF}\|^2)}{\partial\boldsymbol{B}}=\frac{\partial[\mathrm{Tr}(\boldsymbol{F}^{\mathrm{T}}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{BF})]}{\partial\boldsymbol{B}}$$
$$=\frac{\partial[\mathrm{Tr}((\boldsymbol{BF})^{\mathrm{T}}\boldsymbol{BF})]}{\partial(\boldsymbol{BF})^{\mathrm{T}}}\times\frac{\partial(\boldsymbol{BF})^{\mathrm{T}}}{\partial\boldsymbol{B}}$$
$$=2\boldsymbol{BF}\times\boldsymbol{F}^{\mathrm{T}}=2\boldsymbol{BFF}^{\mathrm{T}}.$$

Similarly,

$$\frac{\partial(\|\boldsymbol{BSl}\|^2)}{\partial\boldsymbol{B}}=2\boldsymbol{BSlSl}^{\mathrm{T}},$$
$$\frac{\partial(\|\boldsymbol{BSg}\|^2)}{\partial\boldsymbol{B}}=2\boldsymbol{BSgSg}^{\mathrm{T}}.$$

Next, according to the formula

$$\frac{\partial\,\mathrm{Tr}(\boldsymbol{AXB})}{\partial\boldsymbol{X}}=\boldsymbol{A}^{\mathrm{T}}\boldsymbol{B}^{\mathrm{T}},\quad\frac{\partial[\mathrm{Tr}(\boldsymbol{AX})]}{\partial\boldsymbol{X}}=\boldsymbol{A}^{\mathrm{T}},$$

we have

$$\frac{\partial[\mathrm{Tr}(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{BF})]}{\partial\boldsymbol{B}}=\boldsymbol{WF}^{\mathrm{T}},$$
$$\frac{\partial[\mathrm{Tr}(\boldsymbol{Tl}^{\mathrm{T}}\boldsymbol{BSl})]}{\partial\boldsymbol{B}}=\boldsymbol{TlSl}^{\mathrm{T}},$$
$$\frac{\partial[\mathrm{Tr}(\boldsymbol{Tg}^{\mathrm{T}}\boldsymbol{BSg})]}{\partial\boldsymbol{B}}=\boldsymbol{TgSg}^{\mathrm{T}},$$
$$\frac{\partial[\mathrm{Tr}(\boldsymbol{\Phi B})]}{\partial\boldsymbol{B}}=\boldsymbol{\Phi}^{\mathrm{T}}.$$

Finally,

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{B}}=2\boldsymbol{BFF}^{\mathrm{T}}+2\boldsymbol{BSlSl}^{\mathrm{T}}+2\boldsymbol{BSgSg}^{\mathrm{T}}$$
$$-2\boldsymbol{WF}^{\mathrm{T}}-2\boldsymbol{TlSl}^{\mathrm{T}}-2\boldsymbol{TgSg}^{\mathrm{T}}+\boldsymbol{\Phi}^{\mathrm{T}}.$$

By setting $\partial\mathcal{L}/\partial\boldsymbol{B}=0$, we obtain the updating rule for matrix $\boldsymbol{B}$ as follows in light of the Karush–Kuhn–Tucker (KKT) condition $\phi_{ij}(\boldsymbol{B})_{ij}=0$, *i.e.*,

$$(2\boldsymbol{BFF}^{\mathrm{T}}+2\boldsymbol{BSlSl}^{\mathrm{T}}+2\boldsymbol{BSgSg}^{\mathrm{T}}-2\boldsymbol{WF}^{\mathrm{T}}$$
$$-2\boldsymbol{TlSl}^{\mathrm{T}}-2\boldsymbol{TgSg}^{\mathrm{T}}+\boldsymbol{\Phi}^{\mathrm{T}})_{ij}(\boldsymbol{B})_{ij}=0,$$
$$(\boldsymbol{BFF}^{\mathrm{T}}+\boldsymbol{BSlSl}^{\mathrm{T}}+\boldsymbol{BSgSg}^{\mathrm{T}})_{ij}(\boldsymbol{B})_{ij}$$
$$=(\boldsymbol{WF}^{\mathrm{T}}+\boldsymbol{TlSl}^{\mathrm{T}}+\boldsymbol{TgSg}^{\mathrm{T}})_{ij}(\boldsymbol{B})_{ij},$$
$$\boldsymbol{B}_{ij}=\frac{\boldsymbol{B}_{ij}[\boldsymbol{WF}^{\mathrm{T}}+\boldsymbol{TlSl}^{\mathrm{T}}+\boldsymbol{TgSg}^{\mathrm{T}}]_{ij}}{[\boldsymbol{B}(\boldsymbol{FF}^{\mathrm{T}}+\boldsymbol{SlSl}^{\mathrm{T}}+\boldsymbol{SgSg}^{\mathrm{T}})]_{ij}}. \tag{11}$$

By fixing the matrices $\boldsymbol{B}$, $\boldsymbol{F}$, and $\boldsymbol{Sg}$, the update rule of the matrix $\boldsymbol{Sl}$ can be calculated. Let $(\boldsymbol{\lambda})_{ij}$ be the lagrange multiplier for the constraints $(\boldsymbol{Sl})_{ij}$, the lagrange $\mathcal{L}$ is

$$\mathcal{L}=\alpha\,\mathrm{Tr}(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{SlSl}^{\mathrm{T}}\boldsymbol{Sl})+\|\boldsymbol{BSl}\|^2-2\,\mathrm{Tr}(\boldsymbol{Tl}^{\mathrm{T}}\boldsymbol{BSl})$$
$$-2\alpha\,\mathrm{Tr}(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{SlF}^{\mathrm{T}}\boldsymbol{F})+\mathrm{Tr}(\boldsymbol{\lambda Sl}).$$

According to the formulas

$$\frac{\partial[\mathrm{Tr}(\boldsymbol{X}^2)]}{\partial\boldsymbol{X}}=2\boldsymbol{X}^{\mathrm{T}},\quad\frac{\partial[\mathrm{Tr}(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{AX})]}{\partial\boldsymbol{X}}=\boldsymbol{AX}+\boldsymbol{A}^{\mathrm{T}}\boldsymbol{X},$$

the partial derivative of $\boldsymbol{Sl}$ is formulated as follow:

$$\frac{\partial[\mathrm{Tr}(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{SlSl}^{\mathrm{T}}\boldsymbol{Sl})]}{\partial\boldsymbol{Sl}}=\frac{\partial[\mathrm{Tr}(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl})^2]}{\partial(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl})}\times\frac{\partial(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl})}{\partial\boldsymbol{Sl}}=2\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl}\times 2\boldsymbol{Sl}=4\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{SlSl},$$
$$\frac{\partial(\|\boldsymbol{BSl}\|^2)}{\partial\boldsymbol{Sl}}=\frac{\partial[\mathrm{Tr}(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{BSl})]}{\partial\boldsymbol{Sl}}=\frac{\partial[\mathrm{Tr}(\boldsymbol{Sl}^{\mathrm{T}}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})\boldsymbol{Sl})]}{\partial\boldsymbol{Sl}}=(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})\boldsymbol{Sl}+(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})^{\mathrm{T}}\boldsymbol{Sl}=2\boldsymbol{B}^{\mathrm{T}}\boldsymbol{BSl},$$
$$\frac{\partial[\mathrm{Tr}(\boldsymbol{Tl}^{\mathrm{T}}\boldsymbol{BSl})]}{\partial\boldsymbol{Sl}}=\boldsymbol{B}^{\mathrm{T}}\boldsymbol{Tl},$$
$$\frac{\partial[\mathrm{Tr}(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{SlF}^{\mathrm{T}}\boldsymbol{F})]}{\partial\boldsymbol{SL}}=\frac{\partial[\mathrm{Tr}(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{SlF}^{\mathrm{T}}\boldsymbol{F})]}{\partial(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl})}\times\frac{\partial(\boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl})}{\partial\boldsymbol{Sl}}=2\boldsymbol{F}^{\mathrm{T}}\boldsymbol{FSl},\quad\frac{\partial[\mathrm{Tr}(\boldsymbol{\lambda Sl})]}{\partial\boldsymbol{Sl}}=\boldsymbol{\lambda}^{\mathrm{T}},$$

finally

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{Sl}} = 4\alpha \boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{SlSl} + 2\boldsymbol{B}^{\mathrm{T}}\boldsymbol{BSl} - 2\boldsymbol{B}^{\mathrm{T}}\boldsymbol{Tl} - 4\alpha \boldsymbol{F}^{\mathrm{T}}\boldsymbol{FSl} + \boldsymbol{\lambda}^{\mathrm{T}}.$$

By setting the derivative of $\mathcal{L}(\boldsymbol{Sl})$ with respect to $\boldsymbol{Sl}$ as 0 and the KKT condition, we obtain the rule to update matrix $\boldsymbol{Sl}$ as

$$\boldsymbol{Sl}_{ij} = \frac{\boldsymbol{Sl}_{ij}[\boldsymbol{B}^{\mathrm{T}}\boldsymbol{Tl} + 2\alpha \boldsymbol{F}^{\mathrm{T}}\boldsymbol{FSl}]_{ij}}{[(2\alpha \boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{Sl} + \boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})\boldsymbol{Sl}]_{ij}}. \tag{12}$$

Analogously, the updating rule for matrix $\boldsymbol{Sg}$ is as follows via fixing $\boldsymbol{B}$, $\boldsymbol{F}$, $\boldsymbol{Sl}$

$$\boldsymbol{Sg}_{ij} = \frac{\boldsymbol{Sg}_{ij}[\boldsymbol{B}^{\mathrm{T}}\boldsymbol{Tg} + 2(1-\alpha)\boldsymbol{F}^{\mathrm{T}}\boldsymbol{FSg}]_{ij}}{[(2(1-\alpha)\boldsymbol{Sg}^{\mathrm{T}}\boldsymbol{Sg} + \boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})\boldsymbol{Sg}]_{ij}}. \tag{13}$$

By fixing the matrices $\boldsymbol{B}$, $\boldsymbol{Sl}$, and $\boldsymbol{Sg}$, the updating rule for matrix $\boldsymbol{F}$ is formulated as

$$\boldsymbol{F}_{ij} = \frac{\boldsymbol{F}_{ij}[\boldsymbol{B}^{\mathrm{T}}\boldsymbol{W} + 2\alpha \boldsymbol{Sl}^{\mathrm{T}}\boldsymbol{SlF} + 2(1-\alpha)\boldsymbol{Sg}^{\mathrm{T}}\boldsymbol{SgF}]_{ij}}{[(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{BF} + 2\boldsymbol{F}^{\mathrm{T}}\boldsymbol{FF}]_{ij}}. \tag{14}$$

The above derivative formulas can be found in Ref. [32].

3.3. Evaluation of vertex essentiality

The previous parts mainly describes the construction of the objective function and the process of finding the optimal solution. By randomly giving the initial value of the matrices and iterating in light of the iterative formulas, we can get the attribute matrix $\boldsymbol{F}$ that each attribute is linearly independent. However, our ultimate aim is to identify the most influential vertices in the network. The local structure considered in the objective function is the 2nd neighborhood of the vertex on the one hand, on the other hand, the application of neighborhood rules can improve the accuracy of determining the most influential vertices. Therefore, we apply the first-order neighborhood to evaluate vertex's importance. Finally, the VCJG centrality is defined, as shown in Definition 1.

Definition 1 (**VCJG centrality**)

The matrix $\boldsymbol{F} \in \mathbb{R}_{+}^{k\times n}$ be the solution of Eq. (7), which is expressed the attributes of the network and each other is linearly independent. The VCJG (vertex centrality based on Joint nonnegative matrix factorization and graph embedding) centrality score of vertex v_i is defined as

$$f(v_i) = \sum_{p=1}^{k} \boldsymbol{F}_{pi}, \tag{15}$$

$$\mathrm{VCJG}(v_i) = (1 + f(v_i)) \sum_{v_j \in N_i} f(v_j), \tag{16}$$

where k is the number of attribute (usually $k < n/2$). The procedure of VCJG is illustrated in Algorithm 1.

Algorithm 1 The VCJG centrality algorithm

Input:
$G = \{V, E\}$: Real network;
θ: Parameter;
α: Parameter;
k: Number of attributes;
Output:
A ranking list about vertices' importance;
Part I: Acquire the matrix $\boldsymbol{U}$
Initializing matrices $\boldsymbol{B}_0$, $\boldsymbol{F}_0$, $\boldsymbol{Sl}_0$, and $\boldsymbol{Sg}_0$ with random sampling;
Calculate the error of R;
Let $R_{\mathrm{pre}} = R$; $t = 1$;
while $R_{\mathrm{pre}} \geq R$ do
    Fixing $\boldsymbol{F}_t$, $\boldsymbol{Sl}_t$, and $\boldsymbol{Sg}_t$, update matrix $\boldsymbol{B}_{t+1}$ according to Eq. (8);
    Fixing $\boldsymbol{B}_t$, $\boldsymbol{F}_t$ and $\boldsymbol{Sg}_t$, update matrix $\boldsymbol{Sl}_{t+1}$ according to Eq. (8);
    Fixing $\boldsymbol{B}_t$, $\boldsymbol{F}_t$ and $\boldsymbol{Sl}_t$, update matrix $\boldsymbol{Sg}_{t+1}$ according to Eq. (8);
    Fixing $\boldsymbol{B}_t$, $\boldsymbol{Sg}_t$ and $\boldsymbol{Sl}_t$, update matrix $\boldsymbol{F}_{t+1}$ according to Eq. (8);
    $t = t + 1$;
    Let $R_{\mathrm{pre}} = R$;
    Calculate new error R;
end while
Part 2: Acquire the VCJG score
Get the set N_i of vertex v_i's neighbors;
for $i = 1$ to $|V|$ **do**
    Calculate $f(v_i)$ of vertex v_i using Eq. (15);
    for $j = 1$ to $|N_i|$ **do**
        $\mathrm{sum}(v_i) \leftarrow \mathrm{sum}(v_i) + f(v_j)$;
    end for
    $\mathrm{VCJG}(v_i) \leftarrow (1 + f(v_i))\mathrm{sum}(v_i)$;
end for
Sort the vertices descending based on the VCJG value to get the a ranking list;
return the ranking list.

4. Algorithm analysis

The time complexity of VCJG centrality calculation is mainly composed of two parts, namely joint matrix factorization and vertex influence calculation. We need to spend $O(n)$ for obtaining the matrix $\boldsymbol{W}$. The time complexity for matrix $\boldsymbol{Tg}$ is $O(\log(n))$, which is calculated via matrix fast power algorithm. The time complexity for the graph embedding is $O(n^2)$. The running time for joint factorization is $O(n^2kr)$, where r is the number of iterations and k is the number of attributes. Then the time complexity of the first part is $O(n^2kr)$. Time $O(n \times Dist)$ is required to calculate the influence of vertex, which is the second part. Finally, the time complexity of VCJG is $O(n^2kr)$.

5. Experimental analysis

In this section, simulation strategy and the results of evaluation measures are described in detail. To assess the proposed approach, we compare its performance with four traditional algorithms and five state-of-the-art methods including:

- degree centrality (DC);
- betweenness centrality (BC);
- closeness centrality (CC);
- k-shell (KS);
- the centrality based on the inverse-square law (I);
- hierarchical k-shell centrality (HKS);
- global and local structure centrality (GLS);
- global importance of vertices centrality (GIN) where parameters a and b are set to 1;
- the centrality based on global structure model (GSM).

These algorithms evaluate the influence of vertex from different attribute of vertex. DC algorithm takes the local topology attribute. BC and CC methods think about the global topology attribute. KS and HKS indexes consider the center location of the vertex. I algorithm considers the diversity of neighbors. GLS take into account both local and global topological attribute. GIN and GSM algorithms adopt the propagation characteristics. To illustrate the performance of JNMF, we use the four traditional methods and the five recent methods as baseline methods to build vertex centrality.

5.1. Utilized datasets

Nine different real-world networks, details about which can be found in Table 1, are used to validate the performance of the aforementioned algorithm. There are include karate club network (Karate),[33] the network of selling political books about the presidential election (Polbooks),[34] Jazz musicians network (Jazz),[35] usair network (USAir),[36] co-authorship network of scientists (Netscience),[37] Email network of the University Rovira i Virgili (Email),[38] protein–protein interaction network (Yeast).[39] The user network of the hamsterster.com website (Hamster)[40] and the network of the online social networking platform (Facebook).[41] The statistics characteristics of these datasets are summarized in Table 1, where β_{th} is the infection rate threshold and β is the optimal infection rate value.

Table 1. Statistical characteristics of all datasets.

Network	$\|V\|$	$\|E\|$	Average degree	Maximum degree	Assortativity	β_{th}	β
Karate	34	78	4.588	17	–0.4756	0.129	0.13
Polbooks	105	411	8.400	25	–0.0436	0.084	0.13
Jazz	198	2742	27.967	100	0.0202	0.026	0.026
USAir	332	2126	12.81	139	–0.2079	0.023	0.045
Netscience	379	914	4.823	34	–0.0817	0.125	0.16
Email	1133	5451	9.622	71	0.0782	0.054	0.1
Yeast	2375	23386	9.8467	118	0.4539	0.029	0.07
Hamster	2426	16631	13.711	273	0.0474	0.024	0.04
Facebook	3892	17239	8.859	126	0.5605	0.0375	0.1

5.2. Experiments for the parameters

Recall that VCJG involves three parameters: parameter k denotes the number of potential attributes, parameter θ determines the weight of the l-order global topology, and parameter α governs the direction in which the vertex attributes are smoothed to the topological structure. All possible values of each parameter are considered and ranking lists under these combinations are calculated. The Kendall coefficient τ is used to evaluate the correlation between the ranking and the ranking generated by the SIR model, and finally the value of each parameter is determined.

Table 2. The value of parameter k over all networks.

Network	$\|V\|$	$\|E\|$	k
Karate	34	78	16
Polbooks	105	411	34
Jazz	198	2742	2
USAir	332	2126	1
Netscience	379	914	160
Email	1133	5451	356
Yeast	2375	23386	277
Hamster	2426	16631	459
Facebook	3892	17239	61

The number of attributes of the network is usually less than the number of vertex. Thus for simplicity, the sorted lists

under different k value are obtained via adjusting $\theta = 0.5$ and $\alpha = 0.5$, where $k \in \{1, n/2\}$. The k value is selected corresponding to the optimal Kendall coefficient τ. Table 2 is demonstrate the final selected value of k on all networks. Next, we investigate how parameter α and θ influence the performance of the proposed algorithm. By fixing k, the parameter α and θ increase from 0 and 0.1 to 1 with the gap 0.1, respectively. The variation in the accuracy of VCJG as parameter α and θ increase for four dataset is shown in Fig. 1. As can be observed in the picture, the highest values of τ can be obtained for $\theta = 0.8$ and $\alpha = 0$ over the Karate dataset, for $\theta = 0.5$ and $\alpha = 0.1$ over the Polblogs dataset, for $\theta = 0.8$ and $\alpha = 0.3$ over the Email dataset, and for $\theta = 0.8$ and $\alpha = 0.2$ over the Yeast dataset.

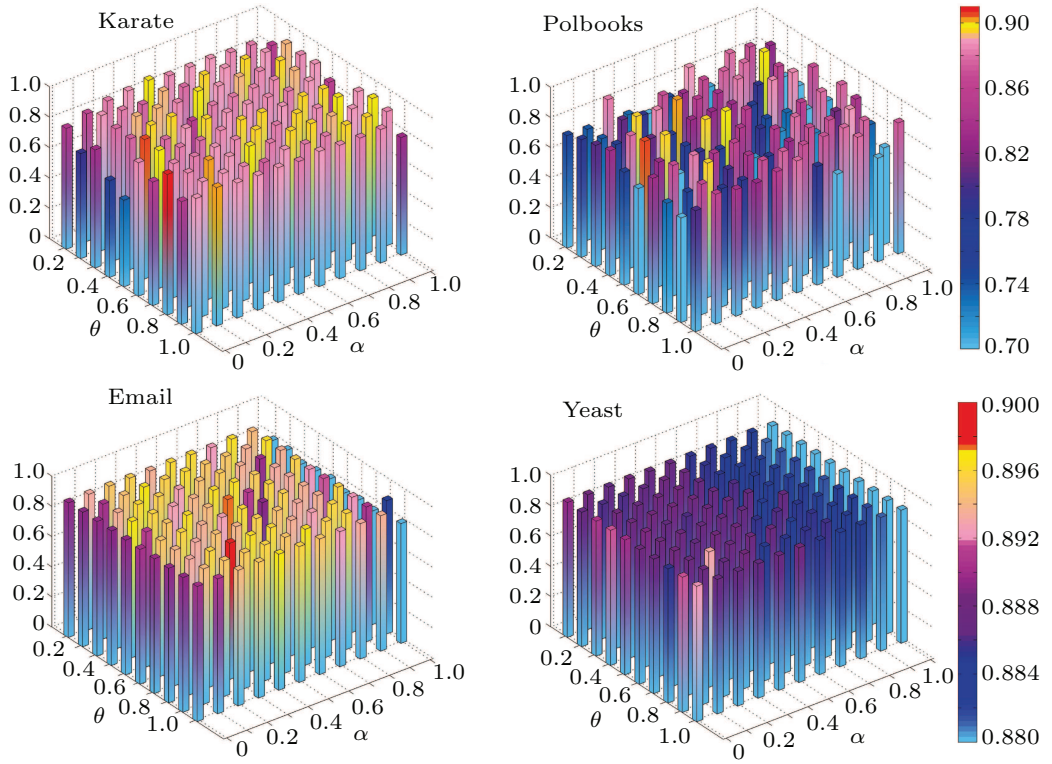


Fig. 1. Effect of variation of parameters θ and α on the accuracy of the proposed method based on Kendall's τ correlation coefficient over the four dataset.

In conclusion, as $\theta = 0.8$, VCJG reaches a good balance between the l-order global topology. In the wake of parameter α increases from 0 to 1, the accuracy of VCJG increases from low to high, and then gradually decreases. The reason is that when α is small, the attributes of vertex are excessively smoothed to global topology. When α is large, the attributes are dominated by local topology, thereby resulting in an undesirable performance. Based on the above results, the extraction of attributes and topology reaches a good balance for $\theta = 0.8$ and $\alpha = 0.2$. Thus, these values are considered for the parameters θ and α for remaining experiments.

5.3. Correlation

In order to evaluate the lists ranked by all the centrality indicators, the list ranked by the real diffusion process of the vertices requires obtaining. The standard susceptible-infected-recovered (SIR) model[42] is adopted to simulate the spreading process on networks and record the diffusion efficiency for every vertex. Every vertex pertains to one of the susceptible state (S), the infected state (I) or the recovered state (R) in the SIR model. Initially, one vertex is set to infected state and other vertices are susceptible vertices. At each step, each infected vertex can infect its susceptible neighbors with probability β, and then can be deleted with probability γ. This dynamic process will repeat until there is no infected vertices.

In this paper, the SIR process is repeated 1000 times for each vertex to increase ranking accuracy and the average value of the vertex's spreading capability is considered as it is importance. The infection probability β is set near the infection rate threshold and $\gamma = 2\beta$. The infection rate range of each network is shown in the abscissa of Fig. 2.

For measure the performance of the centrality algorithms, the Kendall's coefficient τ[43] is employ to evaluate the correlation between ranking list and the one generated by the SIR model. The Kendall's correlation coefficient τ is defined as

$$\tau = \frac{(z_c - z_d)}{0.5z(z-1)},$$

where $\tau \in [-1, +1]$. z_c and z_d denote the number of concordant and discordant pairs respectively. The higher the value of τ, the more accurate the ranking generated by the centrality measure.

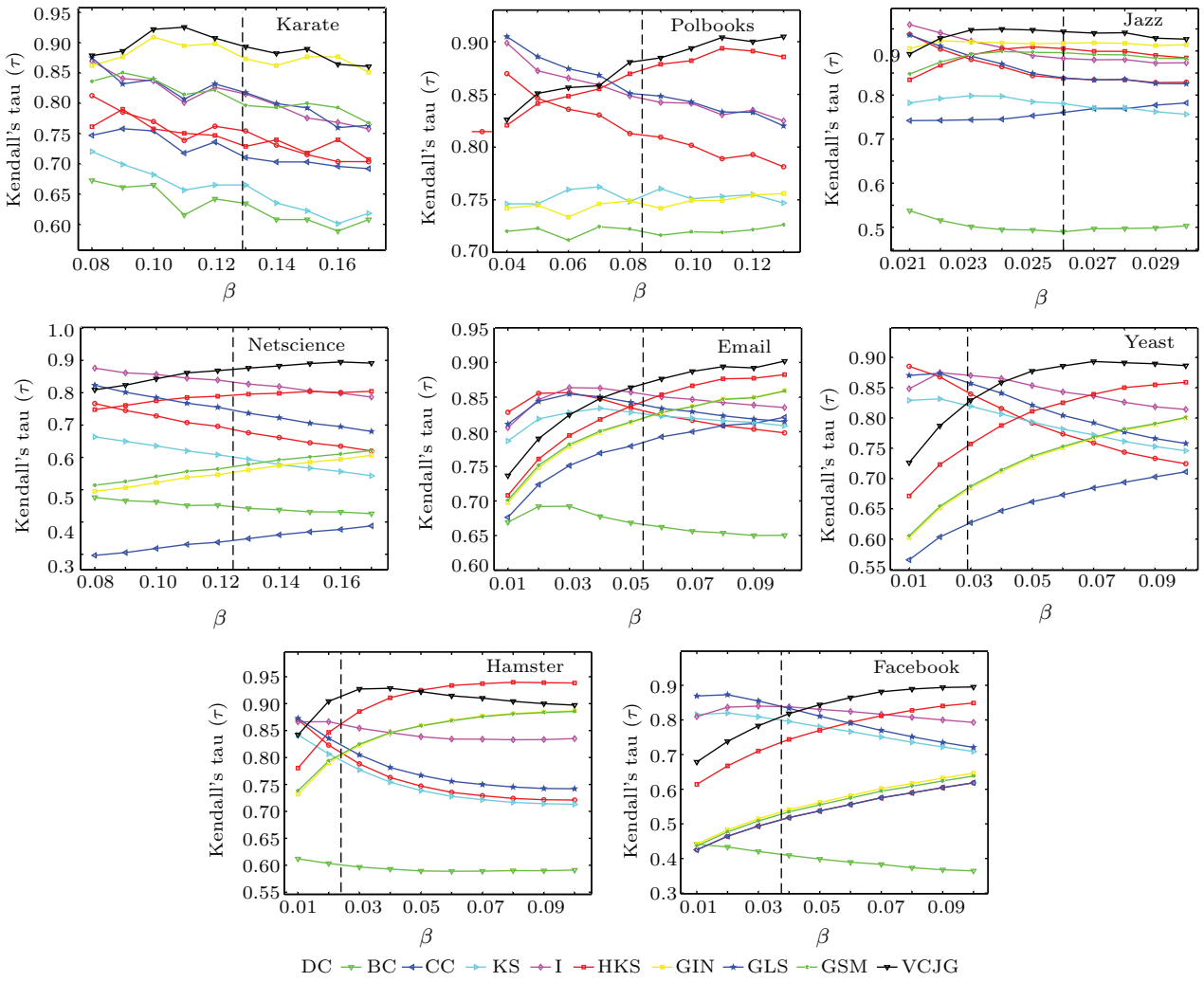


Fig. 2. The influence of the change of infection rate on the accuracy of different methods in eight datasets.

The Kendall's coefficient τ in eight real-world networks has been investigated. The calculations, to evaluate the effect of infection probability β, are shown in Fig. 2. The value of β_{th} in this figure is indicated by dash-dot lines. In Polbooks, Jazz, Netscience, Email, Yeast and Facebook datasets, as the value of β increases, especially as it exceeds β_{th}, the τ of the proposed method is more highly, and outperforms the other measures. In Karate dataset, after the infection rate threshold, the VCJG's τ shows a downward trend. Although it is lower than the GIN algorithm at $\beta = 0.16$, it surpassed it afterwards. In Hamster dataset, after $\beta = 0.05$, the τ of VCJG is higher than that of the compared algorithms except HKS. It can be seen that VCJG has different level of "difficulty" in finding influential vertices on different networks. However in most networks, VCJG still shows better performance than current state-of-the-art algorithms.

5.4. Monotonicity

In addition to the above experiments, we also evaluate the monotonicity (M)[44] of ranking for each centrality method. The M of the indexing method refers to the ability of uniquely identifying vertex from the network. The range of M lies between $[0,1]$ where 0 means all vertices in the network have the same rank and 1 means unique rank assign to each vertex in the network. Monotonicity is defined as follows:

$$M(R) = \left(1 - \frac{\sum_{r\in R} N_r \times (N_r - 1)}{N \times (N-1)}\right)^2,$$

where R is the ranking table generated by a centrality measure, N is the number of levels in sequence R, N_r is the number of vertices with rank r in the ranking table R.

Table 3 describes the ranking monotonicity of all methods on nine real datasets. It can be seen that monotonicity of VCJG method is lower than most other indexing methods in Jazz dataset. On the other hand, the monotonicity of the proposed methods is the same as HKS, I, GIN, and GSM methods in Email dataset. In the other datasets, the monotonic value of the proposed method is better than all the contrasted algorithms, and is close to 1. The results show that the VCJG method has greater ability to really distinguish the vertices than in the other methods.

Table 3. The M value of different methods in nine networks.

Network	M (DC)	M (BC)	M (CC)	M (KS)	M (HKS)	M (I)	M (GIN)	M (GLS)	M (GSM)	M (VCJG)
Karate	0.7079	0.7754	0.8993	0.4958	0.9542	0.9542	0.9542	0.9612	0.9507	**0.9999**
Polbooks	0.8252	0.9974	0.9847	0.4949	0.9996	0.9996	0.9996	0.9999	0.9996	**0.9999**
Jazz	0.9659	0.9885	0.9878	0.7944	0.9993	0.9993	0.9993	**0.9995**	0.9993	0.9560
USAir	0.8586	0.6970	0.9892	0.8114	0.9951	0.9951	0.9951	0.9951	0.9951	**0.9999**
Netscience	0.7642	0.3389	0.9928	0.6421	0.9949	0.9951	0.9954	0.9950	0.9949	**0.9999**
Email	0.8875	0.9400	0.9988	0.8089	**0.9999**	**0.9999**	**0.9999**	0.9992	**0.9999**	**0.9999**
Yeast	0.7210	0.7012	0.9964	0.6643	0.9990	0.9992	0.9992	0.9964	0.9992	**0.9999**
Hamster	0.8980	0.7123	0.9847	0.8714	0.9857	0.9857	0.9857	0.9858	0.9857	**0.9999**
Facebook	0.8487	0.8853	0.9996	0.7823	0.9997	0.9998	0.9998	0.9973	0.9998	**0.9999**

5.5. Accuracy of top-10 vertices ranking

High-ranking vertices in the ranking list are more important than low-ranking vertices, so we require evaluating the accuracy of different methods for identifying high-ranking vertices. Different centrality measures give different lists of the top-10 vertices, depending on how information in the network is considered. The number of same vertices for different measures can show the superiority of different approaches. In the experiment, the top-10 vertex ids are listed to compare the differences in the top-10 vertex results obtained by different measures. And the propagation based on the SIR model to show the superior infection ability of the vertices obtained by the centrality algorithms.

We select indicators from the contrasted algorithms that are less different from the proposed method in the correlation and monotonic experiments for further comparison, such as HKS, GLS and GSM centrality. The top-10 vertices in four datasets are identified and the results are listed in Table 4. A vertex in red color indicates that it not exists in the top-10 lists obtained by the SIR model. For the Polbooks and Email datasets, the distinctness in the number of different vertices between four metrics and the SIR model is relatively small. The distinctness is relatively large in the Jazz dataset because only the top-10 vertices found by VCJG are the same as those generated by SIR. In USAir dataset, the vertices selected by each algorithm are amount to the same those generated by SIR. Therefore, we evaluate the performance of the algorithm from the perspective of vertex arrangement order. The vertices in blue color mean that the order of them are different from the top-10 generated by SIR. For HKS and GSM criteria, there are four vertices in a different order than SIR, while GLS and VCJG methods only have two vertices and the vertices order of VCJG are lower than that of GLS.

Table 4. Vertices ranked by different centrality methods in four datasets.

Rank	Polbooks					Jazz				
	HKS	GLS	GSM	VCJG	SIR	HKS	GLS	GSM	VCJG	SIR
1	85	9	9	9	9	100	100	100	100	8
2	9	13	13	13	13	8	4	8	8	100
3	74	85	31	85	85	4	8	4	4	4
4	73	4	73	73	31	131	131	131	131	131
5	13	73	4	31	67	80	80	80	194	194
6	31	74	10	67	73	129	129	129	80	69
7	67	67	85	74	4	**5**	**5**	**5**	53	80
8	4	31	67	4	74	194	194	**32**	69	53
9	**76**	12	12	**75**	12	69	**85**	**84**	129	162
10	**75**	**48**	**41**	12	10	53	**32**	**85**	162	129
Rank	USAir					Email				
	HKS	GLS	GSM	VCJG	SIR	HKS	GLS	GSM	VCJG	SIR
1	123	123	123	123	123	105	105	105	105	105
2	57	57	57	57	57	16	16	333	16	42
3	201	201	201	201	201	42	333	42	42	333
4	207	**258**	207	207	207	196	42	23	333	23
5	258	**207**	258	258	258	333	23	76	196	41
6	**276**	276	**276**	193	193	23	41	**233**	3	196
7	**117**	25	**117**	276	276	3	196	41	23	3
8	**193**	117	**25**	**117**	25	**204**	**233**	**52**	**49**	16
9	**25**	193	**193**	**25**	117	41	**76**	3	41	21
10	302	302	302	302	302	**49**	21	**378**	**56**	76

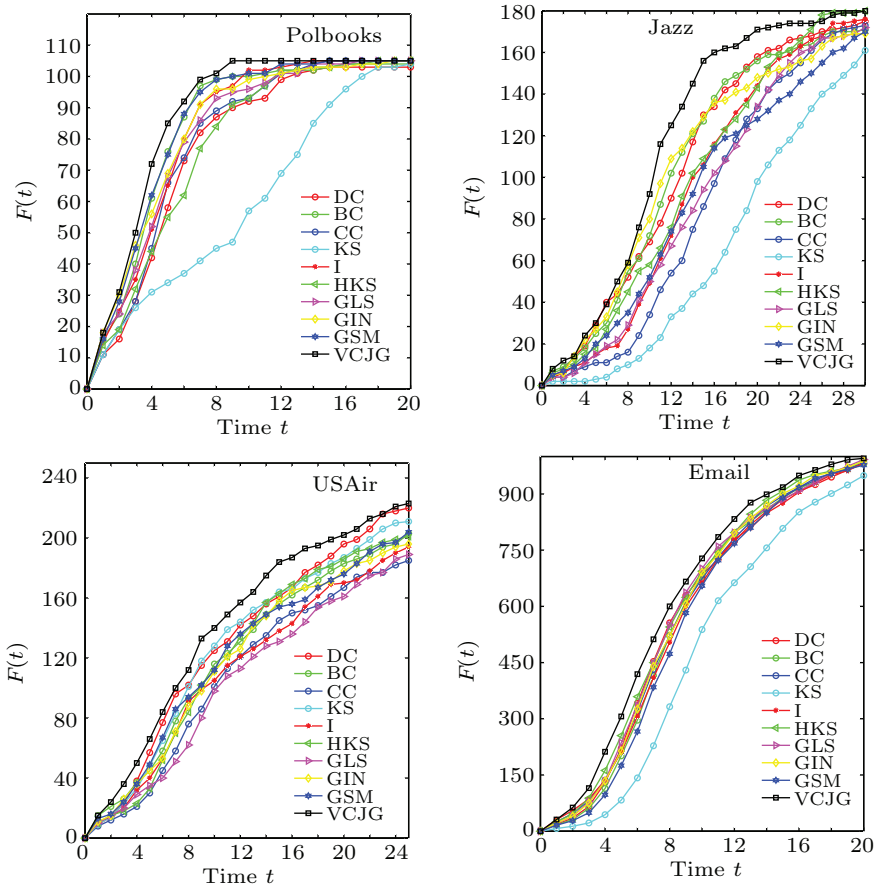


Fig. 3. Number of infected vertices for different initial vertices (top-10 vertices) obtained by the ten methods in four networks.

In order to further evaluate whether the vertices obtained by the proposed algorithm have superior infection ability, the SIR model is applied to measure the infection ability of top-10 selected initial vertices. The top-10 vertices obtained by different approaches are used as the initial infected vertices, and the remaining vertices are defined as susceptible vertices. We set the recovery rate in SIR as 0 in order to more easily and accurately count the number of infected vertices. The number of infected vertices $F(t)$ at particular moment t is selected as the infection ability of the initial infected vertices. A higher $F(t)$ indicates that the infection ability of the initial vertices is stronger and that the initial vertices are more importance.

In the experiment, the infection probability of the SIR model on each network is given in column eight of Table 1. Each experiment was carried out independently, and repeated 1000 times under the condition of a specific infection rate. The results are the average of 100 experiments, as shown in Fig. 3. Among the four networks, the $F(t)$ of VCJG has little difference with the comparison algorithms in the early stage, but it was higher than other algorithms it keeps up with the growth of time. What we want to point out is that although the list of top 10 vertices obtained by HKS, GLS, and GSM methods in the four networks are slightly different from that for VCJG, the VCJG has a stronger ability for evaluating vertices than the comparison algorithms.

6. Conclusion

In this paper, based on joint nonnegative matrix factorization and graph embedding, we proposed the VCJG centrality to find pivotal vertices in complex networks. It overcomes the shortcomings of existing centrality measures that they cannot completely guarantee completely unrelated among multiple attributes and ignore the heterogeneity between network attributes and structure. This method has good stability and accuracy in different complex networks.

However, VCJG centrality still has some inevitable problems. For instance, there is no better strategy to determine the value of parameter k. And the proposed approach has high time complexity and cannot be applied to large-scale networks. How to increase the speed of VCJG without sacrificing accuracy is a key issue. In future research, we will continue to improve.

Acknowledgement

Project supported by the National Natural Science Foundation of China (Grant Nos. 62162040 and 11861045).

References

[1] Gallos L K, Makse H A and Sigman M 2012 *Proc. Natl. Acad. Sci. USA* **109** 2825
[2] Wang X Y and Zhao T F 2017 *Commun. Nonlinear. Sci. Numer. Simul.* **48** 63
[3] Xu D G, XU X Y, Xie Y F and Yang C H 2017 *Commun. Nonlinear Sci. Numer. Simul.* **48** 200
[4] Helbing D, Brockmann D and Chadefaux T 2015 *J. Stat. Phys.* **158** 735
[5] Sheikhahmadi A, Nematbakhsh M A and Zareie A 2017 *Physica A* **486** 517
[6] Hu K, HU T and Tang Y 2010 *Chin. Phys. B* **19** 080206
[7] Buldyrev S V, Parshani R, Paul G, Stanley H E and Havlin S 2010 *Nature* **464** 1025
[8] PastorSatorras R and PastorSatorras A 2002 *Phys. Rev. E* **65** 036104
[9] Freeman L C 1978 *Soc. Netw.* **1** 215
[10] Fei L G, Zhang Q and Deng Y 2018 *Physica A* **512** 1044
[11] Zareie A, Sheikhahmadi A and Jalili M 2019 *Future Gener. Comput. Syst.* **94** 120
[12] Zareie A, Sheikhahmadi A, Jalili M and Fasaei M S K 2020 *Knowl. Based Syst.* **194** 10885
[13] Yang X H, Xiong Z, Ma F, Chen X Z, Ruan Z Y, Jiang P and XU X L 2021 *Physica A* **573** 125971
[14] Freeman L C 1977 *Sociometry* **40** 35
[15] Sabidussi G 1966 *Psychometrika* **31** 581
[16] Kitsak M, Kitsak L K, Havlin S, Liljeros F, Muchnik L, Stanley H E and Makse H A 2010 *Nat. Phys.* **6** 888
[17] Zareie A and Sheikhahmadi A 2018 *Expert Syst. Appl.* **93** 200
[18] Zhao J, Wang Y C and Deng Y 2020 *Chaos, Solitons and Fractals* **133** 0960
[19] Ullah A, Wang B, Sheng J and Sun Z 2021 *Sci. Rep.* **11** 6173
[20] Sheng J, Dai J Y, Wang B, Duan G, Long J, Zhang J, Guan K, Hu S, Chen L and Guan W H 2020 *Physica A* **541** 0378
[21] Zhang J L, Fu Y J, Cheng L and Yang Y Y 2021 *Physica A* **571** 125791
[22] Lee D D and Seung H S 1999 *Nature* **401** 788
[23] Du Y X, Hu Y, Mahadevan S, Gao C and Deng Y 2014 *Physica A* **399** 57
[24] Yang Y Z, Yu L, Zhou Z L and Chen Y 2019 *Physica A* **526** 121118
[25] Yan X L, Cui Y P and Ni S J 2020 *Chin. Phys. B* **29** 048902
[26] Yang Y Z, Yu L, Zhou Z L, Chen Y and Kou T 2019 *Math. Probl. Eng.* **2019** 1
[27] Grover A and Leskovec J 2016 *the 22nd ACM SIGKDD International Conference* 2016 p. 855
[28] Biggs N 1974 *Algebraic Graph Theory* (New York: the Press Syndicate of the University of Cambridge) p. 9
[29] Wang Y C 2021 *Diagital Technology and Application* 39
[30] Xue B G, Li S C, Yu R and Zhao Q 2019 *Appl. Math. Comput.* **340** 156
[31] Horn R A 1990 *Proc. Symp. Appl. Math* **40** 87
[32] Petersen K B and Pedersen M S 2008 *The Matrix Cookbook*
[33] Zachary W W 1977 *J. Anthropol. Res.* **33** 452
[34] Lusseau D, Schneider K, Boisseau O J, Haase P, Slooten E and Dawson S 2003 *Behav. Ecol. Sociobiol* **54** 396
[35] Gleiser P M and Danon L 2003 *Adv. Complex Syst.* **6** 565
[36] Lv Z W, Zhao N, Xiong F and Chen N 2019 *Physica A* **523** 488
[37] Newman M E 2006 *Phys. Rev. E* **74** 036104
[38] Guimera R, Danon L, Diaz-Guilera A, Giralt F and Arenas A 2003 *Phys. Rev. E* **68** 065103
[39] Jeong H, Mason S P and Barabasi A L 2001 *Nature* **411** 41
[40] Zareie A, Sheikhahmadi A and Jalili M 2019 *Future Gener. Comp. Sy.* **94** 120
[41] Mcauley J and Leskovec J 2012 *Proceedings of the* 26*th International Conference on Neural Information Processing Systems* 2012
[42] Huang C Y, Lee C L and Wen T H 2013 *J. Syst. Software* **86** 801
[43] Kendall M G 1945 *Biometrika* **33** 239
[44] Bae J and Kim S 2014 *Physica A* **395** 549
[45] Webber W, Moffat A and Zobel J 2010 *Acm T. Inform. Syst.* **28** 1